|  |  |  |  |
| --- | --- | --- | --- |
| 姓名：欧阳芳霞 | 专业：物联网工程 | 班级：20181191 | 学号：2018119129 |
| 科目：嵌入式系统概论 | | 实验日期：2020.1.1 | |
| 实验题目：串口输出PC-UART | | | |
| 【实验目的】   串口输出PC-UART | | | |
| 【实验内容】  主要代码如下：  main.c  #include "main.h"  #include "usart.h"  #include "gpio.h"  void SystemClock\_Config(void); //系统时钟  int main(void)  {  char str[12]="Hello World\n";  char recv\_buf[12]={0};  HAL\_Init(); //HAL初始化  SystemClock\_Config(); //系统时钟初始化  MX\_GPIO\_Init(); //GPIO初始化  MX\_USART1\_UART\_Init(); //UART初始化  HAL\_UART\_Transmit(&huart1,(uint8\_t\*)str,12,0xFFFF);  while (1)  {  if(HAL\_OK == HAL\_UART\_Receive(&huart1,(uint8\_t\*)recv\_buf,12,0xFFFF))  {  HAL\_UART\_Transmit(&huart1,(uint8\_t\*)recv\_buf,12,0xFFFF);  }  }  }  void SystemClock\_Config(void) //系统时钟初始化  {  RCC\_OscInitTypeDef RCC\_OscInitStruct = {0};  RCC\_ClkInitTypeDef RCC\_ClkInitStruct = {0};  RCC\_PeriphCLKInitTypeDef PeriphClkInit = {0};  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();  }    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();  }  PeriphClkInit.PeriphClockSelection = RCC\_PERIPHCLK\_USART1;  PeriphClkInit.Usart1ClockSelection = RCC\_USART1CLKSOURCE\_PCLK2;  if (HAL\_RCCEx\_PeriphCLKConfig(&PeriphClkInit) != HAL\_OK)  {  Error\_Handler();  }  if (HAL\_PWREx\_ControlVoltageScaling(PWR\_REGULATOR\_VOLTAGE\_SCALE1) != HAL\_OK)  {  Error\_Handler();  }  }  #ifdef USE\_FULL\_ASSERT  #endif /\* USE\_FULL\_ASSERT \*/  gpio.c  #include "gpio.h"  void MX\_GPIO\_Init(void) //GPTO初始化  {  \_\_HAL\_RCC\_GPIOC\_CLK\_ENABLE();  \_\_HAL\_RCC\_GPIOH\_CLK\_ENABLE();  \_\_HAL\_RCC\_GPIOA\_CLK\_ENABLE();  }  usart.c  #include "usart.h"  UART\_HandleTypeDef huart1;  void MX\_USART1\_UART\_Init(void) //UART初始化  {  huart1.Instance = USART1;  huart1.Init.BaudRate = 115200;  huart1.Init.WordLength = UART\_WORDLENGTH\_8B;  huart1.Init.StopBits = UART\_STOPBITS\_1;  huart1.Init.Parity = UART\_PARITY\_NONE;  huart1.Init.Mode = UART\_MODE\_TX\_RX;  huart1.Init.HwFlowCtl = UART\_HWCONTROL\_NONE;  huart1.Init.OverSampling = UART\_OVERSAMPLING\_16;  huart1.Init.OneBitSampling = UART\_ONE\_BIT\_SAMPLE\_DISABLE;  huart1.AdvancedInit.AdvFeatureInit = UART\_ADVFEATURE\_NO\_INIT;  if (HAL\_UART\_Init(&huart1) != HAL\_OK)  {  Error\_Handler();  }  }  void HAL\_UART\_MspInit(UART\_HandleTypeDef\* uartHandle)  {  GPIO\_InitTypeDef GPIO\_InitStruct = {0};  if(uartHandle->Instance==USART1)  {  \_\_HAL\_RCC\_USART1\_CLK\_ENABLE();  \_\_HAL\_RCC\_GPIOA\_CLK\_ENABLE();  GPIO\_InitStruct.Pin = GPIO\_PIN\_9|GPIO\_PIN\_10;  GPIO\_InitStruct.Mode = GPIO\_MODE\_AF\_PP;  GPIO\_InitStruct.Pull = GPIO\_NOPULL;  GPIO\_InitStruct.Speed = GPIO\_SPEED\_FREQ\_VERY\_HIGH;  GPIO\_InitStruct.Alternate = GPIO\_AF7\_USART1;  HAL\_GPIO\_Init(GPIOA, &GPIO\_InitStruct);  }  }  void HAL\_UART\_MspDeInit(UART\_HandleTypeDef\* uartHandle)  {  if(uartHandle->Instance==USART1)  {  \_\_HAL\_RCC\_USART1\_CLK\_DISABLE();  HAL\_GPIO\_DeInit(GPIOA, GPIO\_PIN\_9|GPIO\_PIN\_10);  }  } | | | |
| 【实验结果】  运行时串口发送接收数据，显示Hello World并换行  【实验结论】（写结论，心得，和收获）  实验结论、心得和收获：  ①掌握了串口发送接收数据，并显示原理  ②基本了解了实验的操作过程 | | | |
| **【**教师评语和成绩**】**  **成绩：** **指导教师：** **日期：** | | | |