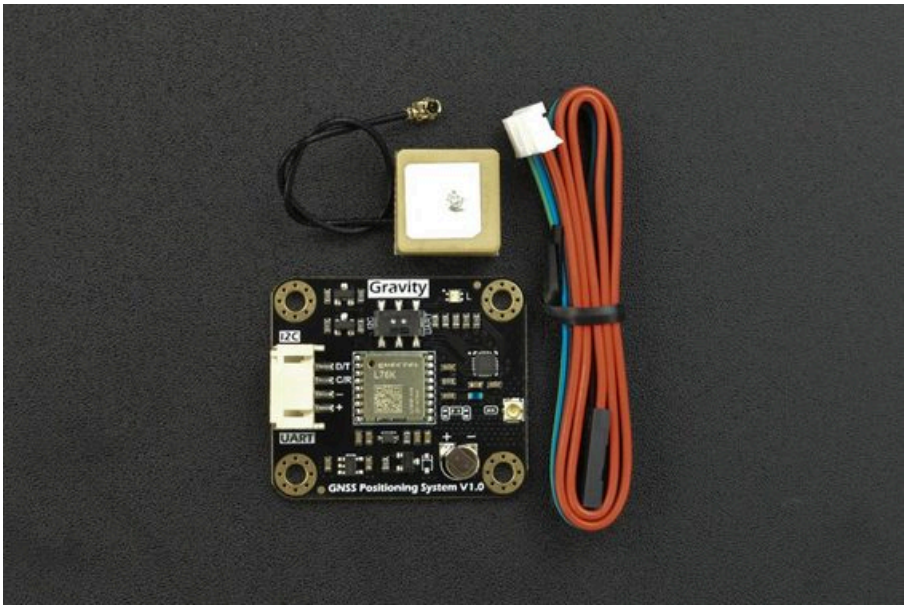
SKU:TEL0157 (https://www.dfrobot.com/product-2651.html)

(https://www.dfrobot.com/product-2651.html)



Introduction

Global Navigation Satellite Systems (GNSS) provide critical timing and positioning functions for device operations.
This Gravity: GNSS positioning module from DFRobot supports both single and multiple systems positioning. It is capable of quick delivery of position data like longitude, latitude, altitude and time. Compared with traditional single GPS positioning, the multi-system combination embraces higher precision and faster speed thanks to the increased number of visible and available satellites, which ensures stable and accurate performance even in complex urban environments.

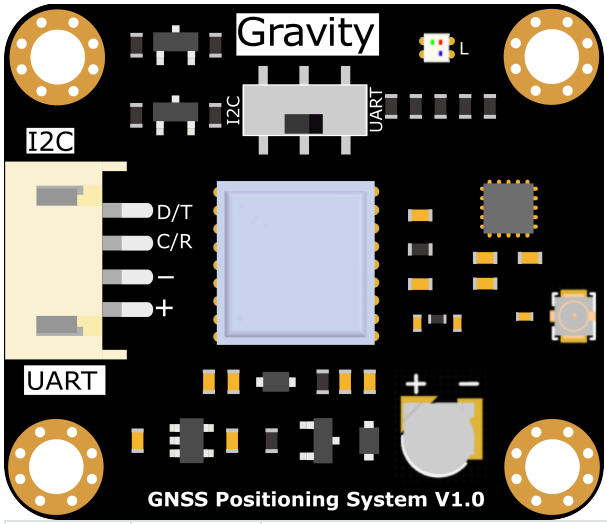
With I2C and UART data outputs, the GNSS positioning module works well with main-controllers like Arduino, ESP32, and Raspberry Pi. It is applicable to outdoor positioning scenarios such as vehicle navigation, handheld positioning tracker, item tracking and weather station.

Specification

- Operating Voltage: 3.3 to 5.5V DC
- Output Signal: I2C/UART
- GNSS module： Quectel-L76K
- GNSS bands:
  - GPS L1 C/A: 1575.42 MHz
  - GLONASS L1: 1602 MHz
  - BeiDou B1: 1561.098 MHz
- Channels: 32 tracking channels

- Channels: 32 tracking channels
- Sensitivity:
  - Auto-aquisition: -147dBm
  - Tracking: -162dBm
  - Re-acquisition: -159dBm
- Accuracy:
  - Position: 2.0m CEP
  - Velocity: 0.1m/s
  - Acceleration: 0.1m/s^2
  - Timing: 30ns
- Time for First Positioning: 30s for cold start; 2s for hot start
- Power Consumption: 40mA
- Antenna Interface: IPEX1
- Antenna Frequency: 1561-1575.42MHz±3MHz
- Dimension: 27mm×37mm/1.06×1.46"

Board Overview



Num	Label	Description
1	D/T	I2C data line SDA/UART Data Transmitting- TX
2	C/R	I2C clock line SCL/UART Data Receiving- RX
3	-	GND
4	+	VCC

When no GPS signal is acquired, the indicator light appears red. When GPS signal is successfully acquired, the indicator light is green

acquired, the indicator light is green.

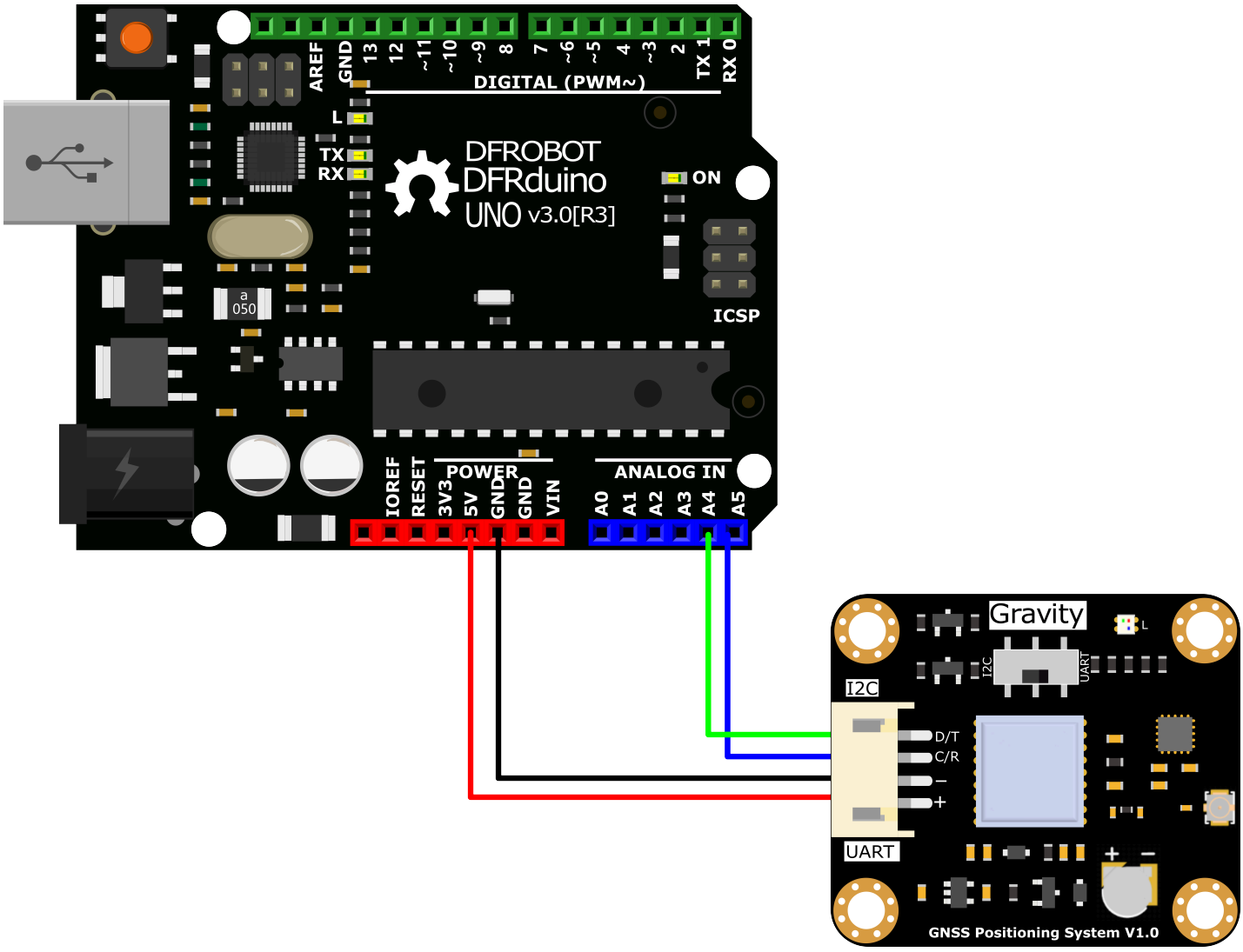
Tutorial for Arduino

Requirements

- **Hardware**
  - DFRduino UNO R3 (https://www.dfrobot.com/product-838.html) (or similar) x 1
  - Gravity: GNSS Positioning Module x 1
- **Software**
  - Arduino IDE (https://www.arduino.cc/en/Main/Software)
  - Download and install the **DFRobot GNSS Library** (https://github.com/DFRobot/DFRobot_GNSS) (About how to install the library? (https://www.arduino.cc/en/Guide/Libraries#.UxU8mdzF9H0))

Read Data via I2C

Connection Diagram



Sample Code

- Connect the module to Arduino according to the connection diagram above. It can also be used with Gravity I/O expansion board to prototype ideas faster.
- Change the select switch on the sensor to I2C.
- Download and install the DFRobot_GNSS library (https://github.com/DFRobot/DFRobot_GNSS) (About how to install the library? (https://www.arduino.cc/en/Guide/Libraries#.UxU8mdzF9H0))
- Open Arduino IDE and upload the following code to Arduino UNO.

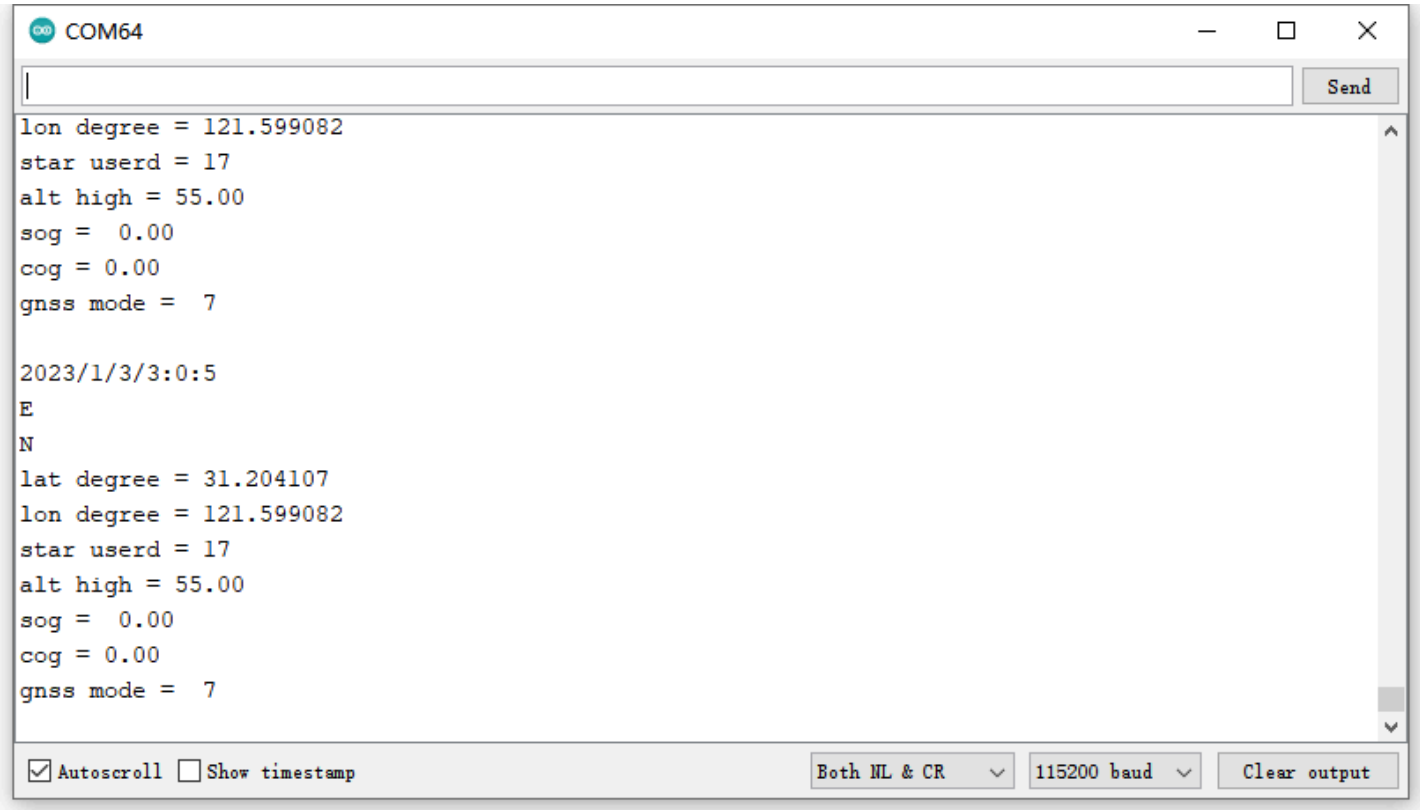
- Open the serial monitor of Arduino IDE, set the baud rate to 115200, and observe the printed result.

```
#include "DFRobot_GNSS.h"

DFRobot_GNSS_I2C gnss(&Wire ,GNSS_DEVICE_ADDR);
/*
#ifdef ESP_PLATFORM
  // ESP32 user hardware uart
  // RX io16
  // TX io17
  DFRobot_GNSS_UART gnss(&Serial2 ,9600);
#else
  // Arduino user software uart
  // RX io10
  // TX io11
  SoftwareSerial mySerial(10 ,11);
  DFRobot_GNSS_UART gnss(&mySerial ,9600);
#endif
*/
void setup()
{
  Serial.begin(115200);
  while(!gnss.begin()){
    Serial.println("NO Deivces !");
    delay(1000);
  }

  gnss.enablePower();

/** Set the galaxy to be used
 *   eGPS               USE gps
 *   eBeiDou            USE beidou
 *   eGPS_BeiDou        USE gps + beidou
 *   eGLONASS           USE glonass
 *   eGPS_GLONASS       USE gps + glonass
 *   eBeiDou_GLONASS    USE beidou +glonass
 *   eGPS_BeiDou_GLONASS USE gps + beidou + glonass
 */
  gnss.setGnss(eGPS_BeiDou_GLONASS);


  // gnss.setRgbOff();
  gnss.setRgbOn();
  // gnss.disablePower();
}
```

```
void loop()
{
  sTim_t utc = gnss.getUTC();
  sTim_t date = gnss.getDate();

  sLonLat_t lat = gnss.getLat();
  sLonLat_t lon = gnss.getLon();
  double high = gnss.getAlt();
  uint8_t starUserd = gnss.getNumSatUsed();
  double sog = gnss.getSog();
  double cog = gnss.getCog();

  Serial.println("");
  Serial.print(date.year);
  Serial.print("/");
  Serial.print(date.month);
  Serial.print("/");
  Serial.print(date.date);
  Serial.print("/");
  Serial.print(utc.hour);
  Serial.print(":");
  Serial.print(utc.minute);
  Serial.print(":");
  Serial.print(utc.second);
  Serial.println();
  Serial.println((char)lat.latDirection);
  Serial.println((char)lon.lonDirection);

  // Serial.print("lat DDMM.MMMMM = ");
  // Serial.println(lat.latitude, 5);
  // Serial.print(" lon DDDMM.MMMMM = ");
  // Serial.println(lon.lonitude, 5);
  Serial.print("lat degree = ");
  Serial.println(lat.latitudeDegree,6);
  Serial.print("lon degree = ");
  Serial.println(lon.lonitudeDegree,6);

  Serial.print("star userd = ");
  Serial.println(starUserd);
  Serial.print("alt high = ");
  Serial.println(high);
  Serial.print("sog =  ");
  Serial.println(sog);
  Serial.print("cog = ");
  Serial.println(cog);
  Serial.print("gnss mode =  ");
  Serial.println(gnss.getGnssMode());
  delay(1000);
}
```

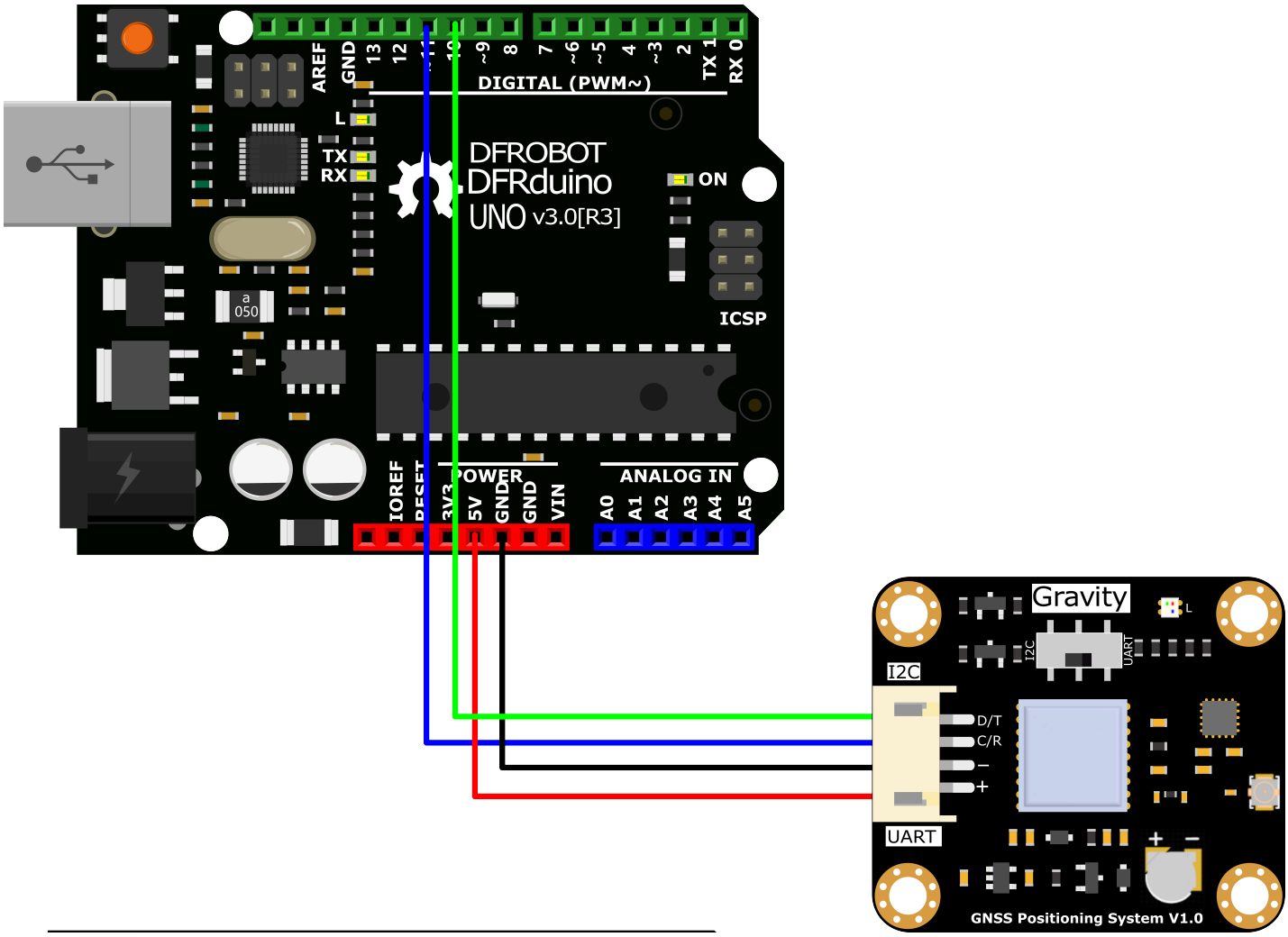
Result

Open serial monitor to see the result.



Read Data via UART

Connection Diagram



Sample Code

- Connect the module to Arduino according to the connection diagram above. It can also be used with Gravity I/O expansion board to prototype ideas faster.
- Change the select switch on the sensor to UART.
- Download and install the DFRobot_GNSS library (https://github.com/DFRobot/DFRobot_GNSS) (About how to install the library? (https://www.arduino.cc/en/Guide/Libraries#.UxU8mdzF9H0))
- Open Arduino IDE and upload the following code to Arduino UNO.
- Open the serial monitor of Arduino IDE, set the baud rate to 115200, and observe the printed result.

result.

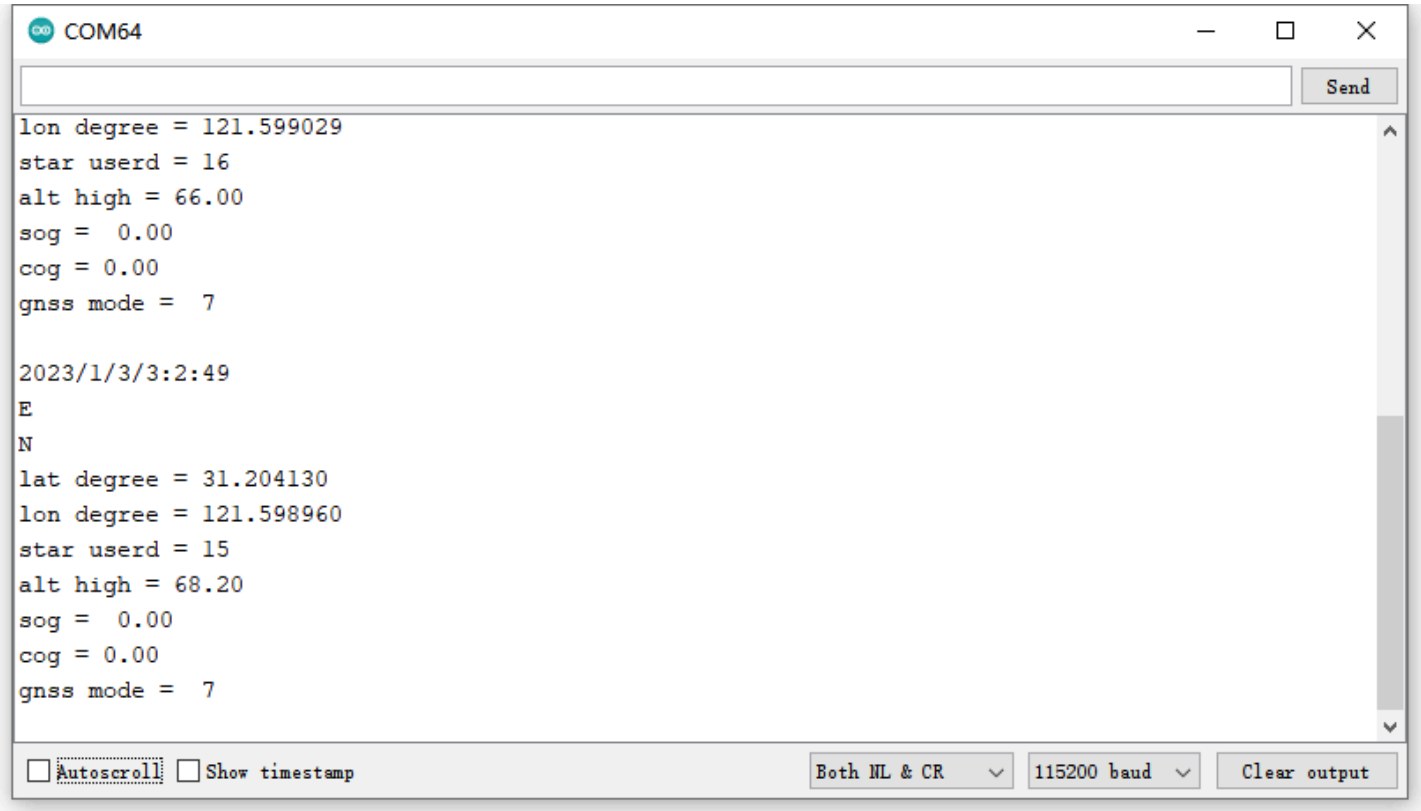
```
 /*!
  * @file  getGNSS.ino
  * @copyright Copyright (c) 2010 DFRobot Co.Ltd (http://www.dfrobot.com)
  * @license The MIT License (MIT)
  * @author ZhixinLiu(zhixin.liu@dfrobot.com)
  * @version V0.1
  * @date 2022-08-15
  * @url https://github.com/dfrobot/DFRobot_GNSS
  */

#include "DFRobot_GNSS.h"

//DFRobot_GNSS_I2C gnss(&Wire ,GNSS_DEVICE_ADDR);

#ifdef ESP_PLATFORM
  // ESP32 user hardware uart
  // RX io16
  // TX io17
  DFRobot_GNSS_UART gnss(&Serial2 ,9600);
#else
  // Arduino user software uart
  // RX io10
  // TX io11
  SoftwareSerial mySerial(10 ,11);
  DFRobot_GNSS_UART gnss(&mySerial ,9600);
#endif

void setup()
{
  Serial.begin(115200);
  while(!gnss.begin()){
    Serial.println("NO Deivces !");
    delay(1000);
  }

  gnss.enablePower();

/** Set the galaxy to be used
 *   eGPS              USE gps
 *   eBeiDou           USE beidou
 *   eGPS_BeiDou       USE gps + beidou
 *   eGLONASS          USE glonass
 *   eGPS_GLONASS      USE gps + glonass
 *   eBeiDou_GLONASS   USE beidou +glonass
```

```
 *   eGPS_BeiDou_GLONASS USE gps + beidou + glonass
 */
  gnss.setGnss(eGPS_BeiDou_GLONASS);



  // gnss.setRgbOff();
  gnss.setRgbOn();
  // gnss.disablePower();
}

void loop()
{
  sTim_t utc = gnss.getUTC();
  sTim_t date = gnss.getDate();
  sLonLat_t lat = gnss.getLat();
  sLonLat_t lon = gnss.getLon();
  double high = gnss.getAlt();
  uint8_t starUserd = gnss.getNumSatUsed();
  double sog = gnss.getSog();
  double cog = gnss.getCog();

  Serial.println("");
  Serial.print(date.year);
  Serial.print("/");
  Serial.print(date.month);
  Serial.print("/");
  Serial.print(date.date);
  Serial.print("/");
  Serial.print(utc.hour);
  Serial.print(":");
  Serial.print(utc.minute);
  Serial.print(":");
  Serial.print(utc.second);
  Serial.println();
  Serial.println((char)lat.latDirection);
  Serial.println((char)lon.lonDirection);

  // Serial.print("lat DDMM.MMMMM = ");
  // Serial.println(lat.latitude, 5);
  // Serial.print(" lon DDDMM.MMMMM = ");
  // Serial.println(lon.lonitude, 5);
  Serial.print("lat degree = ");
  Serial.println(lat.latitudeDegree,6);
  Serial.print("lon degree = ");
  Serial.println(lon.lonitudeDegree,6);

  Serial.print("star userd = ");
  Serial.println(starUserd);
  Serial.print("alt high = ");
```

```
    Serial.println(high);
    Serial.print("sog =  ");
    Serial.println(sog);
    Serial.print("cog = ");

    Serial.println(cog);
    Serial.print("gnss mode =  ");
    Serial.println(gnss.getGnssMode());
    delay(1000);
  }
```

Result

Open serial monitor to see the result.



FAQ

Q1: Why can't I get a GPS signal?

A1: This product only uses GPS functionality and does not use WiFi or Bluetooth for location assistance like mobile phones or other electronic devices. So we recommend that you use this product by a window, on a penthouse, or outdoors. When the on-board indicator light turns from red to green, it means that the product has successfully acquired a GPS signal.

Q2：Encountering I2C address conflicts?

A2: For a comprehensive guide on identifying and resolving I2C address conflicts in embedded systems, check out this detailed article: How to Resolve I2C Address Conflicts (https://wiki.dfrobot.com/How_to_Resolve_I2C_Address_Conflicts_in_Embedded_Systems). It covers

practical ha

More questions and cool idea, please visit DFRobot Forum (https://www.dfrobot.com/forum/)

More Documents

- Schematics (https://dfimg.dfrobot.com/nobody/wiki/347648af182e0024278ae9c33b514125.pdf)
- Dimensions & Component Layout (https://dfimg.dfrobot.com/nobody/wiki/40c2f9f5ba39bd9fee5be7ee3af5b864.pdf)

Get **Gravity: GNSS GPS BeiDou Receiver Module** (https://www.dfrobot.com/product-2651.html) from DFRobot Store or **DFRobot Distributor**. (https://www.dfrobot.com/distributor)

Turn to the Top