Lesson 13 Add APP Control

1. Project Purpose

Use Bluetooth module to implement the communicate with the controller.

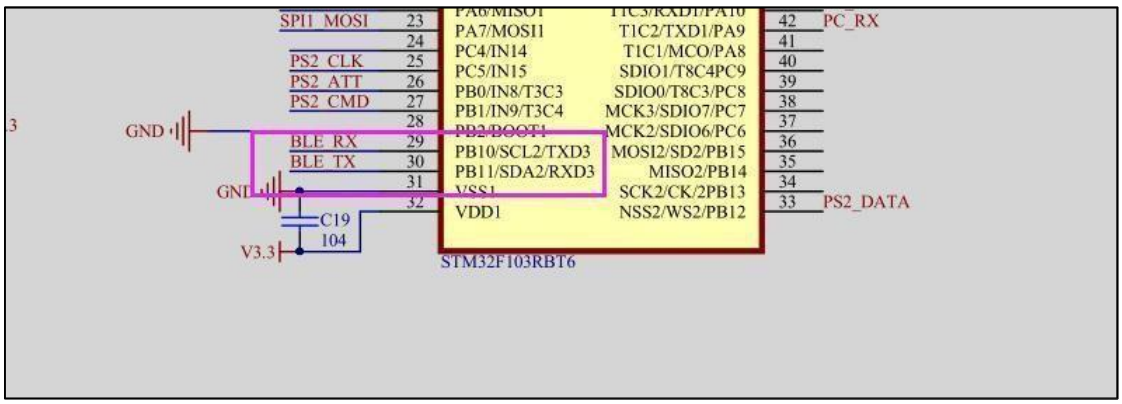
2. Project Principle

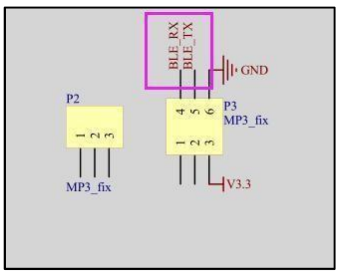
This section will add APP control function to uHand2.0.

The Bluetooth module will be connected to the controller of the palm through the serial port. The serial port used is the same as the communication serial port of the uHand2.0 PC software, so the same protocol is used for Bluetooth communication and uHand2.0 PC software communication. Therefore, we just need to copy a communication protocol, and then adapt it to the serial port communicating with the Bluetooth module.

3. Program Analyst

1) By viewing the schematic diagram, the Bluetooth module is connected to the UART3 serial port of STM32 so the serial port needs to be initialized.





```
void InitUart1(void)
{
    NVIC_InitTypeDef NVIC_InitStructure;

    GPIO_InitTypeDef GPIO_InitStructure;
    USART_InitTypeDef USART_InitStructure;
//  NVIC_InitTypeDef NVIC_InitStructure;

    RCC_APB2PeriphClockCmd(RCC_APB2Periph_USART1|RCC_APB2Periph_GPIOA|RCC_APB2Periph_AFIO, ENABLE);
    //USART1_TX   PA.9
    GPIO_InitStructure.GPIO_Pin = GPIO_Pin_9;
    GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50MHz;
    GPIO_InitStructure.GPIO_Mode = GPIO_Mode_AF_PP;
    GPIO_Init(GPIOA, &GPIO_InitStructure);

    //USART1_RX   PA.10
    GPIO_InitStructure.GPIO_Pin = GPIO_Pin_10;
    GPIO_InitStructure.GPIO_Mode = GPIO_Mode_IPU;
    GPIO_Init(GPIOA, &GPIO_InitStructure);

    //USART Initialization setting

    USART_InitStructure.USART_BaudRate = 9600;//the baud rate is set to 9600;
    USART_InitStructure.USART_WordLength = USART_WordLength_8b;//8 data bits
    USART_InitStructure.USART_StopBits = USART_StopBits_1; // 1 stop bit
    USART_InitStructure.USART_Parity = USART_Parity_No; //Invalid position
    USART_InitStructure.USART_HardwareFlowControl = USART_HardwareFlowControl_None;// None hardware flow control
    USART_InitStructure.USART_Mode = USART_Mode_Rx | USART_Mode_Tx; //use send and receive

    USART_Init(USART1, &USART_InitStructure);  //Configure related registers according to parameters

    USART_ITConfig(USART1, USART_IT_RXNE, ENABLE);//enbale USART

    USART_Cmd(USART1, ENABLE);                    //enable USART


    NVIC_InitStructure.NVIC_IRQChannel = USART1_IRQn; //USART1 interrupt
    NVIC_InitStructure.NVIC_IRQChannelPreemptionPriority=1 ; //Preemption Priority 1
    NVIC_InitStructure.NVIC_IRQChannelSubPriority = 0;      // SubPriority 0
    NVIC_InitStructure.NVIC_IRQChannelCmd = ENABLE;         //IRQ channel enable
    NVIC_Init(&NVIC_InitStructure); //Initial  peripherals NVIC register according to the parameters specified in NVIC_InitStruct.
}
```

2) After initializing the serial port, Bluetooth can be used to communicate. The method of sending polling register to implement that the receiving end uses the reception interrupt.

```
void USART3_IRQHandler(void)                    //串口3中断服务程序
{
  u8 rxBuf;
  static uint8 startCodeSum = 0;
  static bool fFrameStart = FALSE;
  static uint8 messageLength = 0;
  static uint8 messageLengthSum = 2;
  if(USART_GetITStatus(USART3, USART_IT_RXNE) != RESET)  //接收中断(接收到的数据必须是0x0d 0x0a结尾)
  {
    rxBuf =USART_ReceiveData(USART3);//(USART1->DR);  //读取接收到的数据

    if(!fFrameStart)
    {
      if(rxBuf == 0x55)
      {
        startCodeSum++;
        if(startCodeSum == 2)
        {
          startCodeSum = 0;
          fFrameStart = TRUE;
          messageLength = 1;
        }
      }
      else
      {
        fFrameStart = FALSE;
        messageLength = 0;

        startCodeSum = 0;
      }
    }
    if(fFrameStart)
    {
      UartRxBuffer[messageLength] = rxBuf;
      if(messageLength == 2)
      {
        messageLengthSum = UartRxBuffer[messageLength];
        if(messageLengthSum < 2)// || messageLengthSum > 30
        {
          messageLengthSum = 2;
          fFrameStart = FALSE;
        }
      }
      messageLength++;

      if(messageLength == messageLengthSum + 2)
      {
        fUartRxComplete = TRUE;

        fFrameStart = FALSE;
      }
    }
  }
}
```

3) After the reception is completed, process with TaskBLEMsgHandle. TaskBLEMsgHandle will be called in main function, and then perform the corresponding operation based on the received commands.

```
191  void TaskPCMsgHandle(void)
192  {
193
194      uint16 i;
195      uint8 cmd;
196      uint8 id;
197      uint8 servoCount;
198      uint16 time;
199      uint16 pos;
200      uint16 times;
201      uint8 fullActNum;
202      if(UartRxOK())
203      {
204          LED = !LED;
205          cmd = UartRxBuffer[3];
206          switch(cmd)
207          {
208              case CMD_MULT_SERVO_MOVE:
209                  servoCount = UartRxBuffer[4];
210                  time = UartRxBuffer[5] + (UartRxBuffer[6]<<8);
211                  for(i = 0; i < servoCount; i++)
212                  {
213                      id =  UartRxBuffer[7 + i * 3];
214                      pos = UartRxBuffer[8 + i * 3] + (UartRxBuffer[9 + i * 3]<<8);
215
216                      ServoSetPluseAndTime(id,pos,time);
217                      BusServoCtrl(id,SERVO_MOVE_TIME_WRITE,pos,time);
218                  }
219                  break;
220
221              case CMD_FULL_ACTION_RUN:
222                  fullActNum = UartRxBuffer[4];//Action group number
223                  times = UartRxBuffer[5] + (UartRxBuffer[6]<<8);//running times
224                  McuToPCSendData(CMD_FULL_ACTION_RUN, 0, 0);
225                  FullActRun(fullActNum,times);
226                  break;
227
228              case CMD_FULL_ACTION_STOP:
229                  FullActStop();
230                  break;
231
232              case CMD_FULL_ACTION_ERASE:
233                  FlashEraseAll();
234                  McuToPCSendData(CMD_FULL_ACTION_ERASE,0,0);
235                  break;
236
237              case CMD_ACTION_DOWNLOAD:
238                  SaveAct(UartRxBuffer[4],UartRxBuffer[5],UartRxBuffer[6],UartRxBuffer + 7);
239                  McuToPCSendData(CMD_ACTION_DOWNLOAD,0,0);
240                  break;
241          }
242      }
243  }
```