|  |  |  |  |
| --- | --- | --- | --- |
| 姓名：陈文辉 | 专业：物联网工程 | 班级：20181191 | 学号：2018272102 |
| 科目：单片机原理及应用 | | 实验日期：2020/12/28 | |
| 实验题目：LED闪烁 | | | |
| 【实验目的】  1. LED实现闪烁效果 | | | |
| 【实验内容】  1.实验设备   硬件：PC机一台   小熊派开发板一套   软件：XP/win7、8、8.1、10 系统，IDE 工具 [MDK-ARM](http://www2.keil.com/mdk5/)，配置工具[ST 的 CubeMX](https://www.st.com/content/st_com/en/products/development-tools/software-development-tools/stm32-software-development-tools/stm32-configurators-and-code-generators/stm32cubemx.html)。git 客户端 [sourcetree](https://www.sourcetreeapp.com/)，课程使用 sourcetree 和 [github](http://www.github.com/) 作为项目管理和协作系统。  2.实验内容和要求  控制LED实现闪烁效果 | | | |
| 【实验结果】  1.构思 选择芯片型号配置时钟源配置LED的GPIO引脚配置时钟树  * 生成工程  1. 代码实现   main.c  /\* USER CODE BEGIN Header \*/  /\*\*  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* @file : main.c  \* @brief : Main program body  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* @attention  \*  \* <h2><center>&copy; Copyright (c) 2020 STMicroelectronics.  \* All rights reserved.</center></h2>  \*  \* This software component is licensed by ST under BSD 3-Clause license,  \* the "License"; You may not use this file except in compliance with the  \* License. You may obtain a copy of the License at:  \* opensource.org/licenses/BSD-3-Clause  \*  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \*/  /\* USER CODE END Header \*/  /\* Includes ------------------------------------------------------------------\*/  #include "main.h"  #include "gpio.h"  /\* Private includes ----------------------------------------------------------\*/  /\* USER CODE BEGIN Includes \*/  /\* USER CODE END Includes \*/  /\* Private typedef -----------------------------------------------------------\*/  /\* USER CODE BEGIN PTD \*/  /\* USER CODE END PTD \*/  /\* Private define ------------------------------------------------------------\*/  /\* USER CODE BEGIN PD \*/  /\* USER CODE END PD \*/  /\* Private macro -------------------------------------------------------------\*/  /\* USER CODE BEGIN PM \*/  /\* USER CODE END PM \*/  /\* Private variables ---------------------------------------------------------\*/  /\* USER CODE BEGIN PV \*/  /\* USER CODE END PV \*/  /\* Private function prototypes -----------------------------------------------\*/  void SystemClock\_Config(void);  /\* USER CODE BEGIN PFP \*/  /\* USER CODE END PFP \*/  /\* Private user code ---------------------------------------------------------\*/  /\* USER CODE BEGIN 0 \*/  /\* USER CODE END 0 \*/  /\*\*  \* @brief The application entry point.  \* @retval int  \*/  int main(void)  {  /\* USER CODE BEGIN 1 \*/  /\* USER CODE END 1 \*/  /\* MCU Configuration--------------------------------------------------------\*/  /\* Reset of all peripherals, Initializes the Flash interface and the Systick. \*/  HAL\_Init();  /\* USER CODE BEGIN Init \*/  /\* USER CODE END Init \*/  /\* Configure the system clock \*/  SystemClock\_Config();  /\* USER CODE BEGIN SysInit \*/  /\* USER CODE END SysInit \*/  /\* Initialize all configured peripherals \*/  MX\_GPIO\_Init();  /\* USER CODE BEGIN 2 \*/  /\* USER CODE END 2 \*/  /\* Infinite loop \*/  /\* USER CODE BEGIN WHILE \*/  while (1)  {  /\* USER CODE END WHILE \*/  /\* USER CODE BEGIN 3 \*/  HAL\_Delay(200);  HAL\_GPIO\_TogglePin(GPIOC, GPIO\_PIN\_13);  }  /\* USER CODE END 3 \*/  }  /\*\*  \* @brief System Clock Configuration  \* @retval None  \*/  void SystemClock\_Config(void)  {  RCC\_OscInitTypeDef RCC\_OscInitStruct = {0};  RCC\_ClkInitTypeDef RCC\_ClkInitStruct = {0};  /\*\* Initializes the RCC Oscillators according to the specified parameters  \* in the RCC\_OscInitTypeDef structure.  \*/  RCC\_OscInitStruct.OscillatorType = RCC\_OSCILLATORTYPE\_HSE;  RCC\_OscInitStruct.HSEState = RCC\_HSE\_ON;  RCC\_OscInitStruct.PLL.PLLState = RCC\_PLL\_ON;  RCC\_OscInitStruct.PLL.PLLSource = RCC\_PLLSOURCE\_HSE;  RCC\_OscInitStruct.PLL.PLLM = 1;  RCC\_OscInitStruct.PLL.PLLN = 20;  RCC\_OscInitStruct.PLL.PLLP = RCC\_PLLP\_DIV7;  RCC\_OscInitStruct.PLL.PLLQ = RCC\_PLLQ\_DIV2;  RCC\_OscInitStruct.PLL.PLLR = RCC\_PLLR\_DIV2;  if (HAL\_RCC\_OscConfig(&RCC\_OscInitStruct) != HAL\_OK)  {  Error\_Handler();  }  /\*\* Initializes the CPU, AHB and APB buses clocks  \*/  RCC\_ClkInitStruct.ClockType = RCC\_CLOCKTYPE\_HCLK|RCC\_CLOCKTYPE\_SYSCLK  |RCC\_CLOCKTYPE\_PCLK1|RCC\_CLOCKTYPE\_PCLK2;  RCC\_ClkInitStruct.SYSCLKSource = RCC\_SYSCLKSOURCE\_PLLCLK;  RCC\_ClkInitStruct.AHBCLKDivider = RCC\_SYSCLK\_DIV1;  RCC\_ClkInitStruct.APB1CLKDivider = RCC\_HCLK\_DIV1;  RCC\_ClkInitStruct.APB2CLKDivider = RCC\_HCLK\_DIV1;  if (HAL\_RCC\_ClockConfig(&RCC\_ClkInitStruct, FLASH\_LATENCY\_4) != HAL\_OK)  {  Error\_Handler();  }  /\*\* Configure the main internal regulator output voltage  \*/  if (HAL\_PWREx\_ControlVoltageScaling(PWR\_REGULATOR\_VOLTAGE\_SCALE1) != HAL\_OK)  {  Error\_Handler();  }  }  /\* USER CODE BEGIN 4 \*/  /\* USER CODE END 4 \*/  /\*\*  \* @brief This function is executed in case of error occurrence.  \* @retval None  \*/  void Error\_Handler(void)  {  /\* USER CODE BEGIN Error\_Handler\_Debug \*/  /\* User can add his own implementation to report the HAL error return state \*/  /\* USER CODE END Error\_Handler\_Debug \*/  }  #ifdef USE\_FULL\_ASSERT  /\*\*  \* @brief Reports the name of the source file and the source line number  \* where the assert\_param error has occurred.  \* @param file: pointer to the source file name  \* @param line: assert\_param error line source number  \* @retval None  \*/  void assert\_failed(uint8\_t \*file, uint32\_t line)  {  /\* USER CODE BEGIN 6 \*/  /\* User can add his own implementation to report the file name and line number,  tex: printf("Wrong parameters value: file %s on line %d\r\n", file, line) \*/  /\* USER CODE END 6 \*/  }  #endif /\* USE\_FULL\_ASSERT \*/  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* (C) COPYRIGHT STMicroelectronics \*\*\*\*\*END OF FILE\*\*\*\*/   1. 运行 | | | |
| **【**教师评语和成绩**】**  **成绩：** **指导教师：** **日期：** | | | |