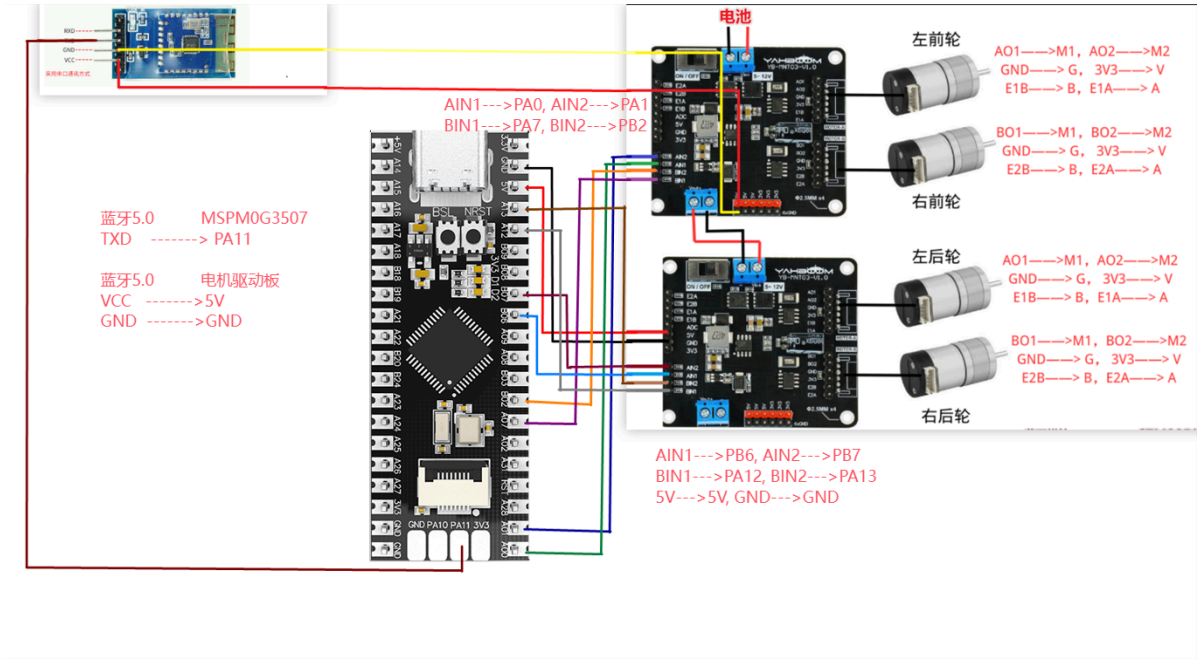
蓝牙控制

一、学习目标

使用yahboomrobotAPP通过蓝牙5.0模块控制小车运动状态

二、硬件连接

蓝牙5.0模块与MSPM0G3507与双路电机驱动板的接线。电机使用的是310电机。



下层驱动板(左前轮、右前轮)	MSPM0G3507		上层驱动板(左后轮、右后轮)	MSPM0G3507
AIN1	PA0		AIN1	PB6
AIN2	PA1		AIN2	PB7
BIN1	PA7		BIN1	PA12
BIN2	PB2		BIN2	PA13
			5V	5V
			GND	GND
蓝牙5.0	**MSPM0G3507**		**蓝牙5.0**	**电机驱动板**
TXD	PA11		VCC	5V
			GND	GND

三、程序说明

- usart.c

```
void USART_Init(void)
{
    // SYSCFG初始化
    // SYSCFG initialization
    SYSCFG_DL_init();
    //清除串口中断标志
    //Clear the serial port interrupt flag
    NVIC_ClearPendingIRQ(UART_0_INST_INT_IRQN);
    //使能串口中断
    //Enable serial port interrupt
    NVIC_EnableIRQ(UART_0_INST_INT_IRQN);
}

//串口的中断服务函数
//Serial port interrupt service function
void UART_0_INST_IRQHandler(void)
{
    uint8_t receivedData = 0;
    static uint8_t rec_state = 0;

    //如果产生了串口中断
    //If a serial port interrupt occurs
    switch(DL_UART_getPendingInterrupt(UART_0_INST) )
    {
        case DL_UART_IIDX_RX://如果是接收中断  If it is a receive interrupt

            // 接收发送过来的数据保存  Receive and save the data sent
            receivedData = DL_UART_Main_receiveData(UART_0_INST);
            switch(rec_state)
            {
                case 0:
                    if((receivedData == '$') && (!recv0_flag))
                    {
                        rec_state = 1;
                        recv0_length = 0;
                    }
                    else
                    {
                        rec_state = 0;
                    }
                    break;
                case 1:
                    if(receivedData == '#')
                    {
                        recv0_flag = 1;
                        rec_state = 0;
                    }
                    else
                    {
                        recv0_buff[recv0_length++] = receivedData;
                    }
```

```
                    break;
            }
            default://其他的串口中断    Other serial port interrupts
            break;
    }
}
```

- USART_Init：系统与串口的初始化函数。
- UART_0_INST_IRQHandler：串口的接收中断函数，用于接收从蓝牙模块发送来的数据。

- motor.h

```
#define   PWMA_IN1(value)
DL_TimerG_setCaptureCompareValue(PWM_0_INST,value,GPIO_PWM_0_C0_IDX);
#define   PWMA_IN2(value)
DL_TimerG_setCaptureCompareValue(PWM_0_INST,value,GPIO_PWM_0_C1_IDX);
#define   PWMB_IN1(value)
DL_TimerG_setCaptureCompareValue(PWM_0_INST,value,GPIO_PWM_0_C2_IDX);
#define   PWMB_IN2(value)
DL_TimerG_setCaptureCompareValue(PWM_0_INST,value,GPIO_PWM_0_C3_IDX);

#define   PWMC_IN1(value)
DL_TimerG_setCaptureCompareValue(PWM_1_INST,value,GPIO_PWM_1_C0_IDX);
#define   PWMC_IN2(value)
DL_TimerG_setCaptureCompareValue(PWM_1_INST,value,GPIO_PWM_1_C1_IDX);
#define   PWMD_IN1(value)
DL_TimerG_setCaptureCompareValue(PWM_2_INST,value,GPIO_PWM_2_C0_IDX);
#define   PWMD_IN2(value)
DL_TimerG_setCaptureCompareValue(PWM_2_INST,value,GPIO_PWM_2_C1_IDX);
```

定义4个电机的pwm占空比设置函数。

- Motor.c

```
void Data_Analyse(void)//解析串口中断串口接收的数据  Parse the data received by the
serial port interrupt
{
        if(recv0_flag == 1)
        {
            switch(recv0_buff[0])
            {
                case '1':
                    printf("Forward!\n");
                    Car_state = 1;
                    break;
                case '2':
                    printf("Backward!\n");
                    Car_state = 2;
                    break;
                case '3':
                    printf("Left!\n");
```

```
                    Car_state = 3;
                    break;
                case '4':
                    printf("Right!\n");
                    Car_state = 4;
                    break;
                case '0':
                    printf("Stop!\n");
                    Car_state = 0;
                    break;
            }
            switch(recv0_buff[2])
            {
                case '1':
                    printf("SpinLeft!\n");
                    Car_state = 5;
                    break;
                case '2':
                    printf("SpinRight!\n");
                    Car_state = 6;
                    break;
            }
            recv0_flag = 0;
        }
}

void Car_Function(unsigned int Car_state)//控制小车不同状态 Control the car in
different states
{
    switch(Car_state)
    {
            case 0:
                    printf("Stop\n");
                    Stop();
                    break;
            case 1:
                    printf("Forward\n");
                    Forward(2300);
                    break;
            case 2:
                    printf("Backward\n");
                    Backward(2300);
                    break;
            case 3:
                    printf("Turnleft\n");
                    Turnleft(2500);
                    break;
            case 4:
                    printf("Turnright\n");
                    Turnright(2500);
                    break;
            case 5:
                    printf("SpinLeft\n");
                    SpinLeft(2500);
                    break;
            case 6:
```

```
                    printf("SpinRight\n");
                    SpinRight(2500);
                    break;
    }
}
```

- Data_Analyse：解析从串口接收来的数据，根据通信协议返回小车的控制状态值。
- Car_Function：根据小车的控制状态值，驱动四个电机。通过Forward、Backward函数实现前进后退；Turnleft、Turnright函数实现左转右转；SpinLeft、SpinRight函数实现左旋右旋；以及Stop函数停止；可以修改其中的数值改变速度。

  如果使用的是本教程提及的电机驱动板与电机，小车的方向运行不对的话，则需要重复检查接线。如果使用的是非本教程提及的电机驱动板及电机，则需要根据自己的情况自行修改各轮胎的正反转。

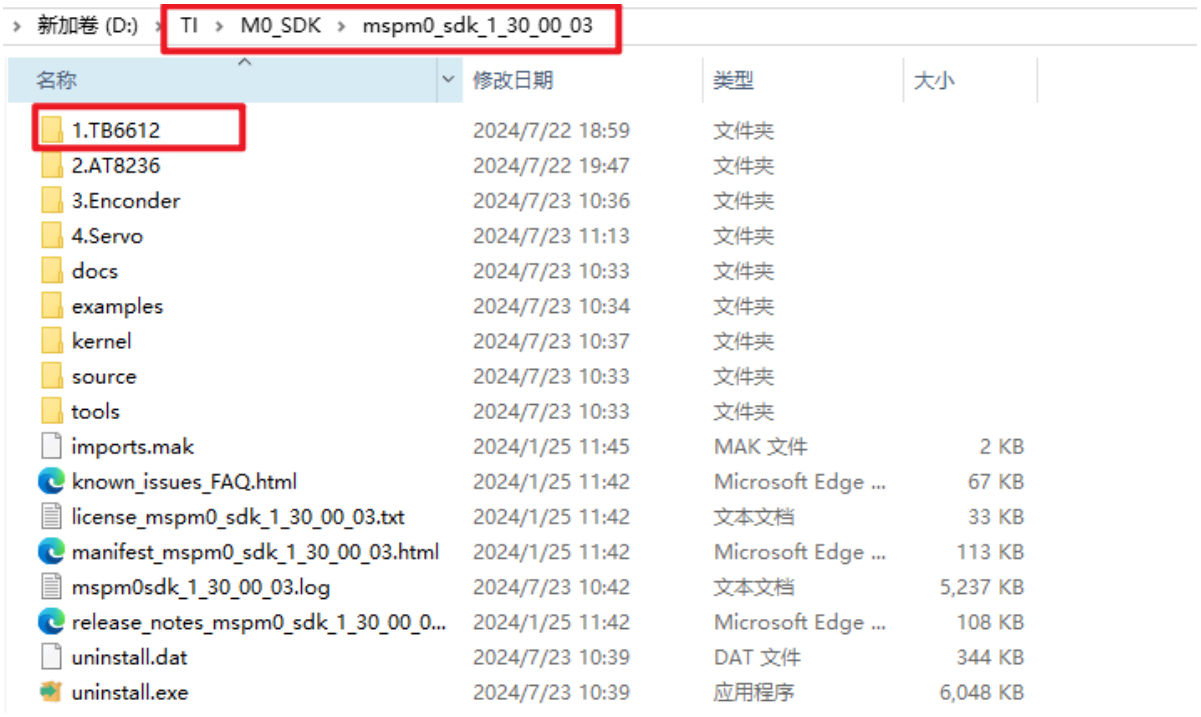
- empty.c

```
int main(void)
{
    USART_Init();
    while (1)
    {
        Data_Analyse();//解析串口中断接收的数据     Parsing data received by serial port interrupt
        Car_Function(Car_state);//控制小车不同状态  Control the car in different states
    }
}
```

程序运行后会一直处于接收数据的状态，接收数据后会进行处理并控制小车状态。

注：必须将工程源码放在SDK路径下进行编译，

例如路径：D:\TI\M0_SDK\mspm0_sdk_1_30_00_03\1.TB6612



四、实验现象

给MSPM0G3507烧录蓝牙控制程序。按照接线图耐心连接好接线，并认真检查。如果连接线接错轻则小车控制不动，重则烧坏板子。确认无误后，打开上下驱动板开关。打开开关后，打开yahboomrobot APP。选择4wd车型，连接蓝牙后就可以通过APP控制小车前进后退、左转右转、左旋右旋运动。

附录：

APP下载：https://www.yahboom.com/download_app

YahboomRobot APP具体操作

通信协议见同文件夹下的**亚博智能小程序通讯协议**文件。