UBX-M8030
GNSS Receiver

User Manual
Version: 3.2

Table of Contents

Overview

The GNSS Receiver provides a USB version 2.0 FS (Full Speed, 12 Mb/s) Interface with Microsoft® certified USB drivers for Windows XP, Windows Vista, Windows 7, Windows 8 and Windows 10 operating systems.

Supports Protocol NMEA ASCII 0183, 4.0 (V2.3 or V4.1 configurable). This is the newest concurrent GNSS USB Receiver that can track multiple GNSS systems (GPS, GLONASS, Galileo, Beidou, SBAS and QZSS signals). Up to 3 GNSS can be received and processed concurrently. The receiver is setup by default to concurrently track and process GPS L1 C/A and GLONASS L1OF.

QZSS and SBAS share the same frequency band as GPS and can always be processed in conjunction with GPS. The exceptional performance of the u-Blox 8 GNSS engine delivers high sensitivity and minimal acquisition times.

Enjoy the GPS life!

Getting Started

You can easily get started by testing your new GPS/GLONASS receiver using the uBlox U-Center GNSS Software which can be downloaded for free at the links provided below. The U-Center software automatically installs the drivers for your device. Also check out the U-Center User Guide to help get your receiver setup and running.

U-Center for Windows:
https://www.u-blox.com/en/product/u-center-windows

If you choose not to use the uBlox U-Center GNSS Software, the drivers are available on the cdrom included with your receiver. You can also download the contents of the cdrom at the following Google Drive link:

http://bit.ly/QGPgnss

Applications

Automotive and Marine Navigation

Fleet Management

AVL and Location-based

Notebook navigation

Distance measurement

Sports and Recreation

Packing list

USB GNSS Receiver
CD-ROM Drivers

Main Features

1. Built in U-blox M8030 concurrent GNSS chipset.

2. 72 channel GPS L1C/A, SBAS L1C/A, QZSS L1C/A, QZSS L1 SAIF, GLONASS L1OF, BeiDou B1I, Galileo E1B/C.

3. Superior sensitivity up to -167dBm.

4. Built-in GAGAN/WAAS/EGNOS/MSAS Demodulator without any additional hardware.

5. Low power consumption

6. Support NMEA0183 4.0 (V2.3 or V4.1 configurable)

7. Waterproof design for industry standard IPX6

8. Support OS: Windows XP/Vista/7/8/10

Specifications

Electrical Characteristics (Receiver)	
GPS Chipset	72-channel U-blox 8 Engine
Frequency	GPS/QZSS L1 C/A L1 GLONASS L1 FDMA Beidou B1I Galileo E1B/C
SBAS	GAGAN, WAAS, EGNOS, MSAS L1C/A
Channels	72
Sensitivity (Tracking)	-167dBm @ GPS& GLONASS
GPS Chipset	72-channel U-blox 8 Engine
Frequency	GPS/QZSS L1 C/A L1 GLONASS L1 FDMA Beidou B1I Galileo E1B/C

Accuracy	
Position Horizontal	Autonomous 2.5m, 2m SBAS
Velocity	0.1m/s
TIMEPULSE	1 pulse per second, synchronized at rising edge, pulse length 100ms
Accuracy of time pulse	RMS 30 ns
Frequency of time pulse signal	0.25 Hz… 10 MHz (configurable)

Datum	
Datum	WGS-84

Acquisition Rate	
Hot start	1 sec., average (with ephemeris and almanac valid)
Warm start	2 sec., average (with almanac but not ephemeris)
Cold start	26 sec., average (neither almanac nor ephemeris)

Protocol and Interface	
GNSS Protocol	NMEA 0183 4.0
GNSS Output Data	Software command setting Default is GPS & GLONSASS: GGA, GSA, GSV, RMC, VTG, GLL
GNSS transfer rate	Auto Baud Rate N,8,1 for NMEA
Navigation update rate	Max to 10HZ, Default 1HZ
Output terminal	USB

Dynamic Condition	
Acceleration Limit	Less than 4g
Altitude Limit	50,000 m
Velocity Limit	500 m/s
Jerk Limit	20 m/ second^3
Heading accuracy	0.5 degrees

Temperature	
Operating	-40°~ 80°C
Storage	-40°~ 85°C
Humidity	Up to 95% non-condensing
Power	
Voltage	5V ±5%
Current	25mA typical
Physical Characteristics	
Dimension	38mm x 49mm x 16mm
USB Cable Length	200 cm
Low Noise Amp	
Amplifier Gain	27 dB Typical
Filtering	-25dB (+100 MHz)
Output VSWR	2.0 Max.
Voltage	DC 3 ~ 5.0V
Current	15mA max @ 5VDC

Due to continuous product improvements, all specifications are subject to change

Supported GNSS Constellations

This concurrent GNSS receiver can receive and track multiple GNSS systems: GPS, Galileo, GLONASS and BeiDou. By default, receivers are configured for concurrent GPS and GLONASS, including SBAS and QZSS reception. Galileo is not enabled as the default configuration.

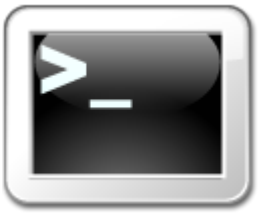
The module can be configured to receive any single GNSS constellation or within the set of permissible combinations shown below. The augmentation systems: SBAS and QZSS can be enabled only if GPS operation is configured.

GPS	Galileo	GLONASS	Beidou
•	•	x	x
•	•	•	x
•	•	x	•
•	x	•	x
•	x	x	•
x	•	•	x
x	•	x	•
x	x	•	•

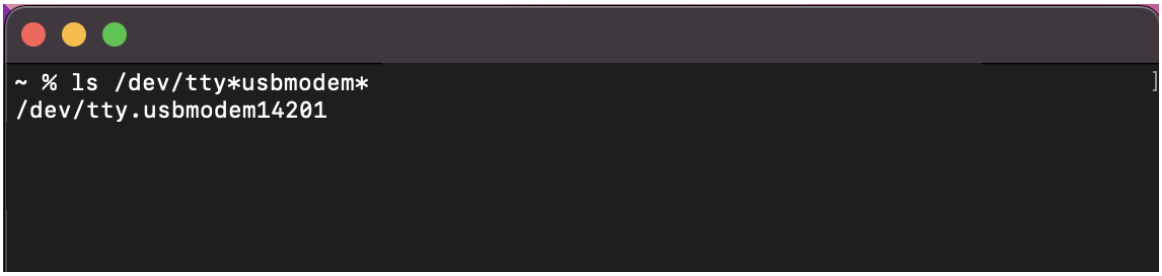
Permissible GNSS combinations (• = enabled **x** = disabled)

Using OpenCPN on MAC OS

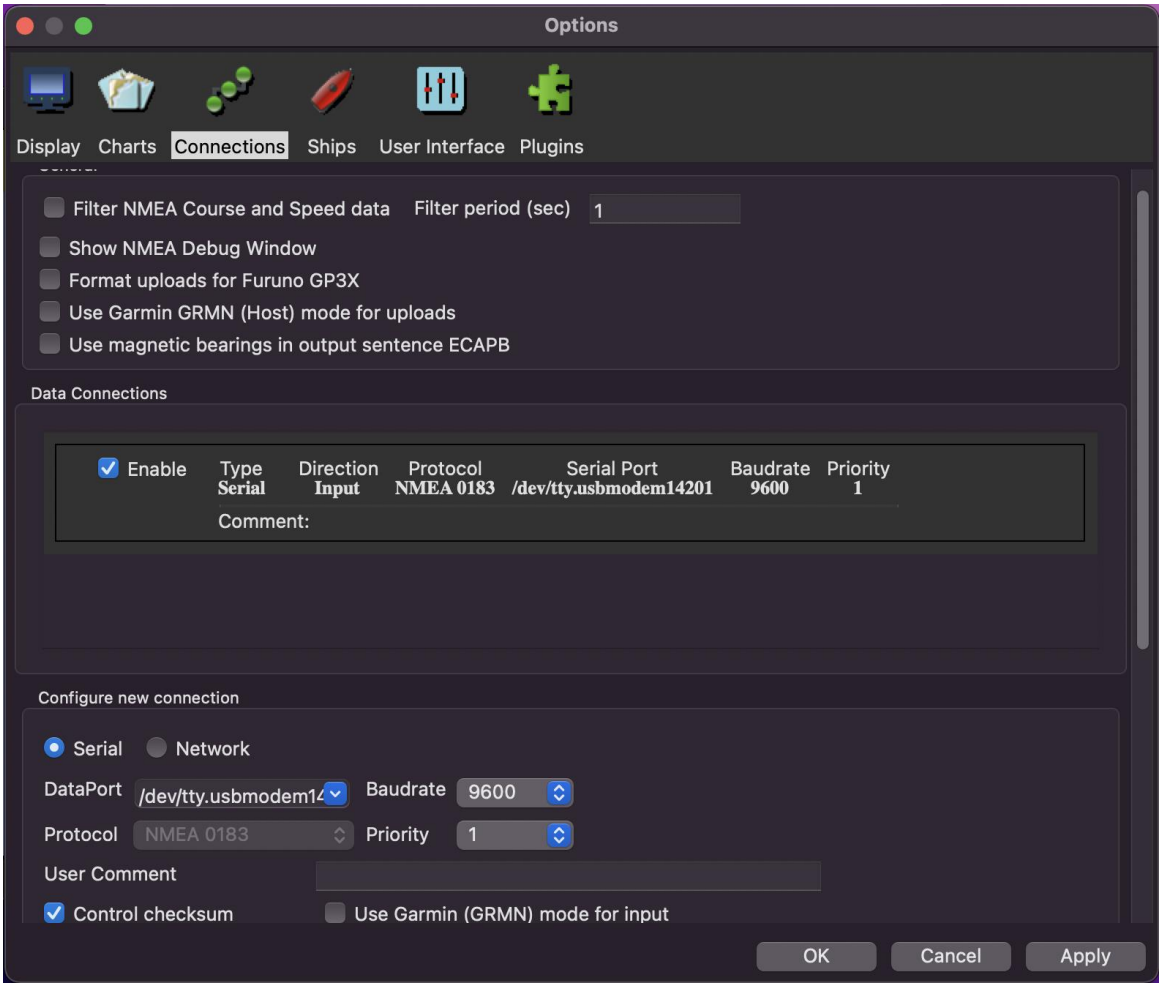
1. Start by opening the Terminal.app. The icon looks like this:



2. Plug in your GPS receiver to you MAC computer.
3. In the Terminal.app you need to type “ls /dev/tty*usbmodem*”

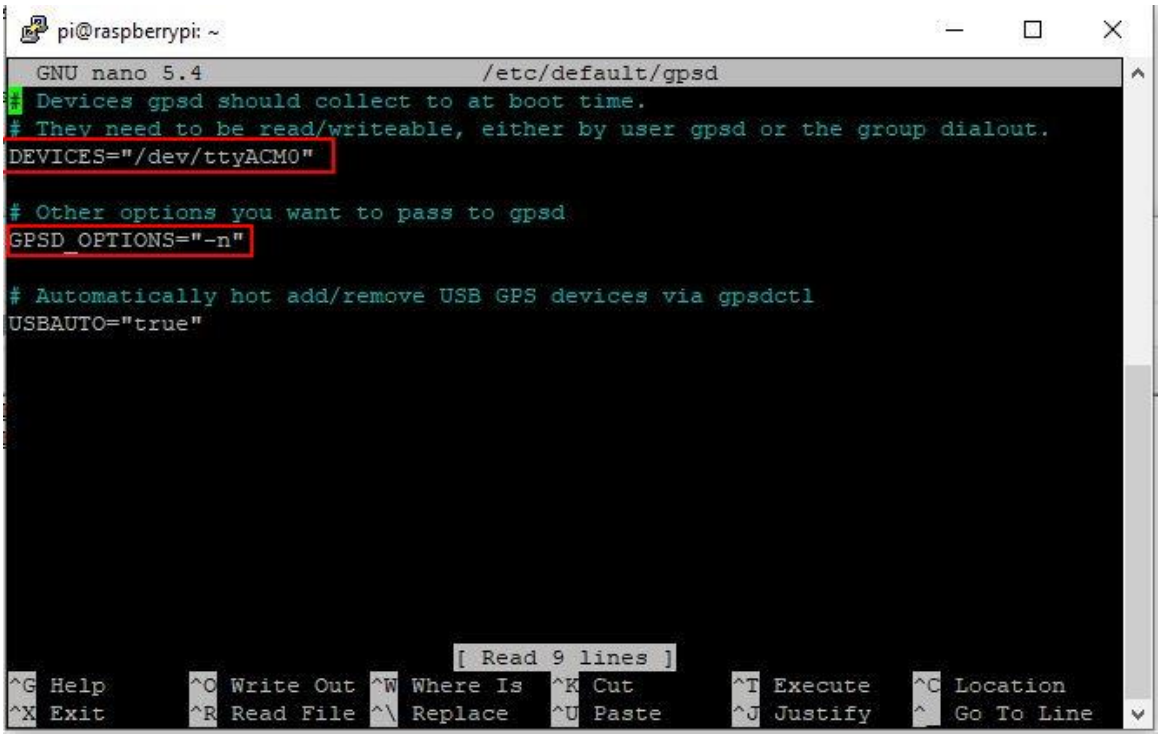


4. In our case, the only /dev/tty.usbmodem on the system is /dev/tty.usbmodem14201 and yours may be slightly different.
5. Within OpenCPN, add a new connection, choose Serial, then add your device into the DataPort field and set Baudrate to 9600.
6. Press OK or Apply to complete the setup.



Raspberry Pi

1. You can connect to your Raspberry Pi via the console connection or remotely through SSH using Putty. https://www.putty.org/
2. Before starting, be sure your Raspberry Pi libraries are updated with the following commands:
   sudo apt-get update
   sudo apt-get upgrade
3. Before plugging the GPS receiver, list out the existing TTY devices on your Raspberry Pi.
   ls /dev/tty*
4. Proceed with plugging in the GPS receiver and then list out TTY devices again
   ls /dev/tty*
5. You should see a change in the TTY devices. Find the TTY device that was added to your list and this will be your GPS Receiver. In our case, we have a new device named ttyACM0. Try displaying the streaming data from the device with this command:
   sudo cat /dev/ttyACM0
6. If you have a stream of data scrolling, then your GPS receiver is working and streaming data to the serial port. The data stream can be halted using Control-C.
7. Now let's install GPSD and GPSD-Clients
   sudo apt -y install gpsd
   sudo apt -y install gpsd-clients
8. Edit the GPSD configuration file using the Nano editor. Add your device to "DEVICES=" and the change the "GPSD_OPTIONS=" as shown in the image below
   sudo nano /etc/default/gpsd



9. Save the configuration file using Control-X and “Y’ to save the file.
10. Reboot your Raspberry Pi to give the GPSD service a clean start
    sudo reboot
11. The GPSD service should now be active. We can check with the following command.
    systemctl is-active gpsd
12. The results of the command should show “active”
13. Check your GPS position using CGPS command as follows.
    cgps -s