STM32 application routines

1. Hardware preparation

Win10 computer, IMU module (data interface leaded), micro-USB data cable, CH340 module, DuPont Several lines.

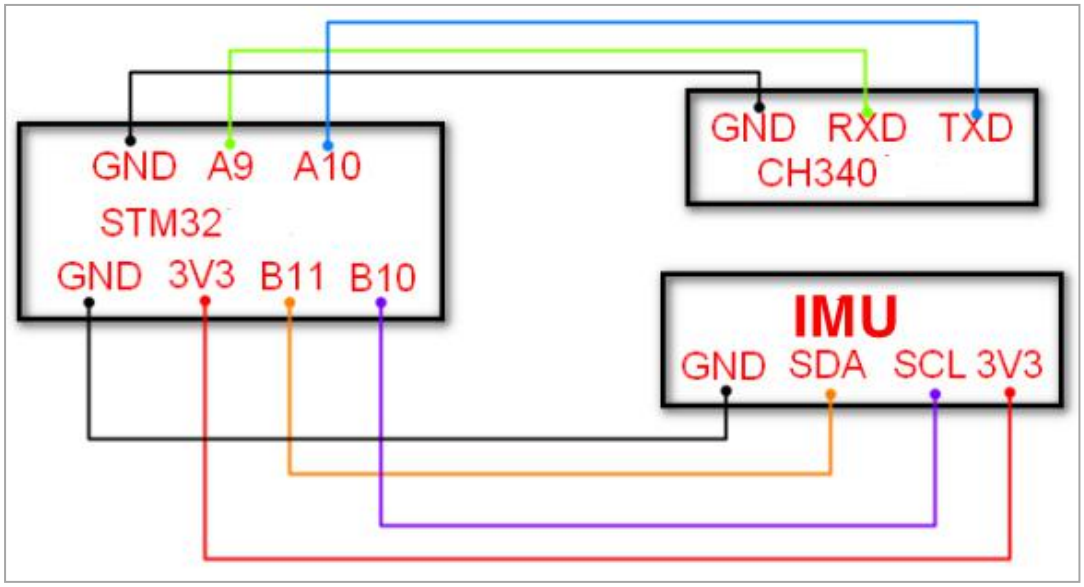
Note: The IMU module interface affects the warranty after welding, please consult customer service for details.

2. Hardware connection

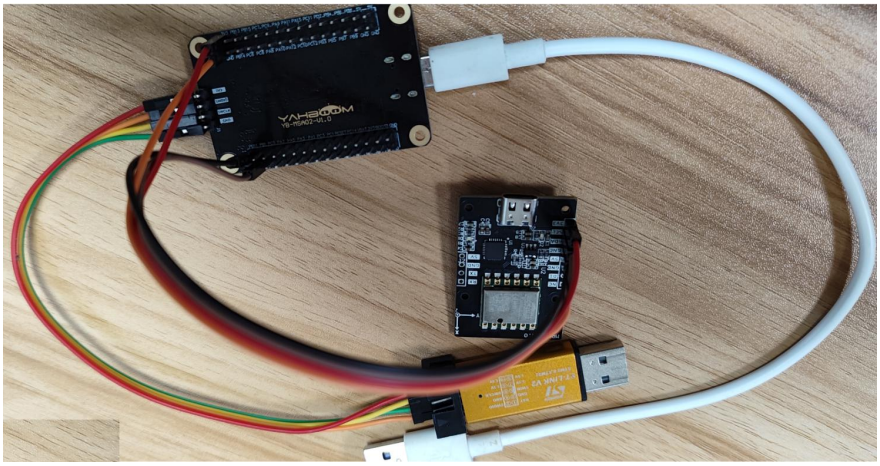
Connect STM32 serial port 1 to the CH340 module, and plug the CH340 module into the USB port of the Win10 computer.

Connect the IIC2 (PB10 PB11) of the STM32 to the IIC interface (SCL SDA) of the IMU module.

Connect the micro-USB cable to the USB port of STM32 and Win10 computers as the power supply for STM32.



If it is our STM32F103RCT6 core board, the on-board Type-C interface can replace the CH340 module, and the computer can be connected directly through this interface.



3. Software programming

For specific codes, please refer to the source code in the data.

This example sets the baud rate of serial port printing to 115200 bps and the baud rate of IMU

module connection to 9600. The baud rate of STM32 serial port 2 can be modified according to the baud rate of the IMU module.

```
Usart1Init(115200);
Usart2Init(9600);
```

Read and print data from the IMU module.

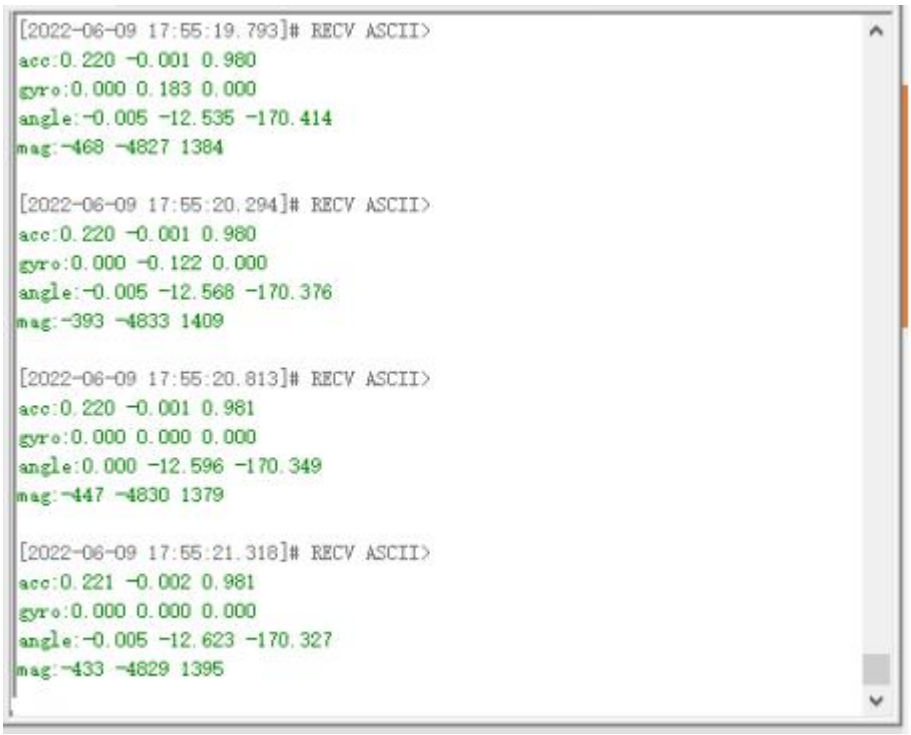
```
while (1)
{
    CmdProcess();
    if(s_cDataUpdate)
    {
        for(i = 0; i < 3; i++)
        {
            fAcc[i] = sReg[AX+i] / 32768.0f * 16.0f;
            fGyro[i] = sReg[GX+i] / 32768.0f * 2000.0f;
        }
        if(sReg[TEMP] - sReg[Roll] > 10 | sReg[Roll] - sReg[TEMP] > 10)
        {
            fAngle[0] = sReg[Roll] / 32768.0f * 180.0f;
            fAngle_tmp = fAngle[0];
        }
        else   fAngle[0] = fAngle_tmp;

        for(i = 1; i<3;i++)
        {
            fAngle[i] = sReg[Roll+i] / 32768.0f * 180.0f;
        }

        if(s_cDataUpdate & ACC_UPDATE)
        {
            printf("acc:%.3f %.3f %.3f\r\n", fAcc[0], fAcc[1], fAcc[2]);
            s_cDataUpdate &= ~ACC_UPDATE;
        }
        if(s_cDataUpdate & GYRO_UPDATE)
        {
            printf("gyro:%.3f %.3f %.3f\r\n", fGyro[0], fGyro[1], fGyro[2]);
            s_cDataUpdate &= ~GYRO_UPDATE;
        }
        if(s_cDataUpdate & ANGLE_UPDATE)
        {
            printf("angle:%.3f %.3f %.3f\r\n", fAngle[0], fAngle[1], fAngle[2]);
            s_cDataUpdate &= ~ANGLE_UPDATE;
        }
        if(s_cDataUpdate & MAG_UPDATE)
        {
            printf("mag:%d %d %d\r\n", sReg[HX], sReg[HY], sReg[HZ]);
            s_cDataUpdate &= ~MAG_UPDATE;
        }
    }
```

4. Experimental effect

After the program is downloaded into STM32, open the serial port assistant (configuration parameters as shown in the figure below), you can see that the data of the IMU module has been printed, and when we change the attitude of the IMU module, the data will change.


[2022-06-09 17:55:19.793]# RECV ASCII>
acc:0.220 -0.001 0.980
gyro:0.000 0.183 0.000
angle:-0.005 -12.535 -170.414
mag:-468 -4827 1384
[2022-06-09 17:55:20.294]# RECV ASCII>
acc:0.220 -0.001 0.980
gyro:0.000 -0.122 0.000
angle:-0.005 -12.568 -170.376
mag:-393 -4833 1409
[2022-06-09 17:55:20.813]# RECV ASCII>
acc:0.220 -0.001 0.981
gyro:0.000 0.000 0.000
angle:0.000 -12.596 -170.349
mag:-447 -4830 1379
[2022-06-09 17:55:21.318]# RECV ASCII>
acc:0.221 -0.002 0.981
gyro:0.000 0.000 0.000
angle:-0.005 -12.623 -170.327
mag:-433 -4829 1395