/\*

\* STM32F4xx\_gpio\_driver.h

\*

\* Created on: Mar 5, 2021

\* Author: 99003772

\*/

**#ifndef** DRIVERS\_INC\_STM32F4XX\_H\_

**#define** DRIVERS\_INC\_STM32F4XX\_H\_

**#include** <stdint.h>

**#include**<stdio.h>

//general macros

**#define** DISABLE 0

**#define** ENABLE 1

**#define** SET 1

**#define** RESET 0

//Macros for different memory

**#define** SRAM1 0x20000000U

**#define** SRAM2 0x2001C000U

**#define** FLASH 0x08000000U

**#define** ROM 0x1FFFFFFFU

//macros for bus system

**#define** BUS\_BASE\_ADDR 0x40000000U

**#define** APB1\_BASEADDR 0x40000000U

**#define** APB2\_BASEADDR 0x40010000U

**#define** AHB1\_BASEADDR 0x40020000U

**#define** AHB2\_BASEADDR 0x50000000U

//MACROS OF GPIO

**#define** GPIOA\_BASEADDR (AHB1\_BASEADDR + 0x0000)

**#define** GPIOB\_BASEADDR (AHB1\_BASEADDR + 0x0400)

**#define** GPIOC\_BASEADDR (AHB1\_BASEADDR + 0x0800)

**#define** GPIOD\_BASEADDR (AHB1\_BASEADDR + 0x0C00)

**#define** GPIOE\_BASEADDR (AHB1\_BASEADDR + 0x1000)

**#define** GPIOF\_BASEADDR (AHB1\_BASEADDR + 0x1400)

**#define** GPIOG\_BASEADDR (AHB1\_BASEADDR + 0x1800)

**#define** GPIOH\_BASEADDR (AHB1\_BASEADDR + 0x1C00)

**#define** GPIOI\_BASEADDR (AHB1\_BASEADDR + 0x2000)

**#define** GPIOJ\_BASEADDR (AHB1\_BASEADDR + 0x2400)

**#define** GPIOK\_BASEADDR (AHB1\_BASEADDR + 0x2800)

//macros for peripherals Hanging AHB1

**#define** CRC\_BASEADDR (AHB1\_BASEADDR + 0x3000)

**#define** RCC\_BASEADDR (AHB1\_BASEADDR + 0x3800)

**#define** FLASH\_AHB1BASEADDR (AHB1\_BASEADDR + 0x3C00)

**#define** DMA1\_BASEADDR (AHB1\_BASEADDR + 0x6000)

**#define** DMA2\_BASEADDR (AHB1\_BASEADDR + 0x6400)

**#define** DMA2D\_BASEADDR (AHB1\_BASEADDR + 0xB000)

**#define** BKPSRAM\_BASEADDR (AHB1\_BASEADDR + 0x4000)

**#define** ETHEERNETMAC\_BASEADDR (AHB1\_BASEADDR + 0x8000)

//Macros for peripherals Hanging onto AHB2

**#define** USBOTGFS\_BASEADDR (AHB2\_BASEADDR + 0x0000)

**#define** DCMI\_BASEADDR 0x50050000U

**#define** CRYP\_BASEADDR 0x50060000U

**#define** HASH\_BASEADDR 0x50060400U

**#define** RNG\_BASEADDR 0x50060800U

//Macros for peripherals Hanging onto APB1

**#define** TIM2\_BASEADDR (APB1\_BASEADDR + 0x0000)//0x40000000U

**#define** TIM3\_BASEADDR (APB1\_BASEADDR + 0x0400)

**#define** TIM4\_BASEADDR (APB1\_BASEADDR + 0x0800)

**#define** TIM5\_BASEADDR (APB1\_BASEADDR + 0x0C00)

**#define** TIM6\_BASEADDR (APB1\_BASEADDR + 0x1000)

**#define** TIM7\_BASEADDR (APB1\_BASEADDR + 0x1400)

**#define** TIM12\_BASEADDR (APB1\_BASEADDR + 0x1800)

**#define** TIM13\_BASEADDR (APB1\_BASEADDR + 0x1C00)

**#define** TIM14\_BASEADDR (APB1\_BASEADDR + 0x2000)

**#define** RTCBKP\_BASEADDR (APB1\_BASEADDR + 0x2800)

**#define** WDDG\_BASEADDR (APB1\_BASEADDR + 0x2C00)

**#define** IWDG\_BASEADDR (APB1\_BASEADDR + 0x3000)

**#define** I2S2EXT\_BASEADDR (APB1\_BASEADDR + 0x3400)

**#define** SPI2\_BASEADDR (APB1\_BASEADDR + 0x3800)

**#define** SPI3\_BASEADDR (APB1\_BASEADDR + 0x3C00)

**#define** I2S3EXT\_BASEADDR (APB1\_BASEADDR + 0x4000)

**#define** USART2\_BASEADDR (APB1\_BASEADDR + 0x4400)

**#define** USART3\_BASEADDR (APB1\_BASEADDR + 0x4800)

**#define** UART4\_BASEADDR (APB1\_BASEADDR + 0x4C00)

**#define** UART5\_BASEADDR (APB1\_BASEADDR + 0x5000)

**#define** I2C1\_BASEADDR (APB1\_BASEADDR + 0x5400)

**#define** I2C2\_BASEADDR (APB1\_BASEADDR + 0x5800)

**#define** I2C3\_BASEADDR (APB1\_BASEADDR + 0x5C00)

**#define** CAN1\_BASEADDR (APB1\_BASEADDR + 0x6400)

**#define** CAN2\_BASEADDR (APB1\_BASEADDR + 0x6800)

**#define** PWR\_BASEADDR (APB1\_BASEADDR + 0x7000)

**#define** DAC\_BASEADDR (APB1\_BASEADDR + 0x7400)

**#define** UART7\_BASEADDR (APB1\_BASEADDR + 0x7800)

**#define** UART8\_BASEADDR (APB1\_BASEADDR + 0x7C00)

//Macros for peripherals Hanging onto APB2

**#define** TIM1\_BASEADDR (APB2\_BASEADDR + 0x0000)

**#define** TIM8\_BASEADDR (APB2\_BASEADDR + 0x0400)

**#define** USART1\_BASEADDR (APB2\_BASEADDR + 0x1000)

**#define** USART6\_BASEADDR (APB2\_BASEADDR + 0x1400)

**#define** ADC\_BASEADDR (APB2\_BASEADDR + 0x2000)

**#define** SDIO\_BASEADDR (APB2\_BASEADDR + 0x2C00)

**#define** SPI1\_BASEADDR (APB2\_BASEADDR + 0x3000)

**#define** SPI4\_BASEADDR (APB2\_BASEADDR + 0x3400)

**#define** SYSCFG\_BASEADDR (APB2\_BASEADDR + 0x3800)

**#define** EXTI\_BASEADDR (APB2\_BASEADDR + 0x3C00)

**#define** TIM9\_BASEADDR (APB2\_BASEADDR + 0x4000)

**#define** TIM10\_BASEADDR (APB2\_BASEADDR + 0x4400)

**#define** TIM11\_BASEADDR (APB2\_BASEADDR + 0x4800)

**#define** SPI5\_BASEADDR (APB2\_BASEADDR + 0x5000)

**#define** SPI6\_BASEADDR (APB2\_BASEADDR + 0x5400)

**#define** SAI1\_BASEADDR (APB2\_BASEADDR + 0x5800)

**#define** LCD\_TFT\_BASEADDR (APB2\_BASEADDR + 0x6800)

//GPIO register definition

**typedef** **struct**

{

**volatile** uint32\_t MODER;//0+00 offset

**volatile** uint32\_t OTYPER;//0+04 offset

**volatile** uint32\_t SPEEDR;//0+08 offset

**volatile** uint32\_t PUPDR;//0+0C

**volatile** uint32\_t IDR;//0+10

**volatile** uint32\_t ODR;//0+14

**volatile** uint32\_t BSRR;//0+18

**volatile** uint32\_t LCKR;//0+1C

**volatile** uint32\_t AFR[2];//0+20

}GPIO\_RegDef;

**typedef** **struct**

{

**volatile** uint32\_t CR;//0x00

**volatile** uint32\_t PLLCFGR;//0x04

**volatile** uint32\_t CFGR;//0x08

**volatile** uint32\_t CIR;//0x0C

**volatile** uint32\_t AHB1RSTR;//0X10

**volatile** uint32\_t AHB2RSTR;

**volatile** uint32\_t APB1RSTR;

**volatile** uint32\_t APB2RSTR;

**volatile** uint32\_t AHB1ENR;

**volatile** uint32\_t AHB2ENR;

**volatile** uint32\_t APB1ENR;

**volatile** uint32\_t APB2ENR;

**volatile** uint32\_t AHB1LPENR;

**volatile** uint32\_t AHB2LPENR;

**volatile** uint32\_t APB1LPENR;

**volatile** uint32\_t APB2LPENR;

**volatile** uint32\_t BDCR;

**volatile** uint32\_t CSR;

**volatile** uint32\_t SSCGR;

**volatile** uint32\_t PLLI2SCFGR;

}RCC\_RegDef;

**#define** RCC ((RCC\_RegDef\*)RCC\_BASEADDR)

//MACROS FOR GPIO POINTER STRUCTURE

**#define** GPIOA (GPIO\_RegDef\*) GPIOA\_BASEADDR

**#define** GPIOB (GPIO\_RegDef\*) GPIOB\_BASEADDR

**#define** GPIOC (GPIO\_RegDef\*) GPIOC\_BASEADDR

**#define** GPIOD (GPIO\_RegDef\*) GPIOD\_BASEADDR

**#define** GPIOE (GPIO\_RegDef\*) GPIOE\_BASEADDR

**#define** GPIOF (GPIO\_RegDef\*) GPIOF\_BASEADDR

**#define** GPIOG (GPIO\_RegDef\*) GPIOG\_BASEADDR

**#define** GPIOH (GPIO\_RegDef\*) GPIOH\_BASEADDR

//MACROS FOR ENABLING CLOCK FOR GPIO'S

**#define** GPIOA\_PCLOCKENABLE (RCC\_RegDef\*) (RCC->AHB1ENR=(1<<0))

**#define** GPIOB\_PCLOCKENABLE (RCC\_RegDef\*) (RCC->AHB1ENR=(1<<1))

**#define** GPIOC\_PCLOCKENABLE (RCC\_RegDef\*) (RCC->AHB1ENR=(1<<2))

**#define** GPIOD\_PCLOCKENABLE (RCC\_RegDef\*) (RCC->AHB1ENR=(1<<3))

**#define** GPIOE\_PCLOCKENABLE (RCC\_RegDef\*) (RCC->AHB1ENR=(1<<4))

**#define** GPIOF\_PCLOCKENABLE (RCC\_RegDef\*) (RCC->AHB1ENR=(1<<5))

**#define** GPIOG\_PCLOCKENABLE (RCC\_RegDef\*) (RCC->AHB1ENR=(1<<6))

**#define** GPIOH\_PCLOCKENABLE (RCC\_RegDef\*) (RCC->AHB1ENR=(1<<7))

**#define** GPIOI\_PCLOCKENABLE (RCC\_RegDef\*) (RCC->AHB1ENR=(1<<8))

//MACROS for disabling CLOCK

**#define** GPIOA\_PCLOCKDISABLE (RCC\_RegDef\*) (RCC->AHB1ENR=~(1<<0))

**#define** GPIOB\_PCLOCKDISABLE (RCC\_RegDef\*) (RCC->AHB1ENR=~(1<<1))

**#define** GPIOC\_PCLOCKDISABLE (RCC\_RegDef\*) (RCC->AHB1ENR=~(1<<2))

**#define** GPIOD\_PCLOCKDISABLE (RCC\_RegDef\*) (RCC->AHB1ENR=~(1<<3))

**#define** GPIOE\_PCLOCKDISABLE (RCC\_RegDef\*) (RCC->AHB1ENR=~(1<<4))

**#define** GPIOF\_PCLOCKDISABLE (RCC\_RegDef\*) (RCC->AHB1ENR=~(1<<5))

**#define** GPIOG\_PCLOCKDISABLE (RCC\_RegDef\*) (RCC->AHB1ENR=~(1<<6))

**#define** GPIOH\_PCLOCKDISABLE (RCC\_RegDef\*) (RCC->AHB1ENR=~(1<<7))

**#define** GPIOI\_PCLOCKDISABLE (RCC\_RegDef\*) (RCC->AHB1ENR=~(1<<8))

**#endif** /\* DRIVERS\_INC\_STM324XX\_H\_ \*/