STM32 code

This routine uses the STM32F103C8T6 chip development board, which is abbreviated as STM32 for the convenience of demonstration.

Products that need to be prepared: Win10 computer, inertial navigation module (data interface has lead), micro-USB data cable, CH340 module, DuPont cable.

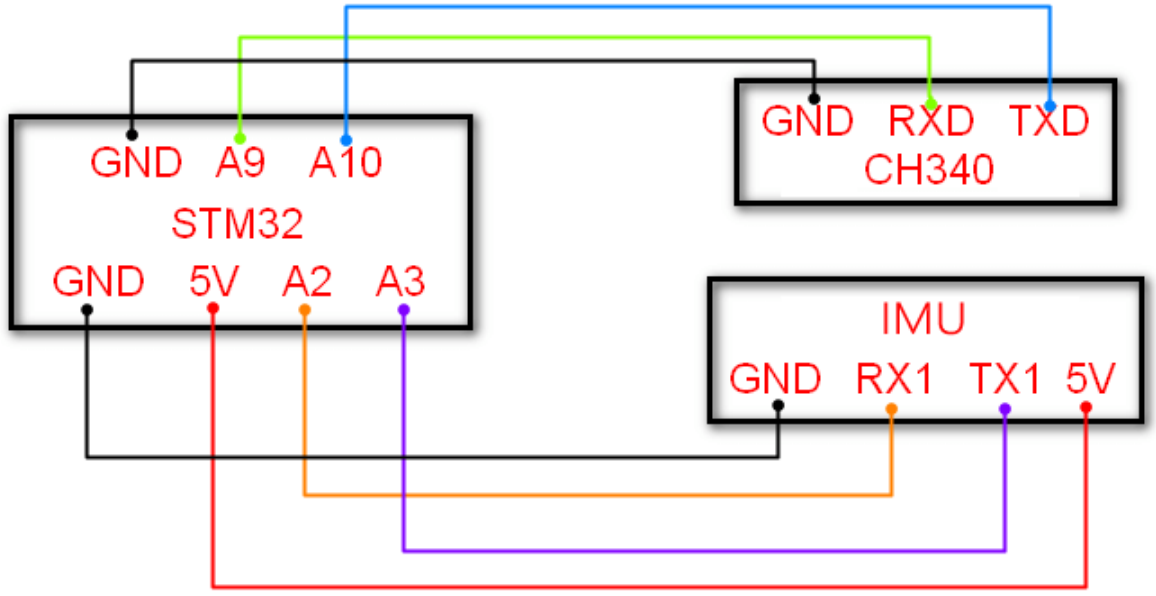
Note: After the inertial navigation module interface is welded, the warranty will be affected. Please consult customer service for details.

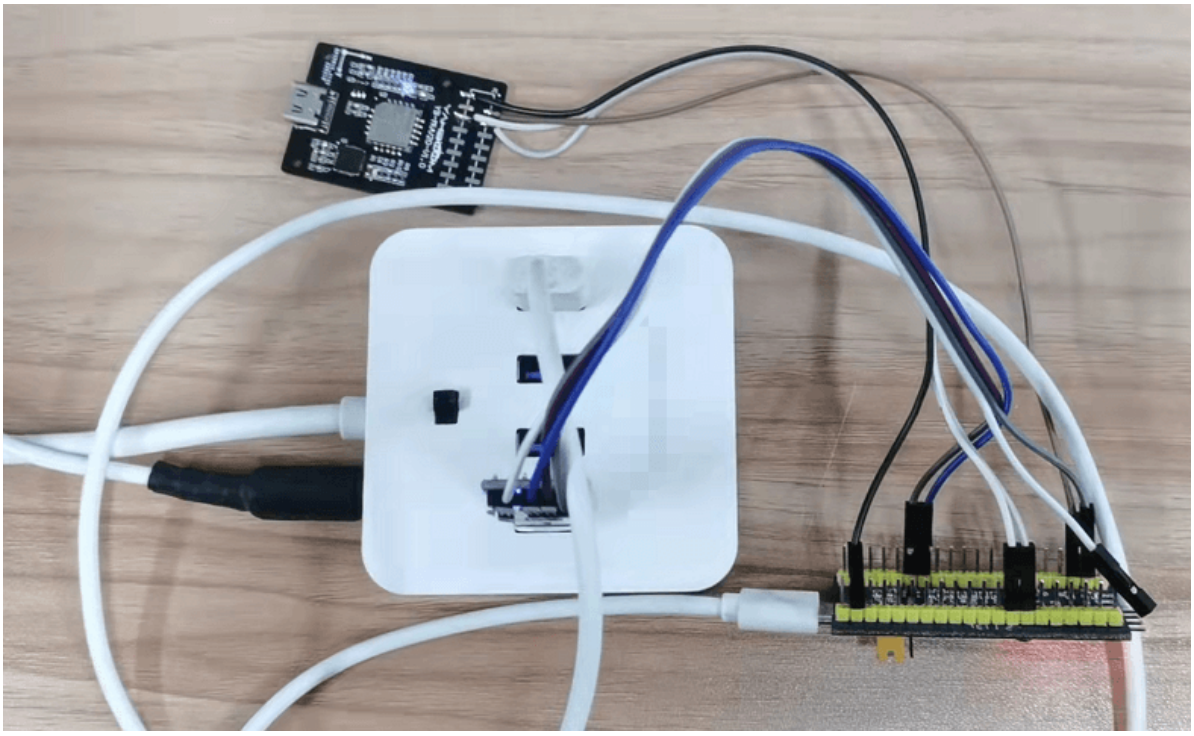
1. Hardware connection

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

Connect the STM32 serial port 2 to the data interface of the inertial navigation module.

Connect the micro-USB data cable to the USB port of the STM32 and Win10 computer to power the STM32.





2. software programming

For specific code, please see the source code in the information.

This routine has set the baud rate to 115200 bps, and the baud rate of STM32 serial port 2 can be modified according to the baud rate of the inertial navigation module.

```
/* 宏定义 definit ------------------------------------------
#define USART1_BAUDRATE                 115200
#define USART2_BAUDRATE                 115200

#define DEBUG_USARTx                    USART1
```

In order to make the printing effect clear, other parameters have been commented out. If you need to view other data, you can modify the following comments, and then save the compiler.

```
// Print AHRS attitude data  打印AHRS姿态数据
void Print_AHRS_Data(void)
{
    printf("RPY-> %f, %f, %f\r\n", AHRSData.Roll, AHRSData.Pitch, AHRSData.Heading);

    // printf("RPY Speed-> %f, %f, %f\r\n", AHRSData.RollSpeed, AHRSData.PitchSpeed,

    // printf("Quaternion-> %f, %f, %f, %f\r\n", AHRSData.Q1, AHRSData.Q2, AHRSData.

    // printf("AHRS Timestamp-> %d\r\n", AHRSData.Timestamp);
}

// Print raw data  打印原始数据
void Print_IMU_Data(void)
{
    // printf("Gyroscope-> %f, %f, %f\r\n", IMUData.Gyroscope_X, IMUData.Gyroscope_Y

    // printf("Accelerometer-> %f, %f, %f\r\n", IMUData.Accelerometer_X, IMUData.Acc

    // printf("IMU:The Magnetometer_X-> %f, %f, %f\r\n", IMUData.Magnetometer_X, IMU

    // printf("IMU Timestamp->  %d\r\n", IMUData.Timestamp);
}
```

3. Experimental effect

After the program is downloaded into STM32, open the serial port assistant (the configuration parameters are shown in the figure below), and you can see that the roll angle, pitch angle, and yaw angle data of the inertial navigation module are always printed. When we change the attitude of the inertial navigation module, the data will occur. Variety.