

# GPS Module With Enclosure (SKU:TEL0094)

## Contents

- 1 Introduction
- 2 Specification
- 3 Application
- 4 Pinout
- 5 Library Explanation
- 6 Tutorial
  - 6.1 Connection Diagram
  - 6.2 Sample code
  - 6.3 Result
- 7 More



(/wiki/index.php/File:TEL0094\_GPS\_front.png)

## Introduction

You must have used a GPS receiver. It is easy to be found in most place, such as smartphones, cars, and computers. They are used to track your loaction all over the globe. GPS Receiver for Arduino is a unit embedding GPS module and antenna in a small foot-print enclosure. By using TinyGPS library, Arduino can retrieve geographic coordinates (latitude and longitude, altitude), speed, heading and GMT time. The update rate is an important performance index of a GPS receiver. Most GPS in mobile phones provide an update rate of 1Hz, which means, only one set of data can be retrieved in one second. For GPS receivers with 1~10Hz, the data interval is much reduced and thus can be used for more demanding applications (e.g. on fast-moving vehicles). For more details ,please refer to the Documents as attached below.

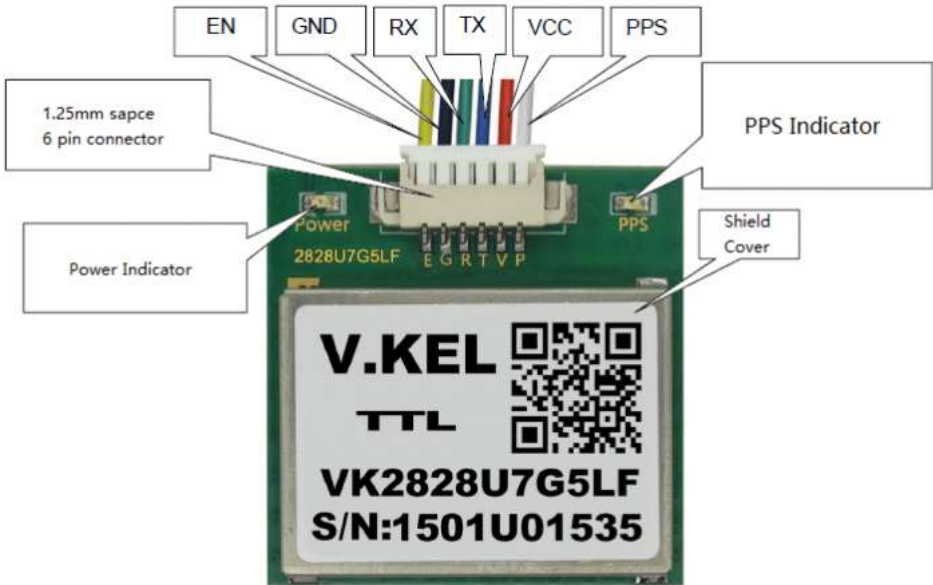
## Specification

- UBX-G7020-KT solution
- 1Hz(default) ~ 10Hz output
- 9600bps(default) [support:4800 , 9600 , 19200 , 38400 , 57600 , 115200 , 230400 , 460800,921600] TTL serial interface
- 5v @ 30mA (support 3.3~5v)
- 56-Channel receiver
- Extremely high sensitivity -161dBm
- Accuracy 2.5m (Autonomous) / <2m[SBAS]
- Operating temperature: -40°C to 85°C
- Hot Start : 1s
- Warm Start : 28s
- Cold Start : 29s
- Module Size 28\*28\*8.6mm
- Enclosure Size 37x48x16mm
- LED indicator

## Application

Item tracking  
Aircraft Control

## Pinout



(/wiki/index.php/File:GPS\_1.png)

Pin Name	Color	Description
PPS	White	time standard pulse output

VCC	Red	Power Supply Input 3.3V-5.V
TX	Blue	UART/TTL Interface
RX	Green	UART/TTL Interface
GND	Black	GND
EN	Yellow	Enable Pin.The device is in shutdown mode when voltage to this pin is LOW and enable when is HIGH or floating

## Library Explanation

To use, simply create an instance of an object like this:

```
1 #include "TinyGPS.h"
2 TinyGPS gps;
```

Feed the object serial NMEA data one character at a time using the encode() method. (TinyGPS does not handle retrieving serial data from a GPS unit.) When encode() returns "true", a valid sentence has just changed the TinyGPS object's internal state. For example:

```
1 void loop()
2 {
3   while (Serial.available())
4   {
5     int c = Serial.read();
6     if (gps.encode(c))
7     {
8       // process new gps info here
9     }
10  }
11 }
```

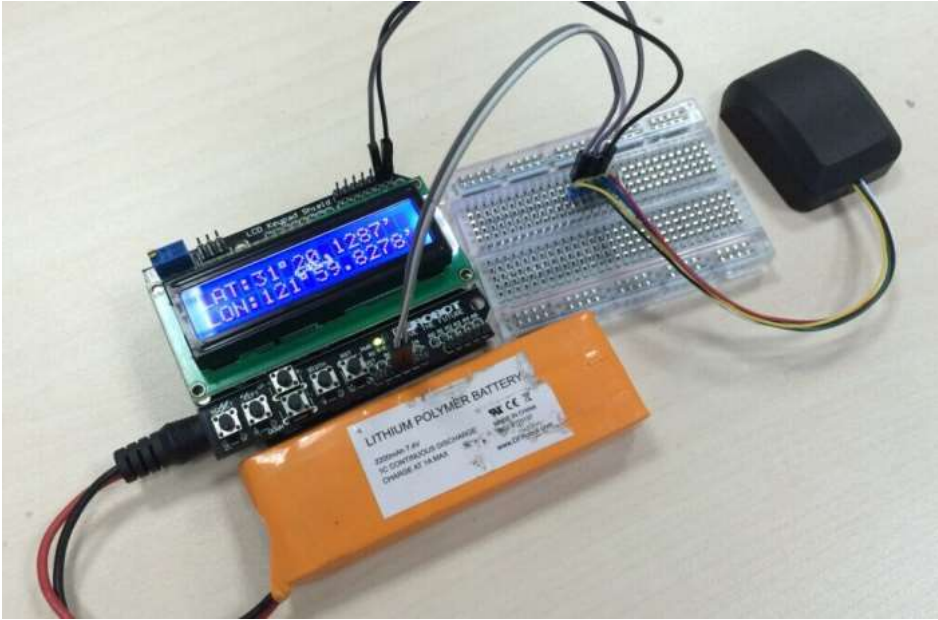
You can then query the object to get various tidbits of data. To test whether the data returned is stale, examine the (optional) parameter "fix\_age" which returns the number of milliseconds since the data was encoded.

```
1 long lat, lon;
2 unsigned long fix_age, time, date, speed, course;
3 unsigned long chars;
4 unsigned short sentences, failed_checksum;
5
6 // retrieves +/- Lat/Long in 100000ths of a degree
7 gps.get_position(&lat, &lon, &fix_age);
8
9 // time in hhmmsscc, date in ddmmyy
10 gps.get_datetime(&date, &time, &fix_age);
11
12 // returns speed in 100ths of a knot
13 speed = gps.speed();
14
15 // course in 100ths of a degree
16 course = gps.course();
```

## Tutorial

GPS with LCD Module Sample

### Connection Diagram



(/wiki/index.php/File:TEL0094\_GPS\_test.jpg)

GPS	Arduino
VCC	VCC
GND	GND
RX	TX
TX	RX

Sample code

Library: TinyGPS V1.3 (<https://github.com/mikalhart/TinyGPS/archive/v13.zip>)

```

#include <TinyGPS.h>
#include <LiquidCrystal.h>

TinyGPS gps;
LiquidCrystal lcd(8, 9, 4, 5, 6, 7);           //LCD driver pins
int led = 13;

long lat, lon;
unsigned long fix_age, time, date, speed, course;
unsigned long chars;
unsigned short sentences, failed_checksum;
//int year;
//byte month, day, hour, minute, second, hundredths;

int DEG;
int MIN1;
int MIN2;

void LAT(){                                //Latitude state
  DEG=lat/1000000;
  MIN1=(lat/10000)%100;
  MIN2=lat%10000;

  lcd.setCursor(0,0);                      // set the LCD cursor position
  lcd.print("LAT:");
  lcd.print(DEG);
  lcd.write(0xDF);
  lcd.print(MIN1);
  lcd.print(".");
  lcd.print(MIN2);
  lcd.print("' ");
}
void LON(){                                //Longitude state
  DEG=lon/1000000;
  MIN1=(lon/10000)%100;
  MIN2=lon%10000;

  lcd.setCursor(0,1);                      // set the LCD cursor position
  lcd.print("LON:");
  lcd.print(DEG);
  lcd.write(0xDF);
  lcd.print(MIN1);
  lcd.print(".");
  lcd.print(MIN2);
  lcd.print("' ");
}

void setup()
{
  Serial.begin(9600);                      //Set the GPS baud rate.

  pinMode(led, OUTPUT);

  lcd.begin(16, 2);                        // start the library
  lcd.setCursor(0,0);                      // set the LCD cursor position
  lcd.print("GPS test");                   // print a simple message on the LCD
  delay(2000);
}

void loop()
{
  while (Serial.available())
  {
    digitalWrite(led, HIGH);
    int c = Serial.read();                 // Read the GPS data
    if (gps.encode(c))                     // Check the GPS data
    {
      // process new gps info here
    }
  }
  digitalWrite(led, LOW);
  gps.get_position(&lat, &lon, &fix_age); // retrieves +/- Lat/Long in 100000ths of a degree
  gps.get_datetime(&date, &time, &fix_age); // time in hhmmsscc, date in ddmmyy

  //gps.crack_datetime(&year, &month, &day, //Date/time cracking
  //&hour, &minute, &second, &hundredths, &fix_age);

  LAT();
  LON();
}

```

## Result

As shown in the picture above , you can get GPS data on the LCD screen.

## More

Library TinyGPS (<https://github.com/CainZ/V.KEL-GPS/raw/master/TinyGPS.rar>)

VK2828U7G5LF datasheet (<https://github.com/CainZ/V.KEL-GPS/raw/master/VK2828U7G5LF%20Data%20Sheet%2020150902.pdf>)

VKEL GPS Test Assistant ([https://github.com/CainZ/V.KEL-GPS/raw/master/u-centersetup\\_v8.12.zip](https://github.com/CainZ/V.KEL-GPS/raw/master/u-centersetup_v8.12.zip))

→ (/wiki/index.php/File:Nextredirectltr.png)Go Shopping Click to buy on DFRobot Store ([http://www.dfrobot.com/index.php?route=product/product&product\\_id=1302&search=TEL0094&description=true#.VbXyCR6Ppjs](http://www.dfrobot.com/index.php?route=product/product&product_id=1302&search=TEL0094&description=true#.VbXyCR6Ppjs))

Categories (/wiki/index.php/Special:Categories): Product Manual (/wiki/index.php/Category:Product\_Manual)

| TEL Series (/wiki/index.php/Category:TEL\_Series) | Wireless (/wiki/index.php/Category:Wireless)

This page was last modified on 9 October 2015, at 09:26.

Content is available under GNU Free Documentation License 1.3 or later (<https://www.gnu.org/copyleft/fdl.html>) unless otherwise noted.



(<https://www.gnu.org/copyleft/fdl.html>)



(<http://www.mediawiki.org/>)