

GPS location information read

1. Learning Objectives

In this course, we mainly learn to use Arduino and GPS module to realize the function of reading location information.

2. Preparation before class

The GPS module uses UART and USB communication. Here, the UART port of the Arduino UNO is used to read the information, and the TX of the module is connected to the D0 pin of the Arduino UNO board. VCC and GND are connected to 5V and GND respectively.

3. Procedure

Initialize the serial port.

```
void setup() //Initialize content
{
    GPSSerial.begin(9600); //Define baud rate 9600
    DEBUGSerial.begin(9600);
}
```

Print out the received data.

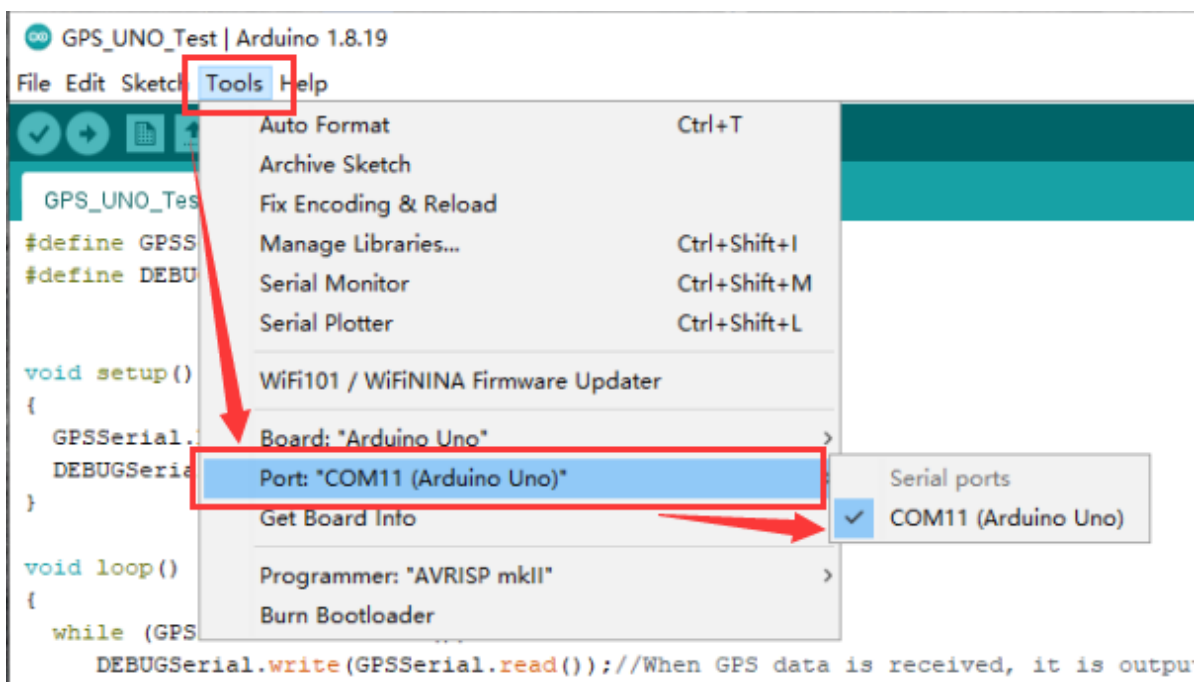
```
{
    while (GPSSerial.available()) {
        DEBUGSerial.write(GPSSerial.read()); //When GPS data is received, it is output through Serial
    }
}
```

4. Compile and download the program

4.1 We need the general Arduino IDE software to open the file, then click "√" in the menu bar to compile the program, and wait for the words "Done compiling" to appear in the lower left corner.



4.2 In the menu bar of Arduino IDE, we need to select [Tools]---[Port]---select the port number just displayed in the device manager, as shown in the figure below.



4.3 After the selection is complete, click "→" under the menu bar to upload the code to the UNO board. When the word "Done uploading" appears in the lower left corner, it means that the program has been successfully uploaded to the UNO board, as shown in the figure below.



The screenshot shows the Arduino IDE interface. At the top, the title bar reads "GPS_UNO_Test | Arduino 1.8.19". Below it is a menu bar with "File", "Edit", "Sketch", "Tools", and "Help". A toolbar contains icons for a checkmark, a right arrow, a document, and upload/download arrows. The sketch name "GPS_UNO_Test" is displayed in a teal box. The main text area contains the following C++ code:

```
#define GPSSerial Serial
#define DEBUGSerial Serial

void setup() //Initialize content
{
  GPSSerial.begin(9600); //Define baud rate 9600
  DEBUGSerial.begin(9600);
}

void loop() //main loop
{
  while (GPSSerial.available()) {
    DEBUGSerial.write(GPSSerial.read()); //When GPS data is received
  }
}
```

At the bottom, a teal status bar says "Done uploading." and a black console window shows the following output:

```
Sketch uses 1452 bytes (4%) of program storage space. Maximum is 3225
Global variables use 184 bytes (8%) of dynamic memory, leaving 1864 b
```

5. Experimental phenomenon

After the module is powered on, it takes about 32s to start, and then the serial port print status light on the module will continue to flash, and data can be received normally at this time.

After the program is downloaded and run, open the serial port monitoring window, open the serial port software, set the baud rate to 9600, and the serial port will print the current location information cyclically. This information is the original information that has not been processed. You can refer to the [CASIC multimode satellite navigation receiver protocol Specification.pdf](#) View the specific content of each message.

Note that the module antenna needs to be outdoors, otherwise the GPS signal may not be searched.