```
/******************************************************************************
* @file    stm32f10x_cl.h
* @purpose: CMSIS Cortex-M3 Core Peripheral Access Layer Header File for the
*           ST STM32F10x Connectivity Line Device Series
* @version: V1.02
* @date:    22. December 2009
*------------------------------------------------------------------------------
*
* Copyright (C) 2009 ARM Limited. All rights reserved.
*
* ARM Limited (ARM) is supplying this software for use with Cortex-Mx
* processor based microcontrollers.  This file can be freely distributed
* within development tools that are supporting such ARM based processors.
*
* THIS SOFTWARE IS PROVIDED "AS IS". NO WARRANTIES, WHETHER EXPRESS, IMPLIED
* OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF
* MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE APPLY TO THIS SOFTWARE.
* ARM SHALL NOT, IN ANY CIRCUMSTANCES, BE LIABLE FOR SPECIAL, INCIDENTAL, OR
* CONSEQUENTIAL DAMAGES, FOR ANY REASON WHATSOEVER.
*
******************************************************************************/

#ifndef __STM32F10x_CL_H__
#define __STM32F10x_CL_H__

/*
* ==========================================================================
* ---------- Interrupt Number Definition -----------------------------------
* ==========================================================================
*/

typedef enum IRQn
{
/****** Cortex-M3 Processor Exceptions Numbers ***************************************************/
  NonMaskableInt_IRQn        = -14,    /*!< 2 Non Maskable Interrupt                */
  MemoryManagement_IRQn      = -12,    /*!< 4 Cortex-M3 Memory Management Interrupt */
  BusFault_IRQn              = -11,    /*!< 5 Cortex-M3 Bus Fault Interrupt         */
  UsageFault_IRQn            = -10,    /*!< 6 Cortex-M3 Usage Fault Interrupt       */
  SVCall_IRQn                = -5,     /*!< 11 Cortex-M3 SV Call Interrupt          */
  DebugMonitor_IRQn          = -4,     /*!< 12 Cortex-M3 Debug Monitor Interrupt    */
  PendSV_IRQn                = -2,     /*!< 14 Cortex-M3 Pend SV Interrupt          */
  SysTick_IRQn               = -1,     /*!< 15 Cortex-M3 System Tick Interrupt      */

/****** STM32 specific Interrupt Numbers *********************************************************/
  WWDG_IRQn                  = 0,      /*!< Window WatchDog Interrupt               */
  PVD_IRQn                   = 1,      /*!< PVD through EXTI Line detection Interrupt */
  TAMPER_IRQn                = 2,      /*!< Tamper Interrupt                        */
  RTC_IRQn                   = 3,      /*!< RTC global Interrupt                    */
  FLASH_IRQn                 = 4,      /*!< FLASH global Interrupt                  */
  RCC_IRQn                   = 5,      /*!< RCC global Interrupt                    */
  EXTI0_IRQn                 = 6,      /*!< EXTI Line0 Interrupt                    */
  DMA1_Channel1_IRQn         = 11,     /*!< DMA1 Channel 1 global Interrupt         */
  DMA1_Channel7_IRQn         = 17,     /*!< DMA1 Channel 7 global Interrupt         */
  ADC_IRQn                   = 18,     /*!< ADC global Interrupt                    */
  CAN1_TX_IRQn               = 19,     /*!< CAN1 TX Interrupts                      */
  CAN1_RX0_IRQn              = 20,     /*!< CAN1 RX0 Interrupts                     */
  CAN1_RX1_IRQn              = 21,     /*!< CAN1 RX1 Interrupt                      */
```



```
  CAN1_SCE_IRQn              = 22,     /*!< CAN1 SCE Interrupt                      */
  EXTI9_5_IRQn               = 23,     /*!< External Line[9:5] Interrupts           */
  TIM1_BRK_IRQn              = 24,     /*!< TIM1 Break Interrupt                    */
  TIM4_IRQn                  = 30,     /*!< TIM4 global Interrupt                   */
  I2C1_EV_IRQn               = 31,     /*!< I2C1 Event Interrupt                    */
  I2C2_ER_IRQn               = 34,     /*!< I2C2 Error Interrupt                    */
  SPI1_IRQn                  = 35,     /*!< SPI1 global Interrupt                   */
  SPI2_IRQn                  = 36,     /*!< SPI2 global Interrupt                   */
  USART1_IRQn                = 37,     /*!< USART1 global Interrupt                 */
  USART2_IRQn                = 38,     /*!< USART2 global Interrupt                 */
  USART3_IRQn                = 39,     /*!< USART3 global Interrupt                 */
  EXTI15_10_IRQn             = 40,     /*!< External Line[15:10] Interrupts         */
  RTCAlarm_IRQn              = 41,     /*!< RTC Alarm through EXTI Line Interrupt   */
  OTG_FS_WKUP_IRQn           = 42,     /*!< USB On-The-Go FS Wakeup through EXTI Line Interrupt */
  TIM5_IRQn                  = 50,     /*!< TIM5 global Interrupt                   */
  SPI3_IRQn                  = 51,     /*!< SPI3 global Interrupt                   */
  UART4_IRQn                 = 52,     /*!< UART4 global Interrupt                  */
  UART5_IRQn                 = 53,     /*!< UART5 global Interrupt                  */
  TIM6_IRQn                  = 54,     /*!< TIM6 global Interrupt                   */
  TIM7_IRQn                  = 55,     /*!< TIM7 global Interrupt                   */
  DMA2_Channel1_IRQn         = 56,     /*!< DMA2 Channel1 global Interrupt          */
  DMA2_Channel5_IRQn         = 60,     /*!< DMA2 Channel5 global Interrupts         */
  ETH_IRQn                   = 61,     /*!< Ethernet global interrupt               */
  ETH_WKUP_IRQn              = 62,     /*!< Ethernet Wakeup through EXTI line interrupt */
  CAN2_TX_IRQn               = 63,     /*!< CAN2 TX interrupts                      */
  CAN2_RX0_IRQn              = 64,     /*!< CAN2 RX0 interrupts                     */
  CAN2_RX1_IRQn              = 65,     /*!< CAN2 RX1 interrupt                      */
  CAN2_SCE_IRQn              = 66,     /*!< CAN2 SCE interrupt                      */
  OTG_FS_IRQn                = 67,     /*!< USB On The Go FS global interrupt       */
} IRQn_Type;

/*
* ==========================================================================
* ---------- Processor and Core Peripheral Section -------------------------
* ==========================================================================
*/
/* Configuration of the Cortex-M3 Processor and Core Peripherals */
#define __MPU_PRESENT             0       /*!< STM32 does not provide a MPU present or not */
#define __NVIC_PRIO_BITS          4       /*!< STM32 uses 4 Bits for the Priority Levels   */
#define __Vendor_SysTickConfig    0       /*!< Set to 1 if different SysTick Config is used */

#include <core_cm3.h>                     /* Cortex-M3 processor and core peripherals    */
#include "system_stm32f10x_cl.h"          /* STM32 System                                */

/******************************************************************************/
/*                Device Specific Peripheral registers structures             */
/******************************************************************************/
/*------------- Analog to Digital Converter ----------------------------------*/
typedef struct
{
  __IO uint32_t SR;
  __IO uint32_t CR1;
  __IO uint32_t CR2;
  __IO uint32_t SMPR1;
  __IO uint32_t SMPR2;
  __IO uint32_t JOFR1;
  __IO uint32_t JOFR2;
```



```
  __IO uint32_t JOFR3;
  __IO uint32_t JOFR4;
  __IO uint32_t HTR;
  __IO uint32_t LTR;
  __IO uint32_t SQR1;
  __IO uint32_t SQR2;
  __IO uint32_t SQR3;
  __IO uint32_t JSQR;
  __IO uint32_t JDR1;
  __IO uint32_t JDR2;
  __IO uint32_t JDR3;
  __IO uint32_t JDR4;
  __IO uint32_t DR;
} ADC_TypeDef;

/*------------- Digital to Analog Converter ----------------------------------*/
typedef struct
{
  __IO uint32_t CR;
  __IO uint32_t SWTRIGR;
  __IO uint32_t DHR12R1;
  __IO uint32_t DHR12L1;
  __IO uint32_t DHR8R1;
  __IO uint32_t DHR12R2;
  __IO uint32_t DHR12L2;
  __IO uint32_t DHR8R2;
  __IO uint32_t DHR12RD;
  __IO uint32_t DHR12LD;
  __IO uint32_t DHR8RD;
  __IO uint32_t DOR1;
  __IO uint32_t DOR2;
} DAC_TypeDef;

/*------------- Backup Registers ---------------------------------------------*/
typedef struct
{  ..} BKP_TypeDef;

/*------------- Controller Area Network --------------------------------------*/
typedef struct
{
  __IO uint32_t TIR;
  __IO uint32_t TDTR;
  __IO uint32_t TDLR;
  __IO uint32_t TDHR;
} CAN_TxMailBox_TypeDef;

typedef struct
{
  __IO uint32_t RIR;
  __IO uint32_t RDTR;
  __IO uint32_t RDLR;
  __IO uint32_t RDHR;
} CAN_FIFOMailBox_TypeDef;

typedef struct
{
  __IO uint32_t FR1;
```



```
  __IO uint32_t FR2;
} CAN_FilterRegister_TypeDef;

typedef struct
{
  __IO uint32_t MCR;
  __IO uint32_t MSR;
  __IO uint32_t TSR;
  __IO uint32_t RF0R;
  __IO uint32_t RF1R;
  __IO uint32_t IER;
  __IO uint32_t ESR;
  __IO uint32_t BTR;
       uint32_t RESERVED0[88];
  CAN_TxMailBox_TypeDef      sTxMailBox[3];
  CAN_FIFOMailBox_TypeDef    sFIFOMailBox[2];
       uint32_t RESERVED1[12];
  __IO uint32_t FMR;
  __IO uint32_t FM1R;
       uint32_t RESERVED2;
  __IO uint32_t FS1R;
       uint32_t RESERVED3;
  __IO uint32_t FFA1R;
       uint32_t RESERVED4;
  __IO uint32_t FA1R;
       uint32_t RESERVED5[8];
  CAN_FilterRegister_TypeDef sFilterRegister[28];
} CAN_TypeDef;

/*------------- DMA Controller -----------------------------------------------*/
typedef struct
{
  __IO uint32_t CCR;
  __IO uint32_t CNDTR;
  __IO uint32_t CPAR;
  __IO uint32_t CMAR;
} DMA_CH_TypeDef;

typedef struct
{
  __IO uint32_t ISR;
  __IO uint32_t IFCR;
} DMA_TypeDef;

/*------------- External Interrupt/Event Controller --------------------------*/
typedef struct
{
  __IO uint32_t IMR;
  __IO uint32_t EMR;
  __IO uint32_t RTSR;
  __IO uint32_t FTSR;
  __IO uint32_t SWIER;
  __IO uint32_t PR;
} EXTI_TypeDef;

/*------------- Flash Memory Interface and Option Bytes Registers ------------*/
typedef struct
```

```
{ ..} FLASH_TypeDef;

typedef struct
{ ..} OB_TypeDef;

/*------------------- General Purpose and Alternate Function IO --------*/
typedef struct
{
  __IO uint32_t CRL;
  __IO uint32_t CRH;
  __IO uint32_t IDR;
  __IO uint32_t ODR;
  __IO uint32_t BSRR;
  __IO uint32_t BRR;
  __IO uint32_t LCKR;
} GPIO_TypeDef;

typedef struct
{
  __IO uint32_t EVCR;
  __IO uint32_t MAPR;
  __IO uint32_t EXTICR[4];
} AFIO_TypeDef;
/*------------------- Inter-integrated Circuit Interface ---------------*/
typedef struct
{ ..} I2C_TypeDef;
/*------------------- Independent Watchdog -----------------------------*/
typedef struct
{  ..} IWDG_TypeDef;
/*------------------- Power Control ------------------------------------*/
typedef struct
{..} PWR_TypeDef;
/*------------------- Reset and Clock Control --------------------------*/
typedef struct
{
  __IO uint32_t CR;
  __IO uint32_t CFGR;
  __IO uint32_t CIR;
  __IO uint32_t APB2RSTR;
  __IO uint32_t APB1RSTR;
  __IO uint32_t AHBENR;
  __IO uint32_t APB2ENR;
  __IO uint32_t APB1ENR;
  __IO uint32_t BDCR;
  __IO uint32_t CSR;
  __IO uint32_t AHBSTR;
  __IO uint32_t CFGR2;
} RCC_TypeDef;

/*------------------- Real-Time Clock ----------------------------------*/
typedef struct
{..} RTC_TypeDef;
/*------------------- Serial Peripheral Interface ----------------------*/
typedef struct
{ ..} SPI_TypeDef;
/*------------------- Advanced Control Timer ---------------------------*/
typedef struct
```



```
{..} TIM_AC_TypeDef;
/*------------------- General Purpose Timer ----------------------------*/
typedef struct
{..} TIM_GP_TypeDef;
/*------------------- Basic Timer --------------------------------------*/
typedef struct
{..} TIM_B_TypeDef;
/*-------------- Universal Synchronous Asynchronous Receiver Transmitter --*/
typedef struct
{
  __IO uint16_t SR;
       uint16_t RESERVED0;
  __IO uint16_t DR;
       uint16_t RESERVED1;
  __IO uint16_t BRR;
       uint16_t RESERVED2;
  __IO uint16_t CR1;
       uint16_t RESERVED3;
  __IO uint16_t CR2;
       uint16_t RESERVED4;
  __IO uint16_t CR3;
       uint16_t RESERVED5;
  __IO uint16_t GTPR;
       uint16_t RESERVED6;
} USART_TypeDef;
/*------------------- Window Watchdog ----------------------------------*/
typedef struct
{ ..} WWDG_TypeDef;
/*------------------- CRC Calculation Unit -----------------------------*/
typedef struct
{
  __IO uint32_t DR;
  __IO uint32_t IDR;
  __IO uint32_t CR;
} CRC_TypeDef;

/*------------------- USB Full Speed Device Interface (USB_FS) ---------*/
typedef struct
{
  __IO uint32_t EP0R;
  __IO uint32_t EP1R;
  __IO uint32_t EP2R;
  __IO uint32_t EP3R;
  __IO uint32_t EP4R;
  __IO uint32_t EP5R;
  __IO uint32_t EP6R;
  __IO uint32_t EP7R;
       uint32_t RESERVED0[8];
  __IO uint32_t CNTR;
  __IO uint32_t ISTR;
  __IO uint32_t FNR;
  __IO uint32_t DADDR;
  __IO uint32_t BTABLE;
} USB_FS_TypeDef;
/*----- Ethernet (ETH): media access control (MAC) with DMA controller -----*/
typedef struct
{..} ETH_TypeDef;
```



```
/*----- USB On-The-Go Full Speed (OTG_FS) Controller -------------------*/
typedef struct
{..} OTG_FS_TypeDef;
/******************************************************************************/
/*                        Peripheral memory map                               */
/******************************************************************************/
/* Peripheral and SRAM base address in the alias region */
#define PERIPH_BB_BASE        ((uint32_t)0x42000000)
#define SRAM_BB_BASE          ((uint32_t)0x22000000)
/* Peripheral and SRAM base address in the bit-band region */
#define SRAM_BASE             ((uint32_t)0x20000000)
□   #define PERIPH_BASE       ((    uint32_t)[illegible])
/* Flash Option Bytes base address */
#define OB_BASE               ((uint32_t)0x1FFFF800)

/* Peripheral memory map */
□   #define APB1PERIPH_BASE       PERIPH_BASE
□   #define APB2PERIPH_BASE       (PERIPH_BASE + [illegible])
#define AHBPERIPH_BASE        (PERIPH_BASE + 0x20000)

#define TIM2_BASE             (APB1PERIPH_BASE + 0x0000)
..#define TIM7_BASE            (APB1PERIPH_BASE + [illegible])
#define RTC_BASE              (APB1PERIPH_BASE + 0x2800)
#define WWDG_BASE             (APB1PERIPH_BASE + 0x2C00)
#define IWDG_BASE             (APB1PERIPH_BASE + 0x3000)
#define SPI2_BASE             (APB1PERIPH_BASE + 0x3800)
#define SPI3_BASE             (APB1PERIPH_BASE + 0x3C00)
□   #define USART2_BASE       (APB1PERIPH_BASE + [illegible])
#define USART3_BASE           (APB1PERIPH_BASE + 0x4800)
..#define USART5_BASE          (APB1PERIPH_BASE + [illegible])
#define I2C1_BASE             (APB1PERIPH_BASE + 0x5400)
#define I2C2_BASE             (APB1PERIPH_BASE + 0x5800)
#define USB_FS_BASE           (APB1PERIPH_BASE + 0x5C00)
#define CAN1_BASE             (APB1PERIPH_BASE + 0x6400)
#define CAN2_BASE             (APB1PERIPH_BASE + 0x6800)
#define BKP_BASE              (APB1PERIPH_BASE + 0x6C00)
#define PWR_BASE              (APB1PERIPH_BASE + 0x7000)
#define DAC_BASE              (APB1PERIPH_BASE + 0x7400)

#define AFIO_BASE             (APB2PERIPH_BASE + 0x0000)
#define EXTI_BASE             (APB2PERIPH_BASE + 0x0400)
#define GPIOA_BASE            (APB2PERIPH_BASE + 0x0800)
..#define GPIOE_BASE           (APB2PERIPH_BASE + [illegible])
#define ADC1_BASE             (APB2PERIPH_BASE + 0x2400)
#define ADC2_BASE             (APB2PERIPH_BASE + 0x2800)
#define TIM1_BASE             (APB2PERIPH_BASE + 0x2C00)
#define SPI1_BASE             (APB2PERIPH_BASE + 0x3000)
#define USART1_BASE           (APB2PERIPH_BASE + 0x3800)

#define DMA1_BASE             (AHBPERIPH_BASE + 0x0000)
#define DMA1_CH1_BASE         (AHBPERIPH_BASE + 0x0008)
..#define DMA1_CH7_BASE        (AHBPERIPH_BASE + [illegible])
#define DMA2_BASE             (AHBPERIPH_BASE + 0x0400)
#define DMA2_CH1_BASE         (AHBPERIPH_BASE + 0x0408)
..#define DMA2_CH7_BASE        (AHBPERIPH_BASE + [illegible])
#define RCC_BASE              (AHBPERIPH_BASE + 0x1000)
#define FLASH_BASE            (AHBPERIPH_BASE + 0x2000)
```



```
#define CRC_BASE              (AHBPERIPH_BASE + 0x3000)
#define ETH_BASE              (AHBPERIPH_BASE + 0x8000)
#define OTG_FS_BASE           (APB1PERIPH_BASE + 0x10000000)

/******************************************************************************/
/*                        Peripheral declaration                              */
/******************************************************************************/
#define TIM2                  ((TIM_GP_TypeDef *) TIM2_BASE       )
..#define TIM7                 ((TIM_B_TypeDef   *) TIM7_BASE      )
#define RTC                   ((RTC_TypeDef     *) RTC_BASE       )
#define WWDG                  ((WWDG_TypeDef    *) WWDG_BASE      )
#define IWDG                  ((IWDG_TypeDef    *) IWDG_BASE      )
#define SPI2                  ((SPI_TypeDef     *) SPI2_BASE      )
#define SPI3                  ((SPI_TypeDef     *) SPI3_BASE      )
□   #define USART2            ((USART_TypeDef   *) USART2_BASE    )
#define USART3                ((USART_TypeDef   *) USART3_BASE    )
#define USART4                ((USART_TypeDef   *) USART4_BASE    )
#define USART5                ((USART_TypeDef   *) USART5_BASE    )
#define I2C1                  ((I2C_TypeDef     *) I2C1_BASE      )
#define I2C2                  ((I2C_TypeDef     *) I2C2_BASE      )
#define USB_FS                ((USB_FS_TypeDef  *) USB_FS_BASE    )
#define CAN1                  ((CAN_TypeDef     *) CAN1_BASE      )
#define CAN2                  ((CAN_TypeDef     *) CAN2_BASE      )
#define BKP                   ((BKP_TypeDef     *) BKP_BASE       )
#define PWR                   ((PWR_TypeDef     *) PWR_BASE       )
#define DAC                   ((DAC_TypeDef     *) DAC_BASE       )
#define AFIO                  ((AFIO_TypeDef    *) AFIO_BASE      )
#define EXTI                  ((EXTI_TypeDef    *) EXTI_BASE      )
#define GPIOA                 ((GPIO_TypeDef    *) GPIOA_BASE     )
..#define GPIOE                ((GPIO_TypeDef    *) GPIOE_BASE     )
#define ADC1                  ((ADC_TypeDef     *) ADC1_BASE      )
#define ADC2                  ((ADC_TypeDef     *) ADC2_BASE      )
#define TIM1                  ((TIM_AC_TypeDef  *) TIM1_BASE      )
#define SPI1                  ((SPI_TypeDef     *) SPI1_BASE      )
#define USART1                ((USART_TypeDef   *) USART1_BASE    )
#define DMA1                  ((DMA_TypeDef     *) DMA1_BASE      )
#define DMA1_CH1              ((DMA_CH_TypeDef  *) DMA1_CH1_BASE)
..#define DMA1_CH7             ((DMA_CH_TypeDef  *) DMA1_CH7_BASE)
#define DMA2                  ((DMA_TypeDef     *) DMA2_BASE      )
#define DMA2_CH1              ((DMA_CH_TypeDef  *) DMA2_CH1_BASE)
..#define DMA2_CH7             ((DMA_CH_TypeDef  *) DMA2_CH7_BASE)
#define RCC                   ((RCC_TypeDef     *) RCC_BASE       )
#define FLASH                 ((FLASH_TypeDef   *) FLASH_BASE     )
#define CRC                   ((CRC_TypeDef     *) CRC_BASE       )
#define ETH                   ((ETH_TypeDef     *) ETH_BASE       )
#define OTG_FS                ((OTG_FS_TypeDef  *) OTG_FS_BASE    )
#define OB                    ((OB_TypeDef      *) OB_BASE        )

#endif
```