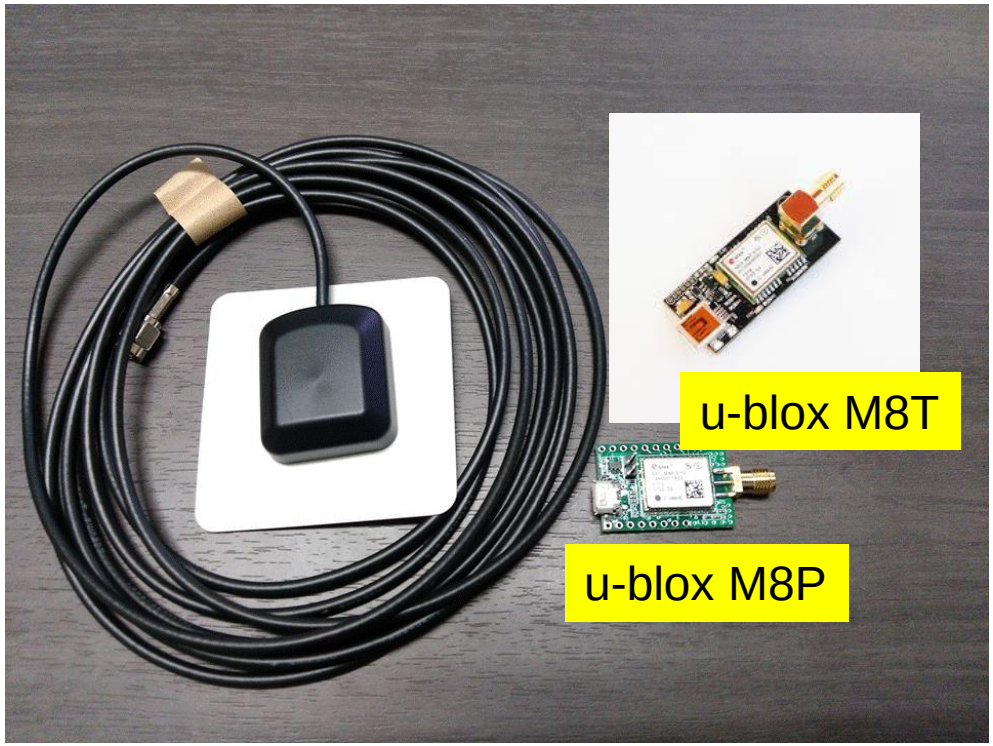
MGA Webinar 1st June

How to use free RTK Software?

Tokyo Univ. of Marine Science and Technology : Nobuaki Kubo

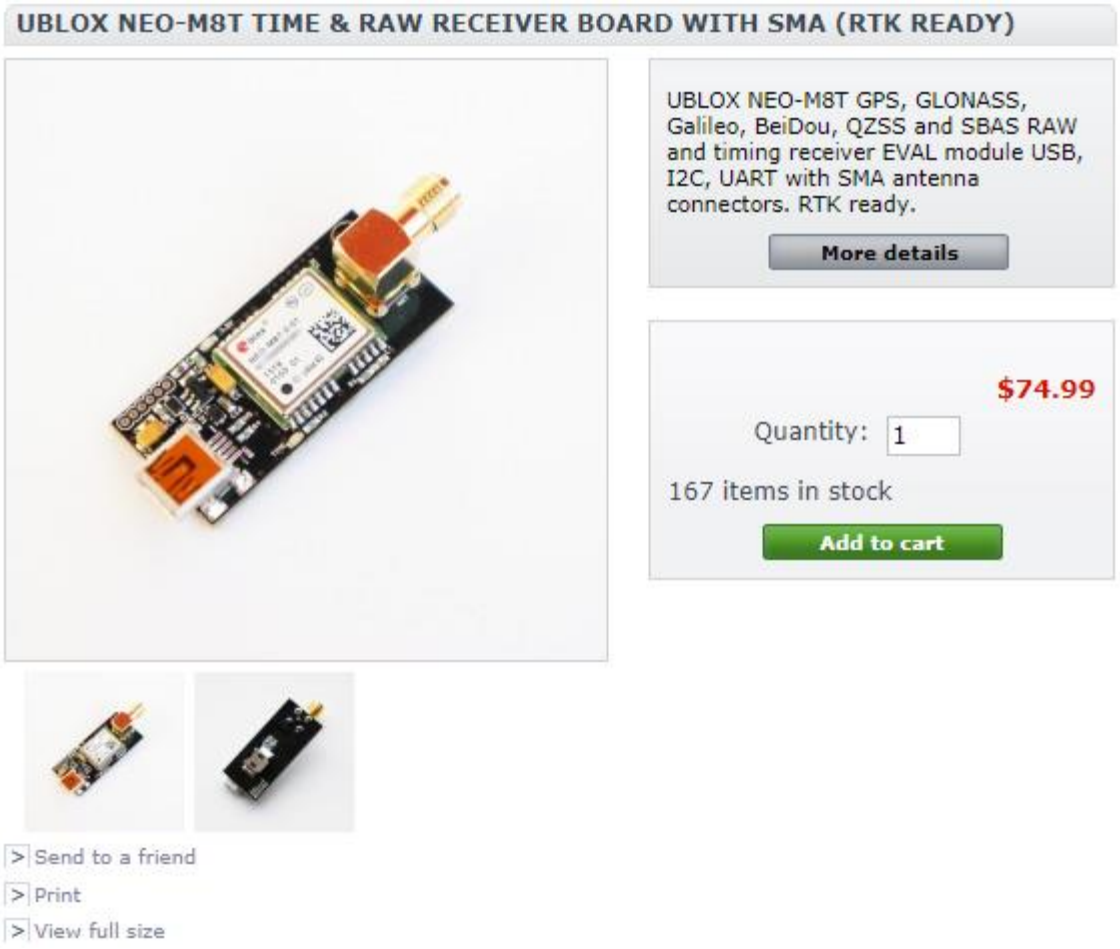
Receiver used in this training

1. u-blox GNSS receiver (M8P or M8T)
2. Trimble NetR9



You can purchase M8T

- http://www.csgshop.com/product.php?id_product=205



Some Data used in Practice

Static raw data set (24h)

- u-blox M8T (ref/rover) + Trimble/NovAtel ant.
- Trimble NetR9 (ref) Trimble ant.

Static raw data set (1h)

- u-blox M8P (ref/rover) + Trimble/NovAtel ant.
- Trimble NetR9 (ref/rover) + Trimble/NovAtel ant.

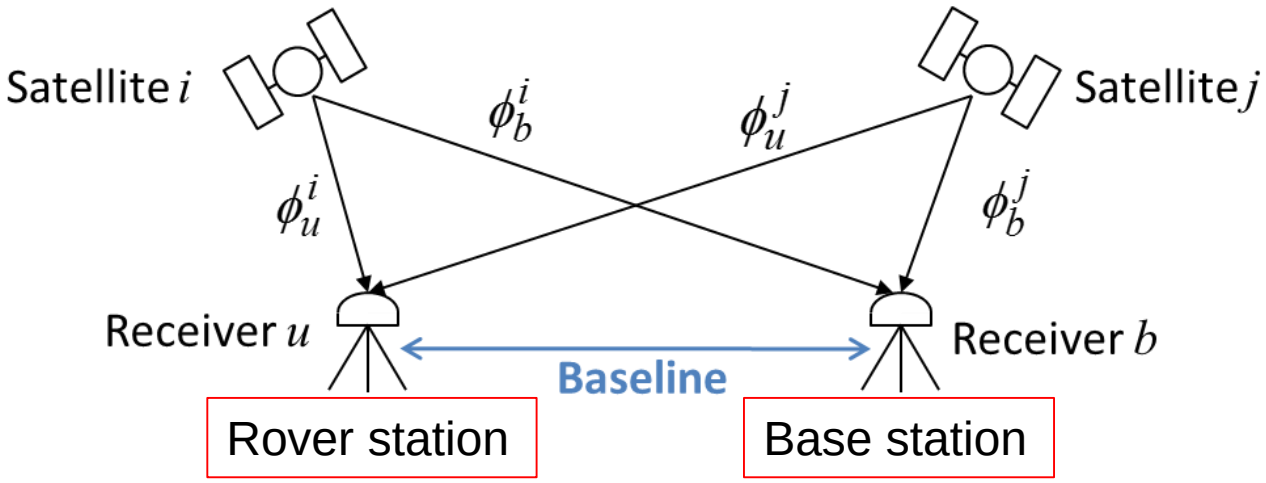
Kinematic raw data set (0.5h)

- u-blox M8T (ref/rover) + Trimble/NovAtel ant.
- Trimble NetR9 (ref/rover) + Trimble/NovAtel ant.

You will learn…

- Using the “RTKLIB” and “internet(smart-phone)” + Ntrip server, you can check RTK.
- Through this webinar, you will learn how to process RTKLIB in the post-processing or real-time processing.
- You need to decide which is more important for your purpose.
- Please keep in mind that there are specific terms in GNSS.

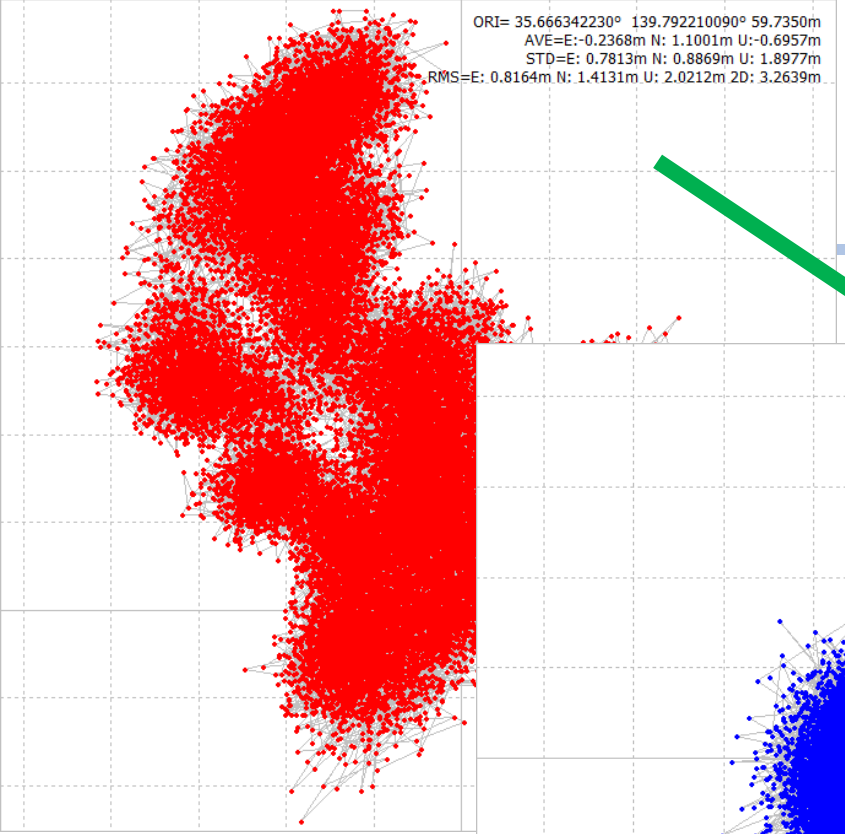
RTK tells you not absolute position but vector from base station

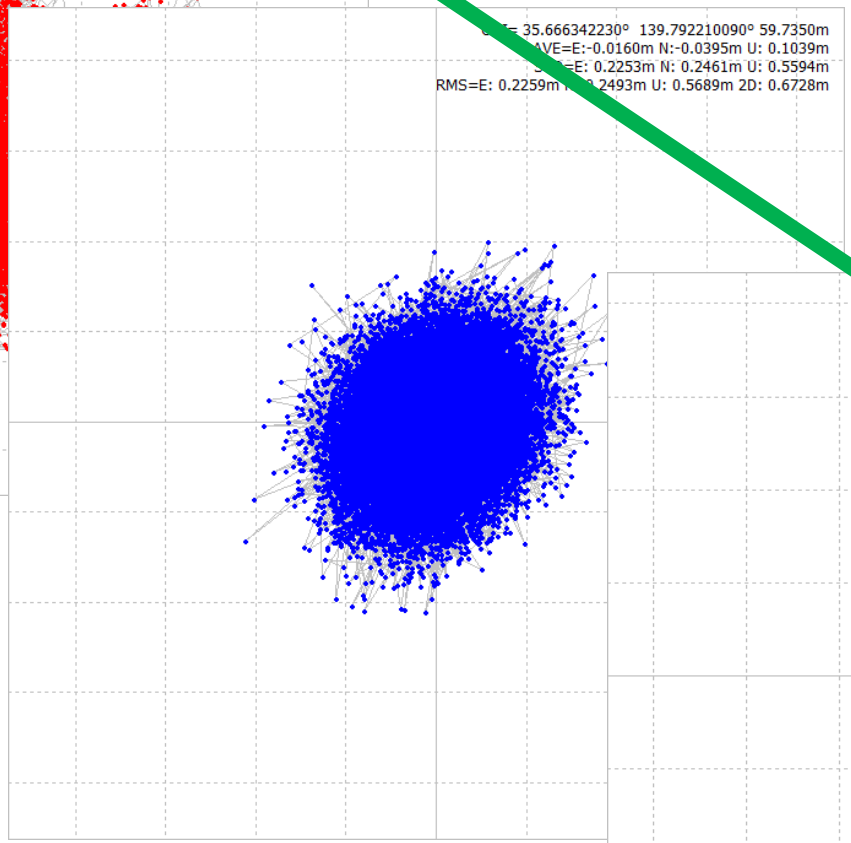


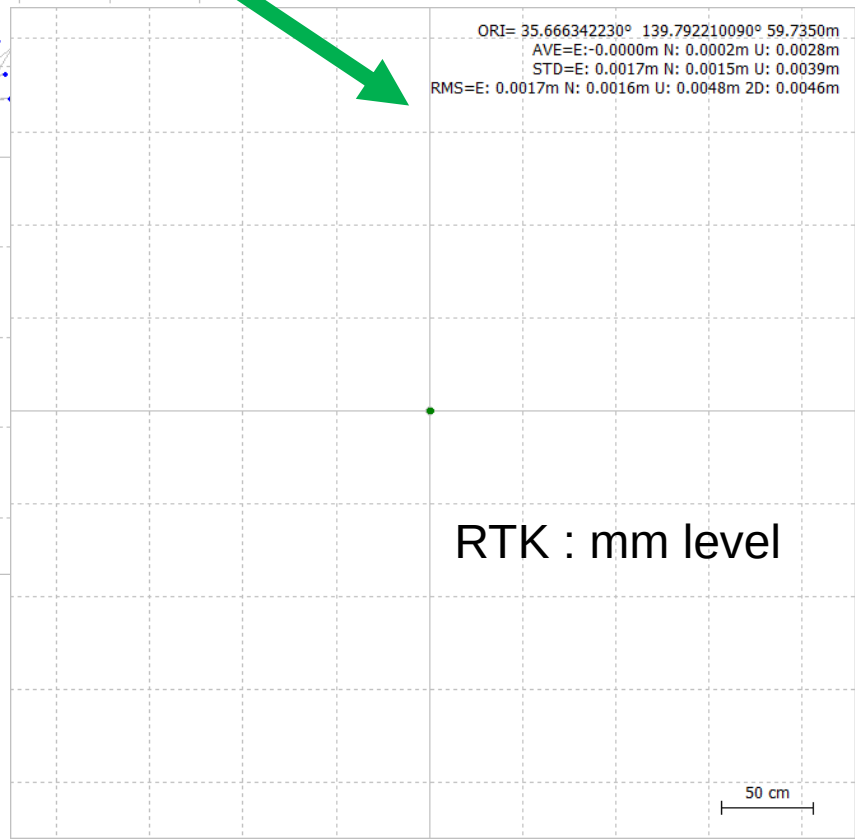
Once you determine the precise position at base station, you will know
the precise vector from base antenna to rover antenna.
Then you can know your precise position : (X,Y,Z) = (Xref,Yref,Zref) + (dX,dY,dZ)
If the position at base station deviates 1m, your RTK position will be deviates 1m

RTK performance

12h, rooftop, our building





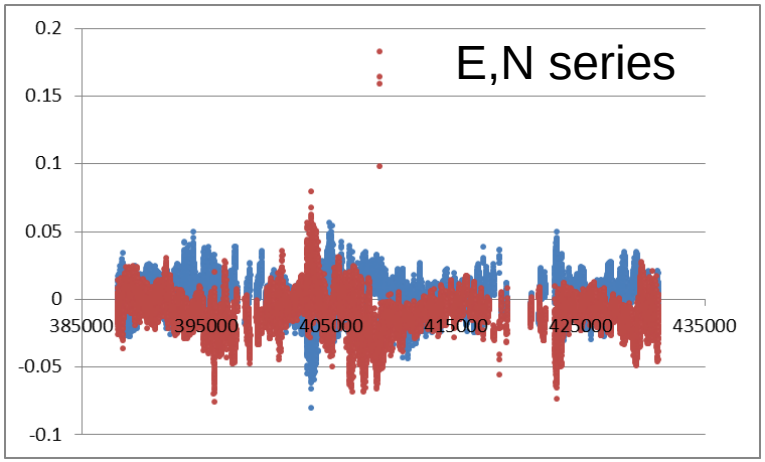


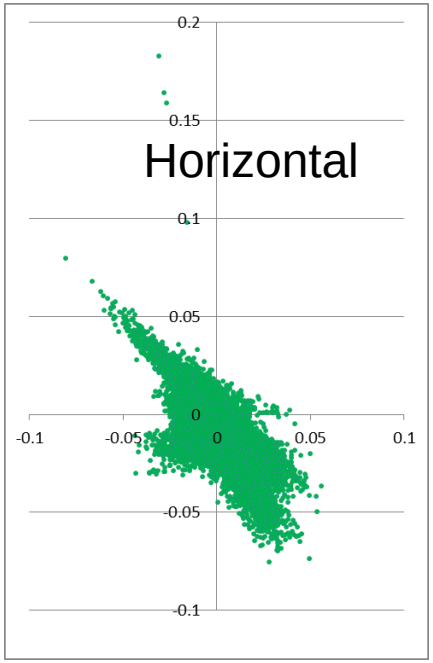
RTK : mm level

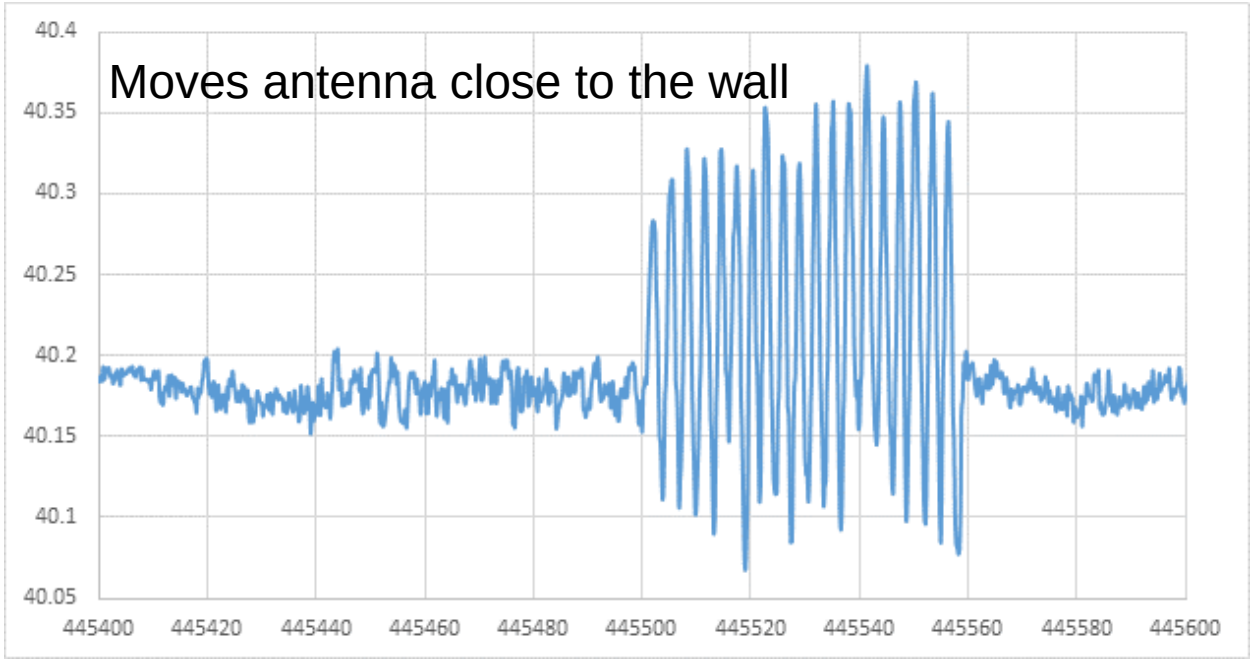
Same scale

Recent Test : RTK on the wall

Monitoring for structure deformations

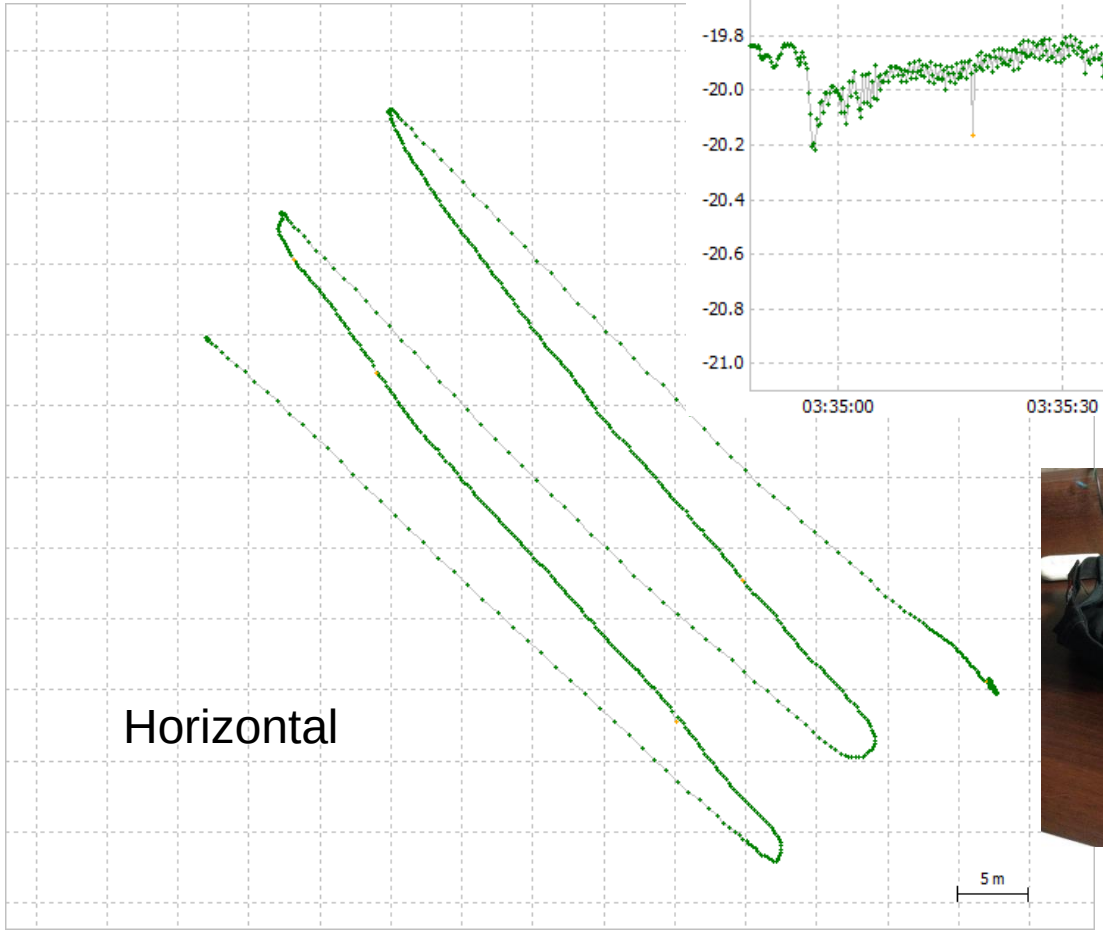






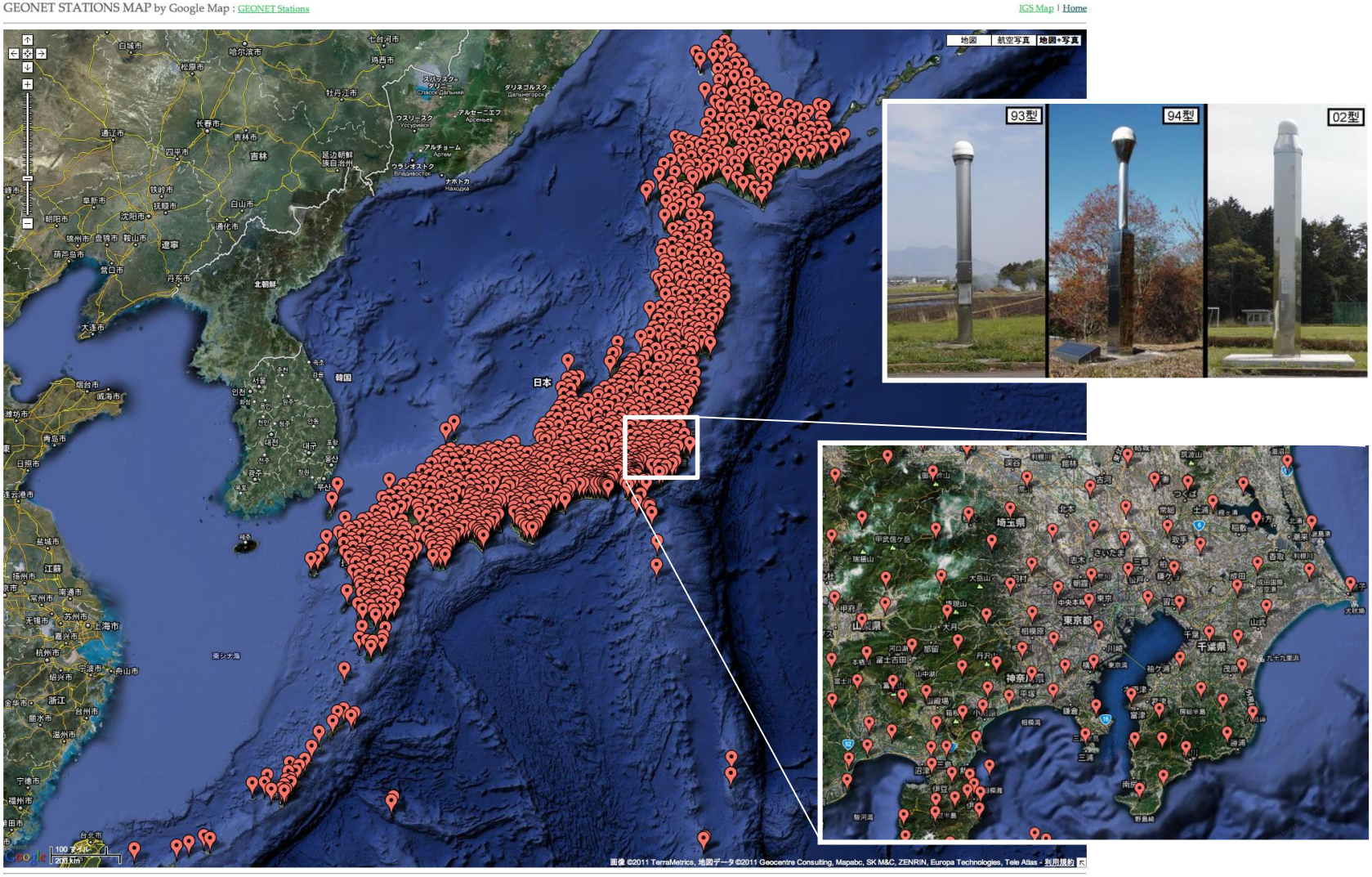
Recent Test : Running

Height series



Horizontal

Japanese GEONET



(http://terras.gsi.go.jp/ja/index.htm)

Difference between expensive and low-cost receiver

	Survey-grade receiver	Low-cost receiver
Cost	$100,00～	$100～
Multiple GNSS	Perfect	BeiDou or Glonass Other are OK
Multiple Frequency	Perfect	L1/B1/E1/G1 only
Number of channel	400-500-	-100
RTK (short baseline) + open sky	**Perfect**	**Almost perfect**
RTK (over 20 km baseline) + open sky	**Almost perfect up to 100 km or more**	**Impossible**
RTK under mid obstructed area (short)	**Almost perfect**	**May be difficult**
RTK under dense obstructed area (short)	**Sometimes not good**	**Difficult**
Accuracy of fixed position + open	mm	→
Accuracy of code position + open	Deci-meter	1-2 meter

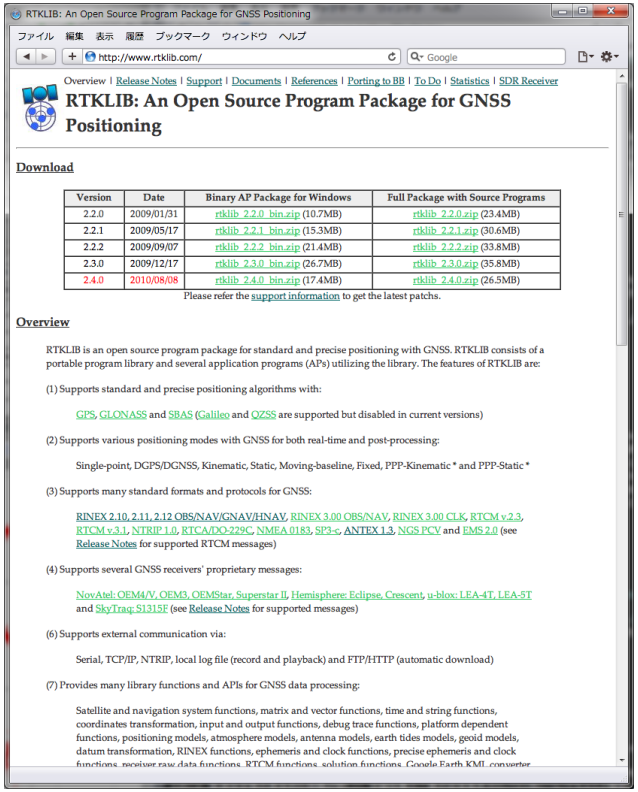
Error source mitigation (Typical)

Source/Error	SPP	DGNSS	RTK	PPP
Satellite clock model 1 m (rms)	→	0.0 m	0.0 m	0.01 – 0.1 m
Satellite ephemeris 1 m (rms)	→	0.0 m	0.0 m	0.01 - 0.1 m
Ionospheric delay 2-10 m (zenith) × 3 at 5°	1 - 2 m (zenith)	0.1 - 0.2 m	0.01 m	0.01 m
Tropospheric delay 2.3-2.5m (zenith) × 10 at 5°	0.1 - 0.5 m (zenith)	0.1 - 0.2 m	0.01 m	0.01 m
Multipath (open sky) Code : 0.5-1 m Carrier : -1 cm	→ Code	→ Code	→ Carrier	→ Carrier
Receiver Noise Code : 0.1-0.5 m Carrier : 1-2 mm	→ Code	→ Code	→ Carrier	→ Carrier
Finally…	2-3 m	- 1 m	- 1 cm	- 10 cm

RTKLIB Practice (1)

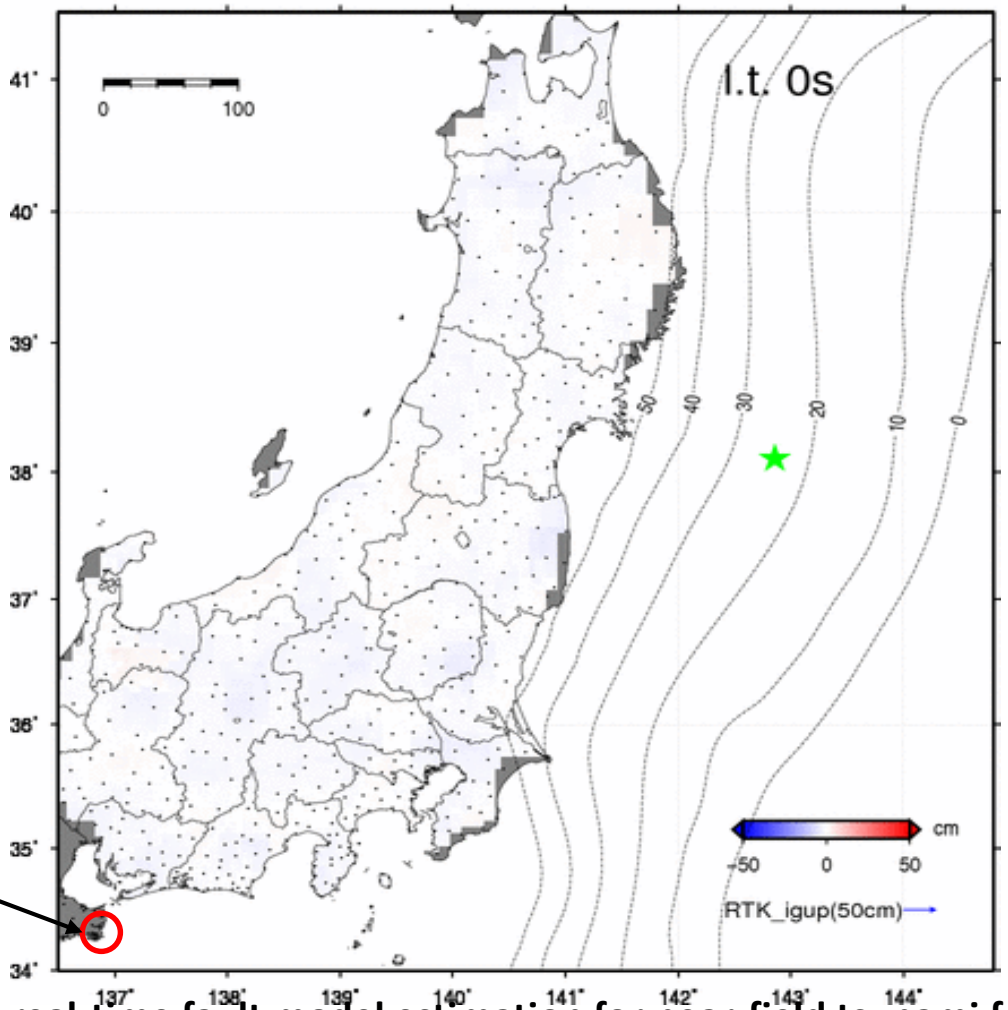
RTKLIB

- **An Open Source Software Package for GNSS Positioning**
  - Has been developed since 2006
  - The latest version 2.4.2 p12 distributed under BSD license
- **Portable APIs and Useful APs**
  - "All-in-one" package for Windows
  - CLI APs for any environments



http://www.rtklib.com or
https://github.com/tomojitakasu/RTKLIB

RTKLIB: Application



Reference Station

Y. Ohta et al., Quasi real-time fault model estimation for near-field tsunami forecasting base on RTK-GPS analysis: Application to the 2011 Tohoku-Oki earthquake (Mw 9.0), JGR-solid earth, 2012

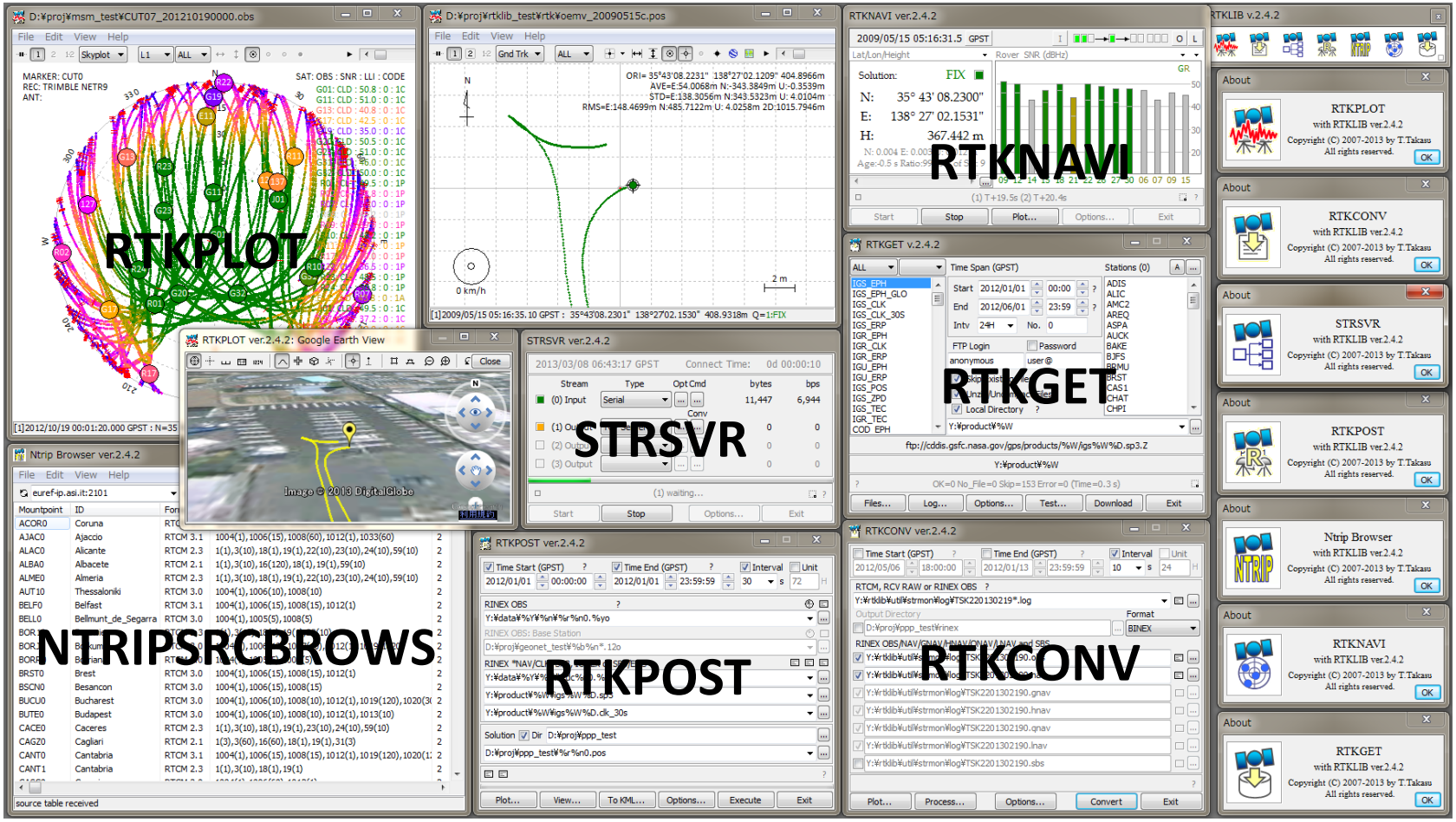
RTKLIB: History

- 2006/4 v.0.0.0 First version for RTK+C program lecture
- 2007/1 v.1.0.0 Simple post processing AP
- 2008/7 v.2.1.0 Add APs, support medium-range
- 2009/1 v.2.2.0 Add real-time AP, support NTRIP, start to distribute as **Open Source S/W**
- 2009/5 v.2.2.1 Support RTCM, NRTK, many receivers
- 2009/12 v.2.3.0 Support GLONASS, several receivers
- 2010/8 v.2.4.0 Support PPP Real-time/Post-processing PPP and Long-baseline RTK (<1000 km)
- 2011/6 v.2.4.1 Support QZSS, JAVAD receiver, …
- 2013/4 v.2.4.2 Support Galileo, Enable BeiDou, …
- 2016/12 v.2.4.3 TBD

RTKLIB: Features

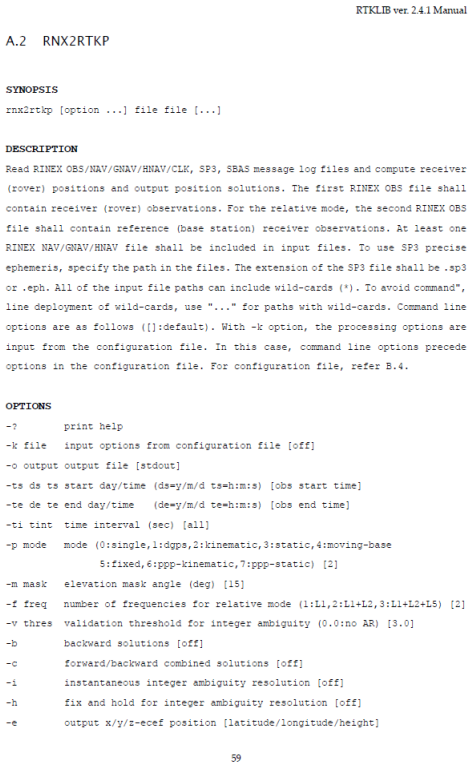
- **Standard and precise positioning algorithms with**:
  - GPS, GLONASS, QZSS, Galileo, BeiDou and SBAS
- **Real-time and post-processing by various modes**:
  - Single, SBAS, DGPS, RTK, Static, Moving-base and PPP
- **Supports many formats/protocols and receivers**:
  - RINEX 2/3, RTCM 2/3, BINEX, NTRIP 1.0, NMEA0183, SP3, RINEX CLK, ANTEX, NGS PCV, IONEX, RTCA-DO-229, EMS,
  - NovAtel, JAVAD, Hemisphere, u-blox, SkyTraq, NVS, …
- **Supports real-time communication via**:
  - Serial, TCP/IP, NTRIP and file streams

RTKLIB: GUI APs



RTKLIB: CLI APs

- **RNX2RTKP (rnx2rtkp)**
  Post-processing Positioning
- **RTKRCV (rtkrcv)**
  Real-time Positioning
- **CONVBIN (convbin)**
  RINEX Translator
- **STR2STR (str2str)**
  Stream Server
- **POS2KML (pos2kml)**
  Google Earth Converter



RTKLIB ver. 2.4.1 Manual

A.2 RNX2RTKP

SYNOPSIS

rnx2rtkp [option ...] file file [...]

DESCRIPTION

Read RINEX OBS/NAV/GNAV/HNAV/CLK, SP3, SBAS message log files and compute receiver (rover) positions and output position solutions. The first RINEX OBS file shall contain receiver (rover) observations. For the relative mode, the second RINEX OBS file shall contain reference (base station) receiver observations. At least one RINEX NAV/GNAV/HNAV file shall be included in input files. To use SP3 precise ephemeris, specify the path in the files. The extension of the SP3 file shall be .sp3 or .eph. All of the input file paths can include wild-cards (*). To avoid command", line deployment of wild-cards, use "..." for paths with wild-cards. Command line options are as follows ([]:default). With -k option, the processing options are input from the configuration file. In this case, command line options precede options in the configuration file. For configuration file, refer B.4.

OPTIONS

```
-?        print help
-k file   input options from configuration file [off]
-o output output file [stdout]
-ts ds ts start day/time (ds=y/m/d ts=h:m:s) [obs start time]
-te de te end day/time   (de=y/m/d te=h:m:s) [obs end time]
-ti tint  time interval (sec) [all]
-p mode   mode (0:single,1:dgps,2:kinematic,3:static,4:moving-base
               5:fixed,6:ppp-kinematic,7:ppp-static) [2]
-m mask   elevation mask angle (deg) [15]
-f freq   number of frequencies for relative mode (1:L1,2:L1+L2,3:L1+L2+L5) [2]
-v thres  validation threshold for integer ambiguity (0.0:no AR) [3.0]
-b        backward solutions [off]
-c        forward/backward combined solutions [off]
-i        instantaneous integer ambiguity resolution [off]
-h        fix and hold for integer ambiguity resolution [off]
-e        output x/y/z-ecef position [latitude/longitude/height]
```

59

CLI Command Reference

RTKLIB: Package Structure

rtklib_2.4.2.zip

```
/src                    : Source programs of RTKLIB libraries
     /rcv               : Source programs depending on GPS/GNSS receiv.
/bin                    : Executable binary APs and DLLs for Windows
/data                   : Sample data for APs
/app                    : Build environment for APs
     /rtknavi           : RTKNAVI (GUI)
     /strsvr            : STRSVR (GUI)
     /rtkpost           : RTKPOST (GUI)
     /rtkpost_mkl       : RTKPOST_MKL (GUI)
     /rtkplot           : RTKPLOT (GUI)
     /rtkconv           : RTKCONV (GUI)
     /srctblbrows       : NTRIP source table browser (GUI)
     /rtkrcv            : RTKRCV (console)
     /rnx2rtkp          : RNX2RTKP (console)
     /pos2kml           : POS2KML (console)
     /convbin           : CONVBIN (console)
     /str2str           : STR2STR (console)
     /appcmn            : Common routines for GUI APs
     /icon              : Icon data for GUI APs
/mkl                    : Intel MKL libraries for Borland environment
/test                   : Test program and data
/util                   : Utilities
/doc                    : Document files
```

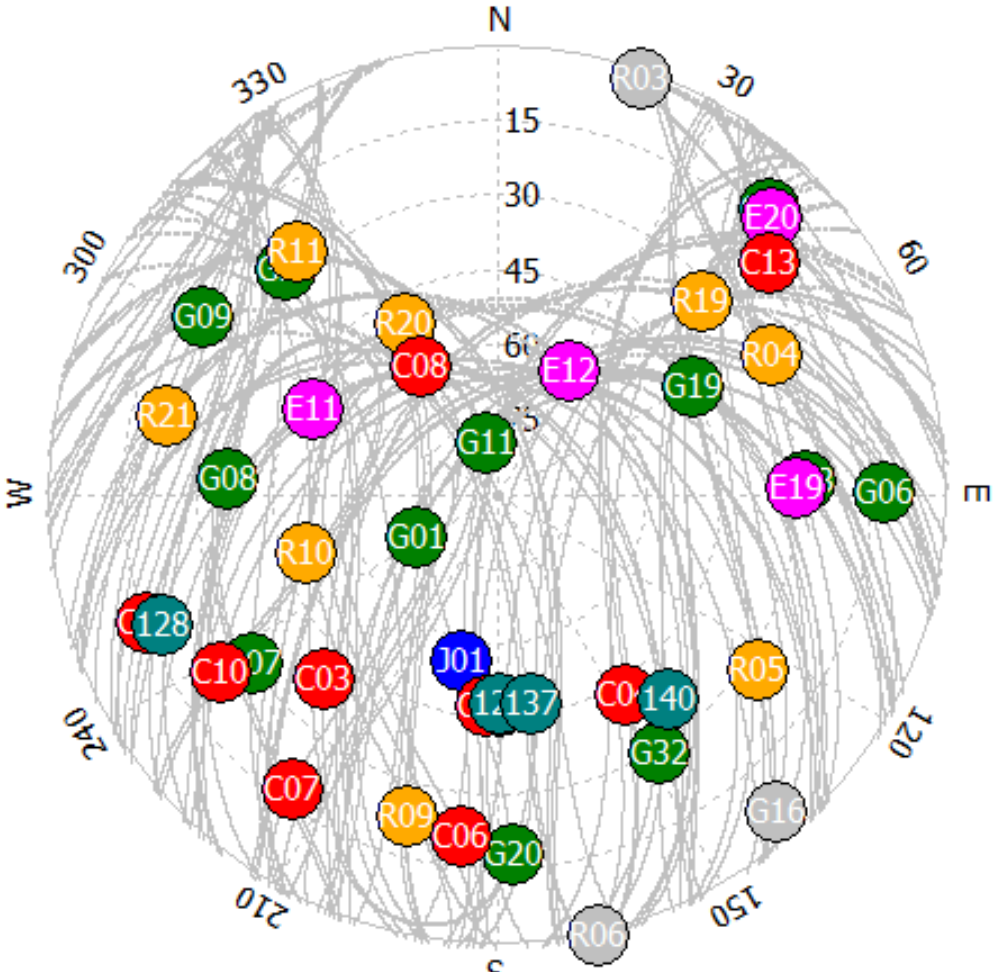
RTKLIB: APIs

```
/* matrix and vector functions */
mat(),imat(),zeros(),eye(),dot(),norm(),matcpy(),matmul(),matinv(),solve(),lsq(),filter(),smoother(),matprint(),matfprint()
/* time and string functions */
str2num(),str2time(),time2str(),epoch2time(),time2epoch(),gpst2time(),time2gpst(),timeadd(),timediff(),gpst2utc(),utc2gpst(),
timeget(),time2doy(),adjgpsweek(),tickget(),sleepms()
/* coordinates functions */
ecef2pos(),pos2ecef(),ecef2enu(),enu2ecef(),covenu(),covecef(),xyz2enu(),geoidh(),loaddatump(),tokyo2jgd(),jgd2tokyo()
/* input/output functions */
readpcv(),readpos(),sortobs(),uniqeph(),screent()
/* positioning models */
eph2pos(),geph2pos(),satpos(),satposv(),satposiode(),satazel(),geodist(),dops(),ionmodel(),ionmapf(),tropmodel(),tropmapf(),
antmodel(),csmooth()
/* single-point positioning */
pntpos(),pntvel()
/* rinex functions */
readrnx(),readrnxt(),outrnxobsh(),outrnxnavh(),outrnxnavb(),uncompress(),convrnx()
/* precise ephemeris functions */
readsp3(),readsap(),eph2posp(),satposp()
/* receiver raw data functions */
getbitu(),getbits(),crc32(),crc24q(),decode_word(),decode_frame(),init_raw(),free_raw(),input_raw(),input_rawf(),input_oem4(),
input_oem3(),input_ubx(),input_ss2(),input_cres(),input_oem4f(),input_oem3f(),input_ubxf(),input_ss2f(),input_cresf()
/* rtcm functions */
init_rtcm(),free_rtcm(),input_rtcm2(),input_rtcm3(),input_rtcm2f(),input_rtcm3f()
/* solution functions */
readsol(),readsolt(),outsolheads(),outsols(),outsolexs(),outsolhead(),outsol(),outsolex(),setsolopt(),setsolformat(),
outnmea_rmc(),outnmea_gga(),outnmea_gsa(),outnmea_gsv(),
/* SBAS functions */
sbsreadmsg(),sbsreadmsgt(),sbsoutmsg(),sbsupdatestat(),sbsdecodemsg(),sbssatpos(),sbspntpos()
/* integer least-square estimation */
lambda()
/* realtime kinematic positioning */
rtkinit(),rtkfree(),rtkpos()
/* post-processing positioning */
postpos(),postposopt(),readopts(),writeopts()
/* stream data input/output */
strinitcom(),strinit(),strlock(),strunlock(),stropen(),strclose(),strread(),strwrite(),strsync(),strstat(),strsum(), strsetopt(),
strgettime()
/* stream server functions */
strsvrinit(),strsvrstart(),strsvrstop(),strsvrstat()
/* rtk server functions */
rtksvrinit(),rtksvrstart(),rtksvrstop(),rtksvrlock(),rtksvrunlock(),rtksvrostat(),rtksvrsstat() ...
```

RTKLIB: Supported Receivers

Format	Data Message Types							
	GPS Raw Meas Data	GLONASS Raw Meas	GPS Ephemeris	GLONASS Ephemeris	ION/UTC Parameters	Antenna Info	SBAS Messages	Others
RTCM v.2.3	Type 18, 19	Type 18, 19	Type 17	-	-	Type 3, 22	-	Type 1, 9, 14, 16
RTCM v.3.1	Type 1002, 1004	Type 1010, 1012	Type 1019	Type 1020	-	Type 1005, 1006, 1007, 1008, 1033	-	SSR corrections
NovAtel OEM4/V, OEMStar	RANGEB, RANGECMPB	RANGEB, RANGECMPB	RAWEPHEMB	GLO-EPHEMERISB	IONUTCB	-	RAWWAAS-FRAMEB	-
NovAtel OEM3	RGEB, RGED	-	REPB	-	IONB, UTCB	-	FRMB	-
NovAtel Superstar II	ID#23	-	ID#22	-	-	-	ID#67	ID#20, #21
u-blox LEA-4T, LEA-5T	UBX RXM-RAW	-	UBX RXM-SFRB	-	UBX RXM-SFRB	-	UBX RXM-SFRB	-
Hemisphere Crescent, Eclipse	bin 96	-	bin 95	-	bin 94	-	bin 80	-
SkyTraq S1315F	msg 0xDD (221)	-	msg 0xE0 (224)	-	msg 0xE0 (224)	-	-	msg 0xDC (220)
JAVAD (GRIL/GREIS)	[R*],[r*],[*R],[*r],[P*],[p*],[*p],[D*],[*d],[E*],[*E],[F*]	[R*],[r*],[*R],[*r],[P*],[p*],[*p],[D*],[*d],[E*],[*E],[F*]	[GE],[GD],[gd]	[NE],[LD]	[IO],[UO],[GD]	-	[WD]	[~~],[::],[RD],[SI],[NN],[TC],QZSS Data, Galileo Data
Furuno GW10 II	msg 0x08	-	msg 0x24	-	msg 0x26	-	msg 0x03	msg 0x20

Multi-GNSS Support



2013-06-12 10:20 GPST

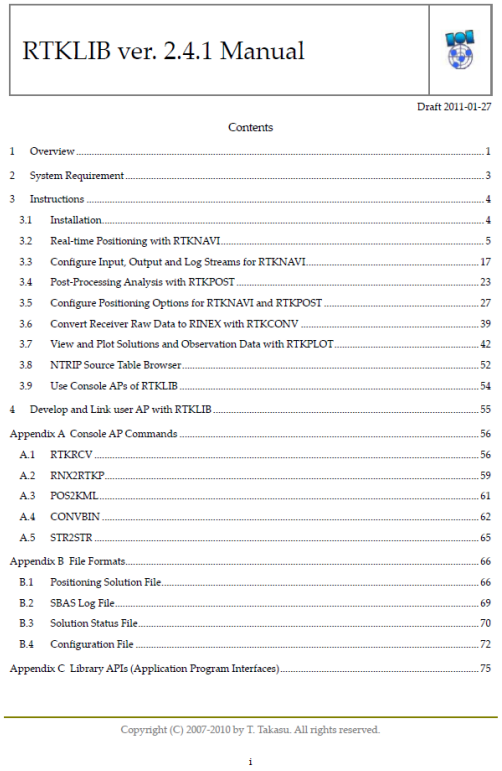
Visibility at Tokyo by RTKPLOT

- GPS (12)
- GLONASS (8)
- Galileo (4)
- QZSS (1)
- BeiDou (10)
- SBAS (4)

Total (39)

(El>10deg)

RTKLIB: References

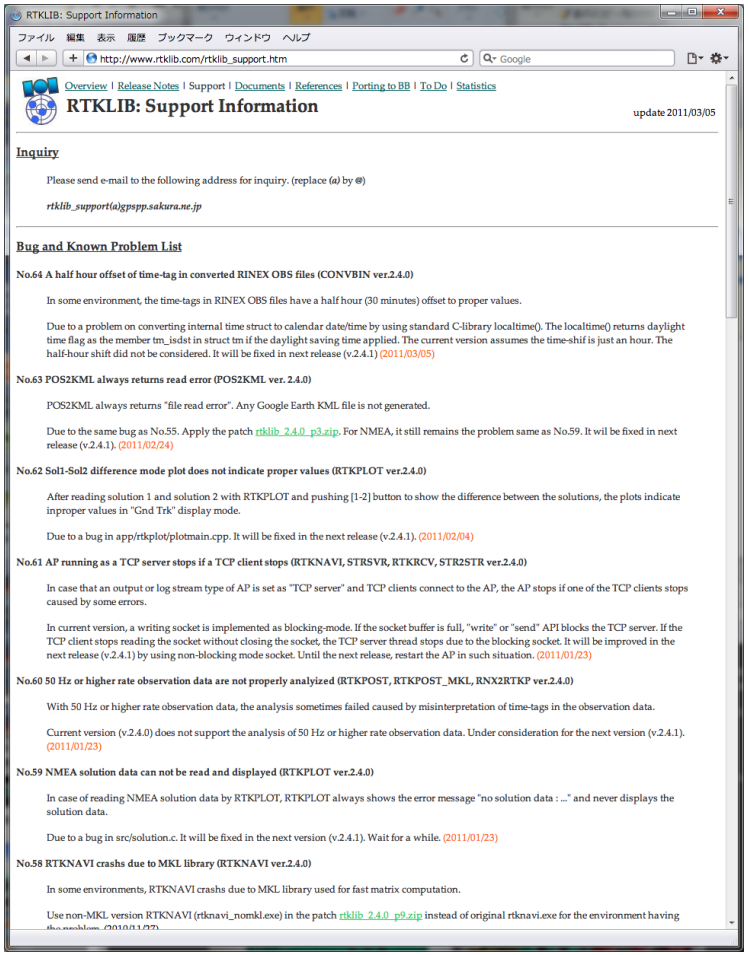


RTKLIB ver. 2.4.1 Manual

Draft 2011-01-27

Contents

rtklib_2.4.2/doc/manual_2.4.2.pdf



RTKLIB: Support Information

ファイル 編集 表示 履歴 ブックマーク ウィンドウ ヘルプ

http://www.rtklib.com/rtklib_support.htm

Google

Overview | Release Notes | Support | Documents | References | Porting to BB | To Do | Statistics

RTKLIB: Support Information

update 2011/03/05

Inquiry

Please send e-mail to the following address for inquiry. (replace *(a)* by @)

rtklib_support(a)gpspp.sakura.ne.jp

Bug and Known Problem List

No.64 A half hour offset of time-tag in converted RINEX OBS files (CONVBIN ver.2.4.0)

In some environment, the time-tags in RINEX OBS files have a half hour (30 minutes) offset to proper values.

Due to a problem on converting internal time struct to calendar date/time by using standard C-library localtime(). The localtime() returns daylight time flag as the member tm_isdst in struct tm if the daylight saving time applied. The current version assumes the time-shif is just an hour. The half-hour shift did not be considered. It will be fixed in next release (v.2.4.1) (2011/03/05)

No.63 POS2KML always returns read error (POS2KML ver. 2.4.0)

POS2KML always returns "file read error". Any Google Earth KML file is not generated.

Due to the same bug as No.55. Apply the patch rtklib_2.4.0_p3.zip. For NMEA, it still remains the problem same as No.59. It wil be fixed in next release (v.2.4.1). (2011/02/24)

No.62 Sol1-Sol2 difference mode plot does not indicate proper values (RTKPLOT ver.2.4.0)

After reading solution 1 and solution 2 with RTKPLOT and pushing [1-2] button to show the difference between the solutions, the plots indicate inproper values in "Gnd Trk" display mode.

Due to a bug in app/rtkplot/plotmain.cpp. It will be fixed in the next release (v.2.4.1). (2011/02/04)

No.61 AP running as a TCP server stops if a TCP client stops (RTKNAVI, STRSVR, RTKRCV, STR2STR ver.2.4.0)

In case that an output or log stream type of AP is set as "TCP server" and TCP clients connect to the AP, the AP stops if one of the TCP clients stops caused by some errors.

In current version, a writing socket is implemented as blocking-mode. If the socket buffer is full, "write" or "send" API blocks the TCP server. If the TCP client stops reading the socket without closing the socket, the TCP server thread stops due to the blocking socket. It will be improved in the next release (v.2.4.1) by using non-blocking mode socket. Until the next release, restart the AP in such situation. (2011/01/23)

No.60 50 Hz or higher rate observation data are not properly analyized (RTKPOST, RTKPOST_MKL, RNX2RTKP ver.2.4.0)

With 50 Hz or higher rate observation data, the analysis sometimes failed caused by misinterpretation of time-tags in the observation data.

Current version (v.2.4.0) does not support the analysis of 50 Hz or higher rate observation data. Under consideration for the next version (v.2.4.1). (2011/01/23)

No.59 NMEA solution data can not be read and displayed (RTKPLOT ver.2.4.0)

In case of reading NMEA solution data by RTKPLOT, RTKPLOT always shows the error message "no solution data : ..." and never displays the solution data.

Due to a bug in src/solution.c. It will be fixed in the next version (v.2.4.1). Wait for a while. (2011/01/23)

No.58 RTKNAVI crashs due to MKL library (RTKNAVI ver.2.4.0)

In some environments, RTKNAVI crashs due to MKL library used for fast matrix computation.

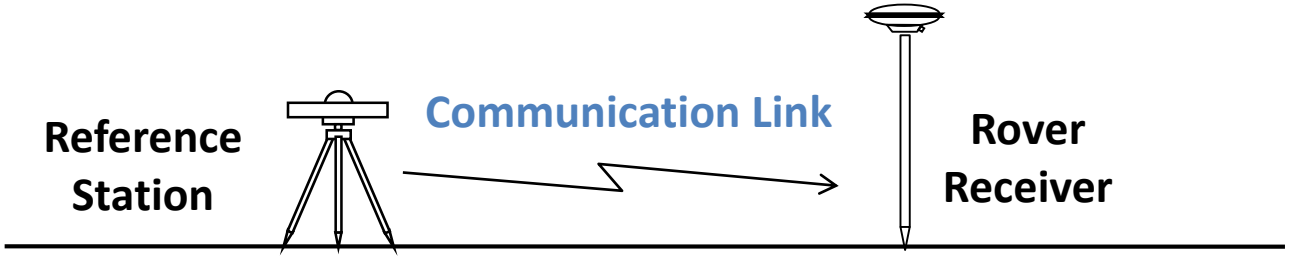
Use non-MKL version RTKNAVI (rtknavi_nomkl.exe) in the patch rtklib_2.4.0_p9.zip instead of original rtknavi.exe for the environment having

http://www.rtklib.com

RTKLIB Practice (1)

- **Install RTKLIB**
- **Setup Receivers and Antennas**
- **Use RTKLIB in Post Processing Mode**
- **RTKLIB in Real-Time Mode**

RTK Practice



- Post processing: Observation and Navigation data are required (RINEX).
- Real-Time: Communication link and differential data reception are required (RTCM/NTRIP).

What if you can't receive correction data for 1 minutes ?

- RTK requires the robust communication link. This is demerit of RTK because continuous data link is sometimes troublesome…
- How long can RTK keep their fixed solutions without correction data ? We can't answer. Some receivers give up 10 seconds or 30 seconds…
- It also depends on the environment.

RTCM

- The standard for differential global navigation satellite system was defined in RTCM Special Committee 104 and its current version is Version 3. RTCM standard for differential global navigation satellite services are communication protocols between reference stations and mobile receivers which allow very high accurate positioning, when compared with positioning system without augmentation.

NTRIP

- The NTRIP was also defined in the RTCM Special Committee 104. NTRIP stands for "Networked Transport for RTCM via Internet Protocol". It is based on Hypertext transfer Protocol version 1.1 and the intention is to disseminate differential correction data through the internet.

https://igs.bkg.bund.de/index/index

Download

- http://www.denshi.e.kaiyodai.ac.jp/kubo/kubo_data.html
- rtklib.zip
- 1h.zip
- 24h.zip
- car.zip

If you need the raw-data used in this presentation, please download the following 4 files.

Install RTKLIB

You can refer to the latest update : https://github.com/tomojitakasu/RTKLIB

u-blox NEO-M8P

NEO-M8P

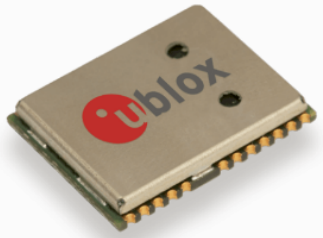
u-blox M8 high precision GNSS modules

ublox

☐ Standard ☑ Professional ☐ Automotive

Highlights

- Centimeter-level GNSS positioning for the mass market
- Integrated Real Time Kinematics (RTK) for fast time-to-market
- Smallest, lightest, and energy-efficient RTK module
- Complete and versatile solution due to base and rover variants
- World-leading GNSS positioning technology



NEO-M8P
12.2 x 16 x 2.4 mm

Product variants

NEO-M8P-0	u-blox M8 high precision module with rover functionality
NEO-M8P-2	u-blox M8 high precision module with rover and base station functionality

Order online | Evaluation Kit

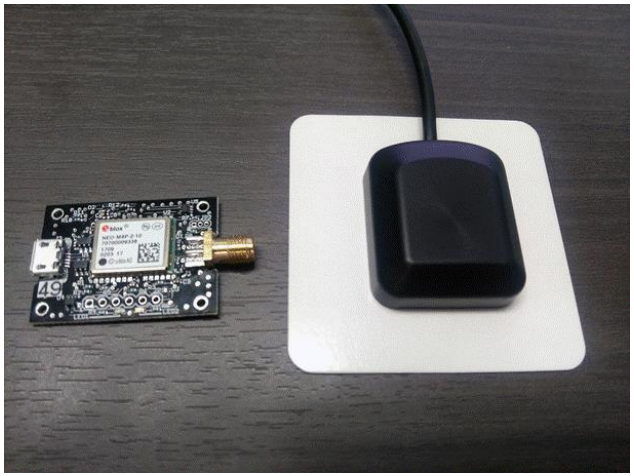
Model	Category				GNSS					Supply	Interfaces				Features								Grade		
	Standard Precision GNSS	High Precision GNSS	Dead Reckoning	Timing	GPS / QZSS	GLONASS	Galileo	BeiDou	Number of Concurrent GNSS	2.7 V – 3.6 V	UART	USB	SPI	DDC (I²C compliant)	Programmable (Flash)	Data logging	Carrier phase output	Additional SAW	Additional LNA	RTK rover	Base station with survey-in	Timepulse	Standard	Professional	Automotive
NEO-M8P-0		•			•	•		•	2	•	•	•	•	•	•	•	•	•	•	•		1			
NEO-M8P-2		•			•	•		•	2	•	•	•	•	•	•	•	•	•	•	•	•	1			

https://www.u-blox.com/en/product/neo-m8p

Setup u-blox Receiver/u-center

- **Install Support S/W to your laptop PC**
  - **u-blox u-center**
    (u-centersetup_v8.2*.zip)

u-blox NEO-M8P-2
Mini-EVK Card + antenna



u-blox
u-center v8.26

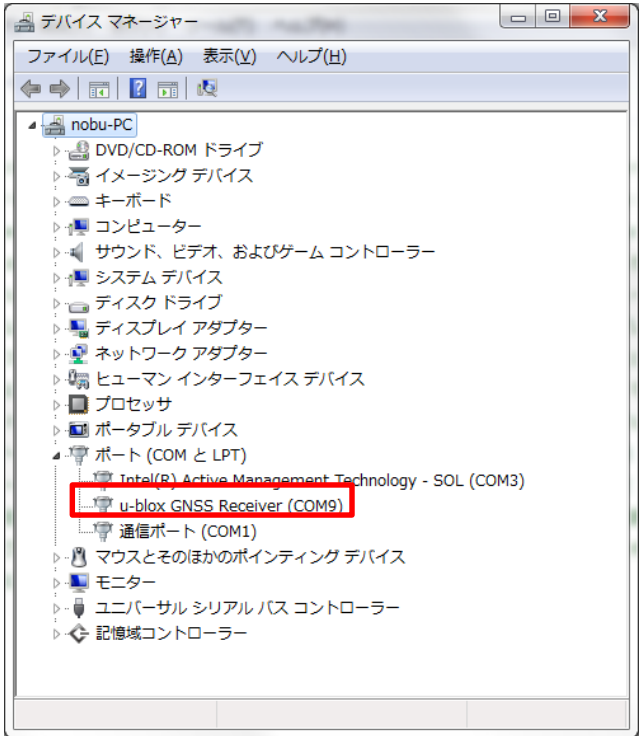
micro
USB

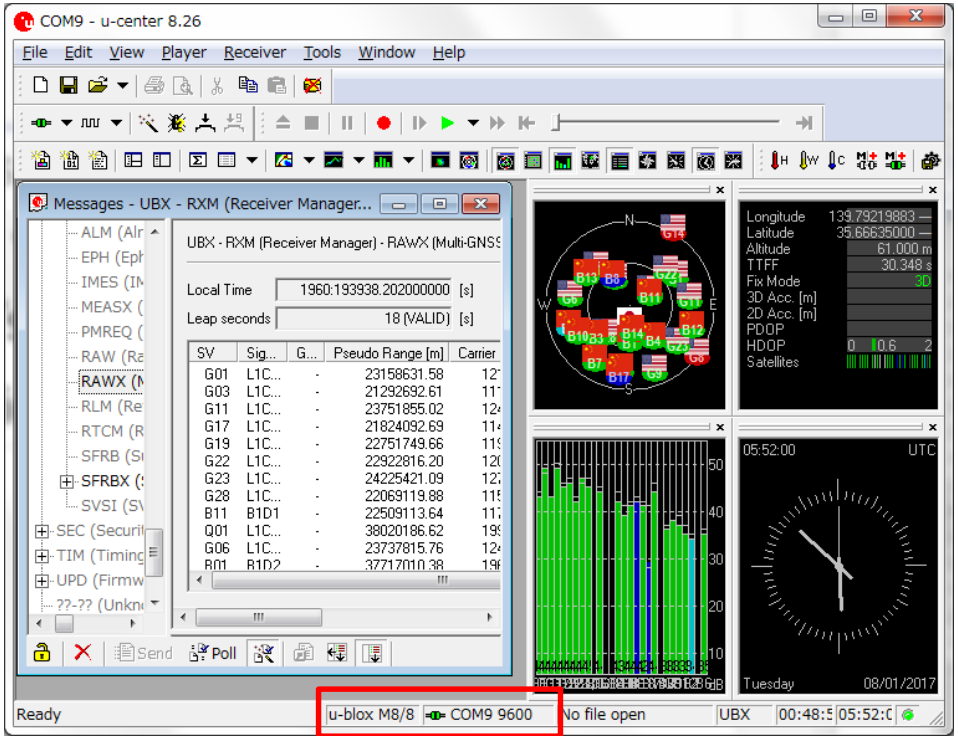
your Laptop PC

Which is good ?

Connection

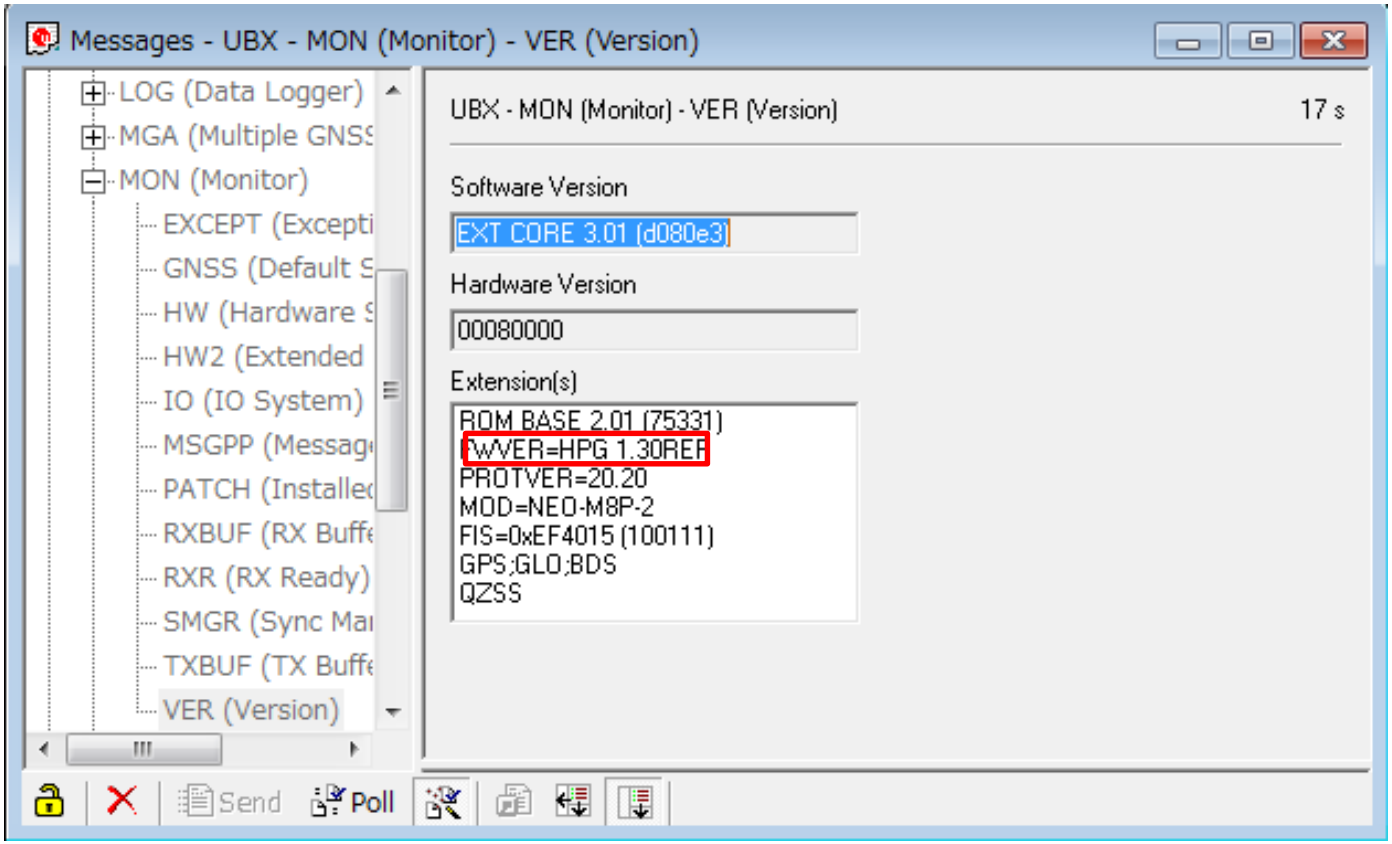
- Start u-center and select Receiver->Port
- You will see COM*. You can check your COM port in your laptop’s devise manager
- If you have any difficulties(win10), please catch us.





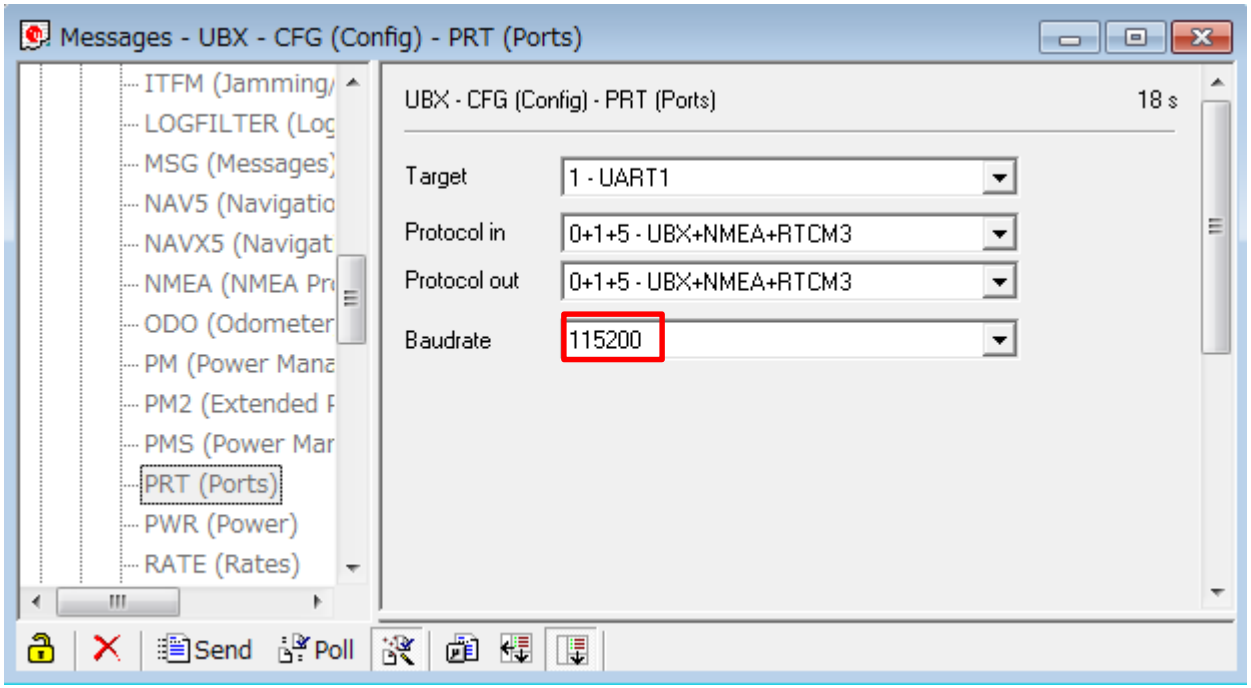
Firmware Check

- View->Message View->UBX->MON->VER



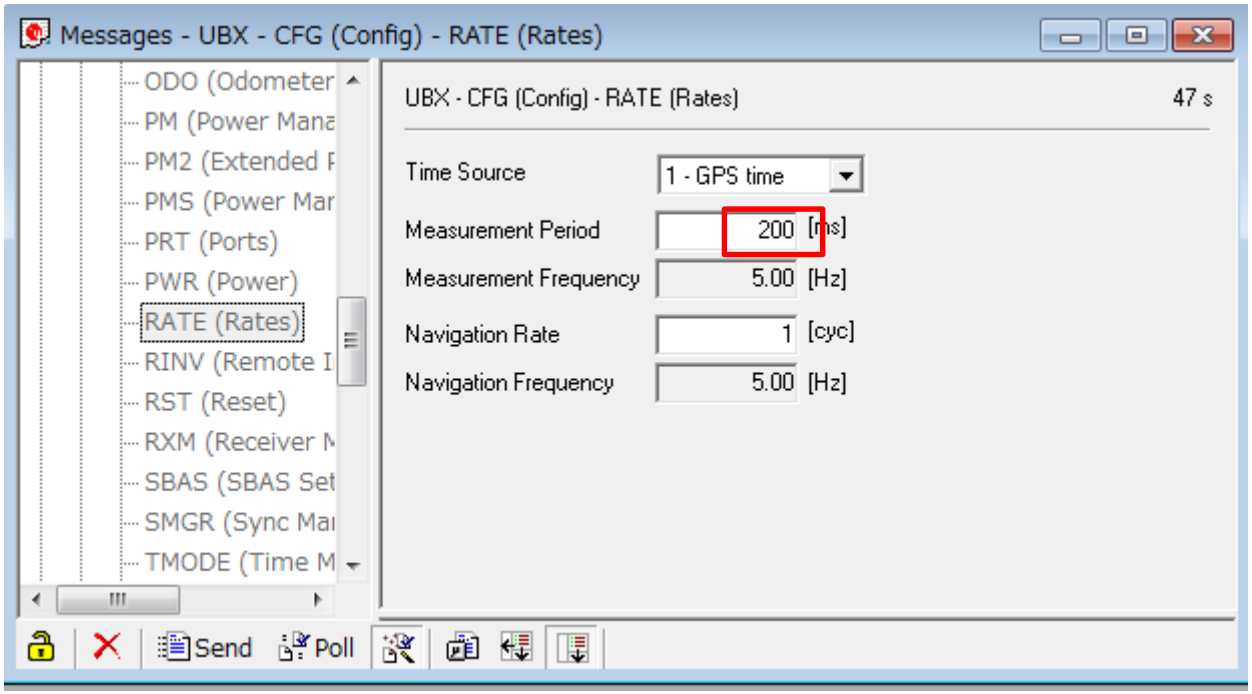
Baud rate Check

- View->Message View->UBX->CFG->PRT
- Please change from 9600 to 115200



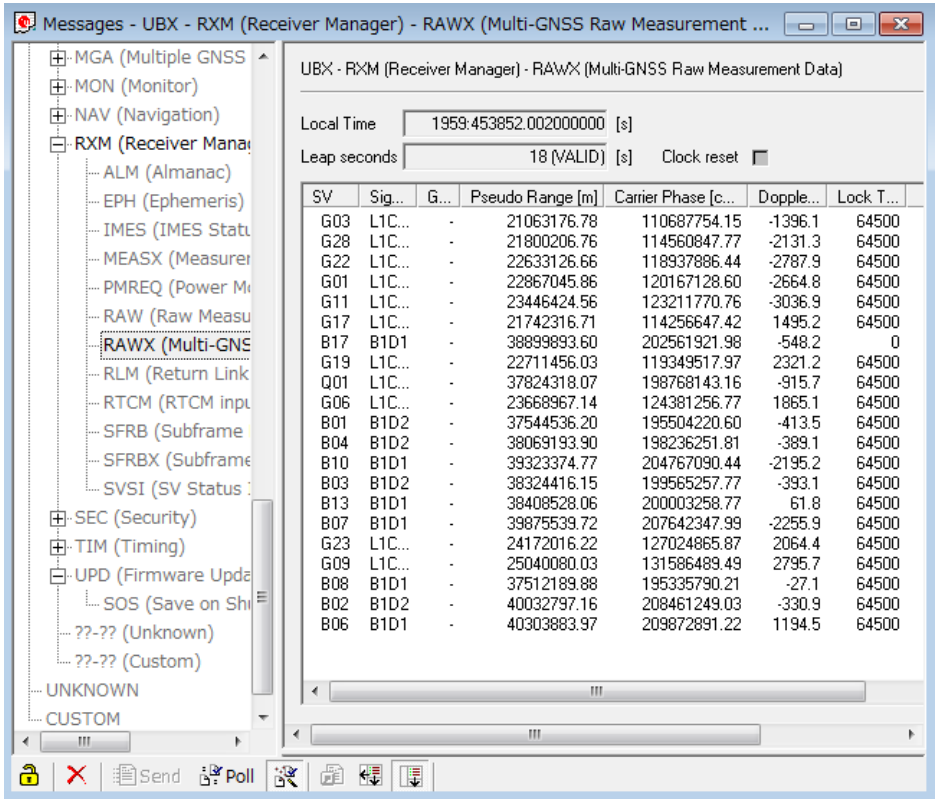
If you want to change the rate…

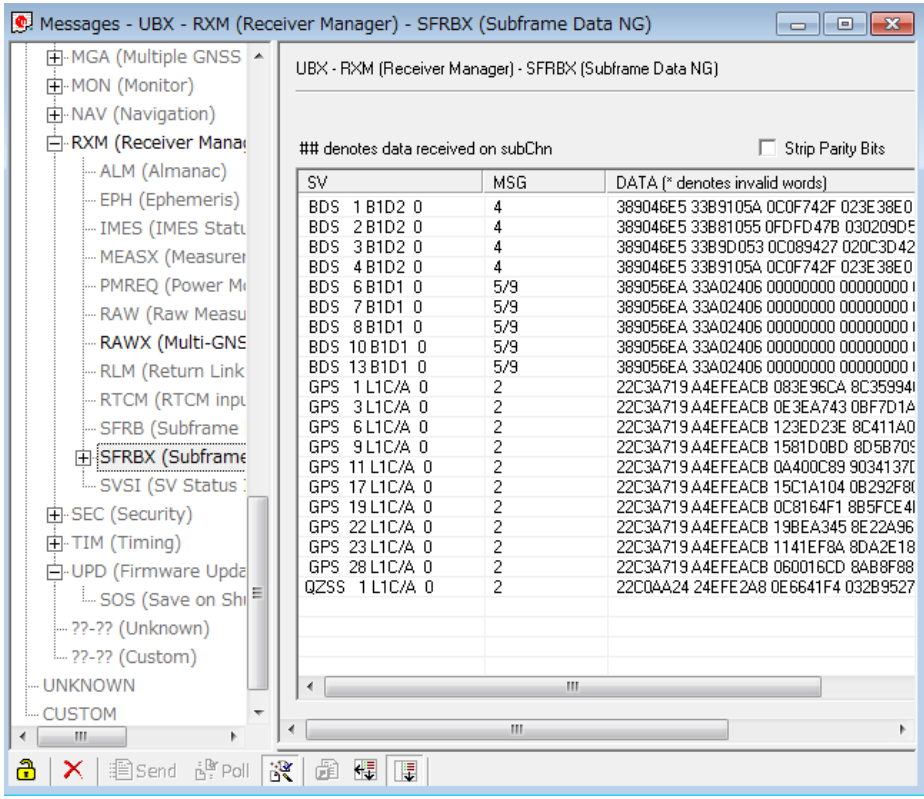
- View->Message View->UBX->CFG->RATE
- Please change the measurement period



If you want to save the raw-data…

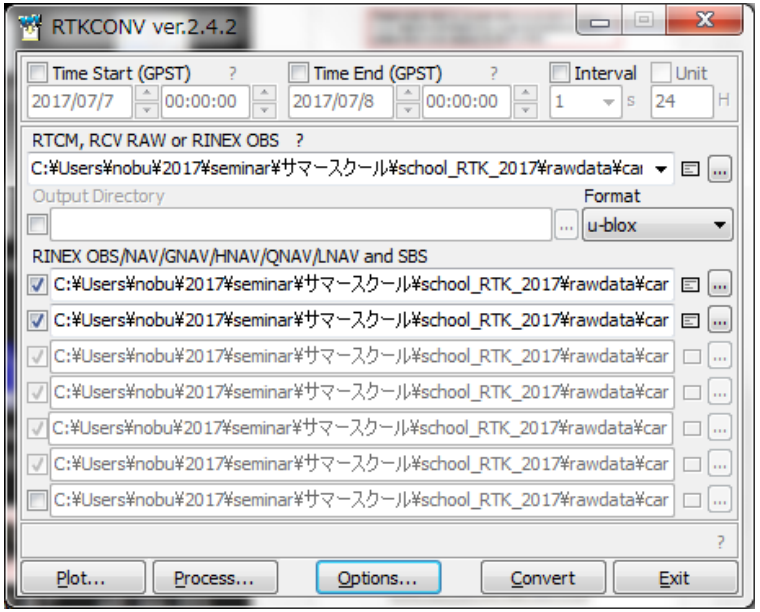
- View->Message View->UBX->RXM->”RAWX” and “SFRBX”
- Right click->Enable Message

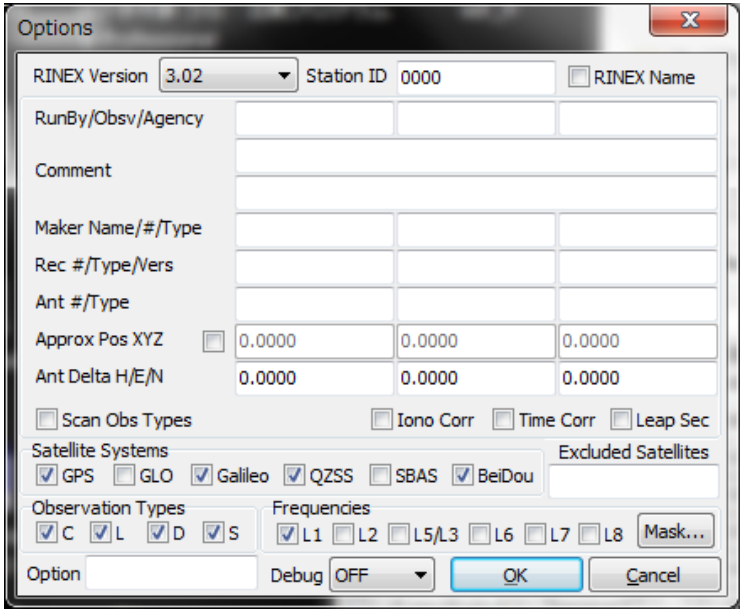




RTKCONV

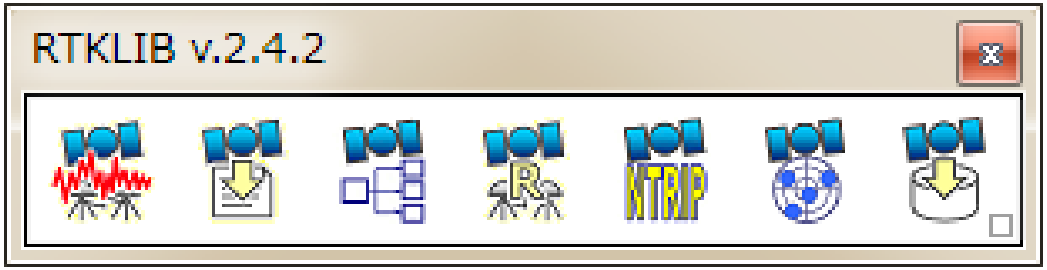
- When you post-process of GNSS raw data, RINEX format is quite popular.
- You can convert other receiver's raw data to RINEX format using rtkconv.exe.
- In the case of Trimble T02 file, you can use "Convert To RINEX" which is available in the Trimble website.





Use RTKLIB (1)

- **Execute RTKLAUNCH.**
  RTKLIB_bin-master¥bin¥rtklaunch.exe

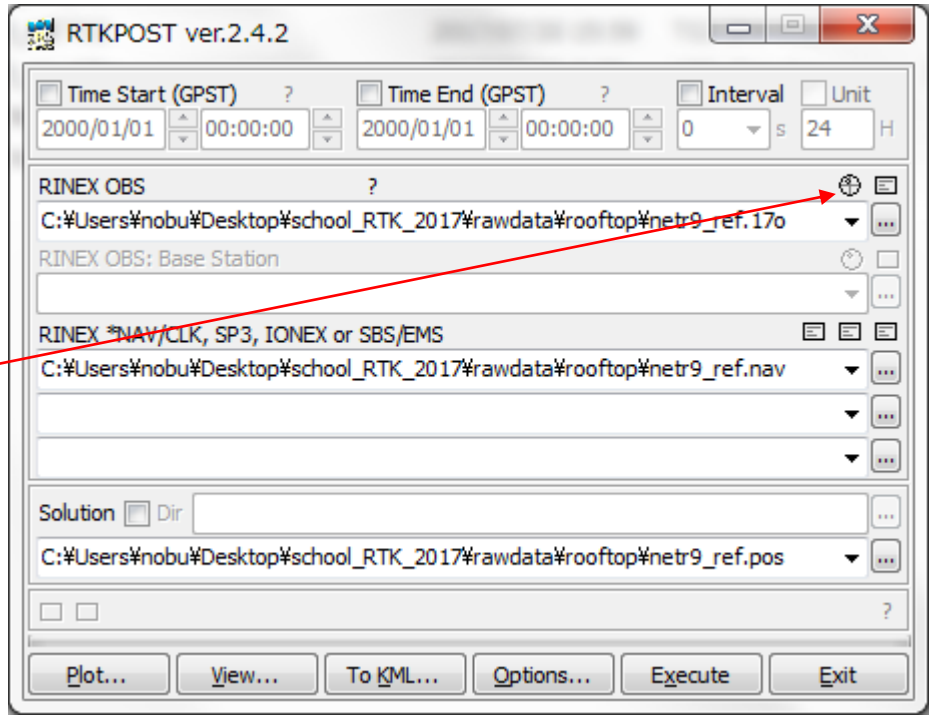


RTKPLOT　STRSVR　NTRIPBRS　RTEGET

RTKCONV　RTKPOST　RTKNAVI

Use RTKLIB (2)

- **Execute RTKPOST by RTKLAUNCH**
- **Execute Menu of RTKPLOT:**
  rawdata¥rooftop¥
  netr9_ref.17o and netr9_ref.nav
- **Click here**



1h data was obtained on 20th July 2017 using NetR9 on the rooftop.
2017/7/20 5:00:00-5:59:59 (GPST)

Use RTKLIB (3)

Satellite ID

G: GPS

R: GLO

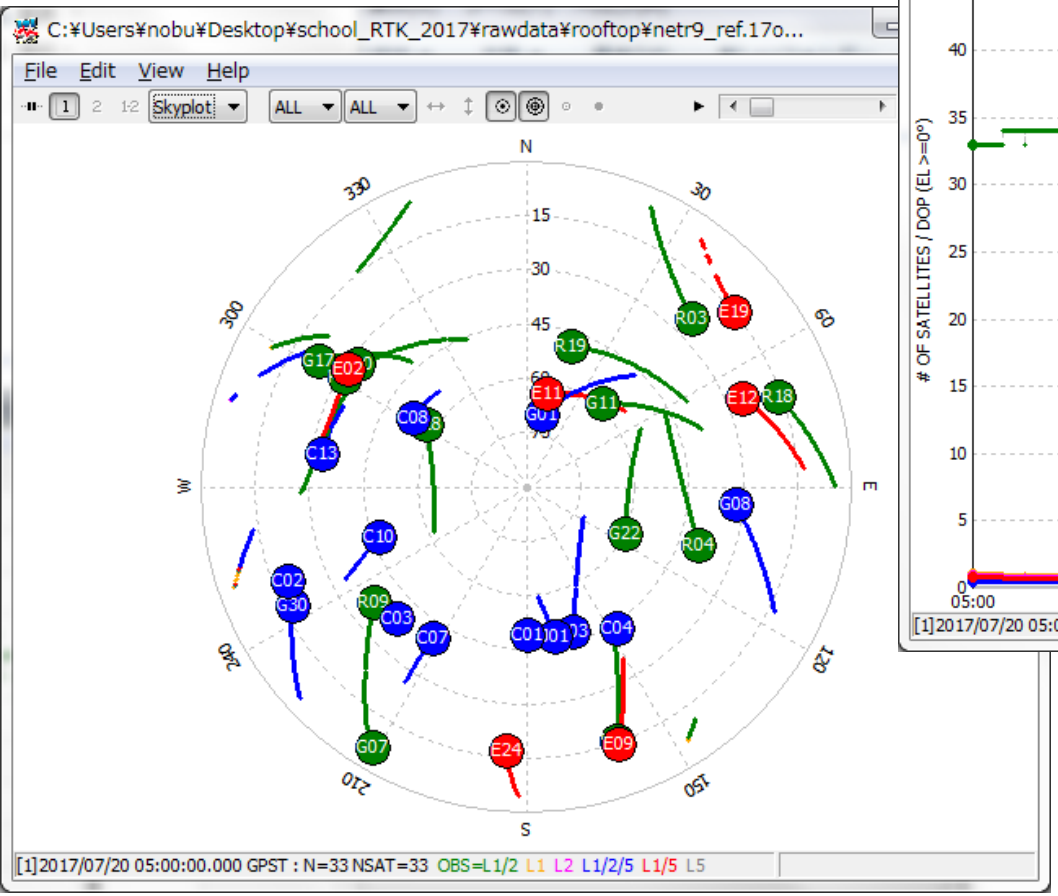
E: GAL

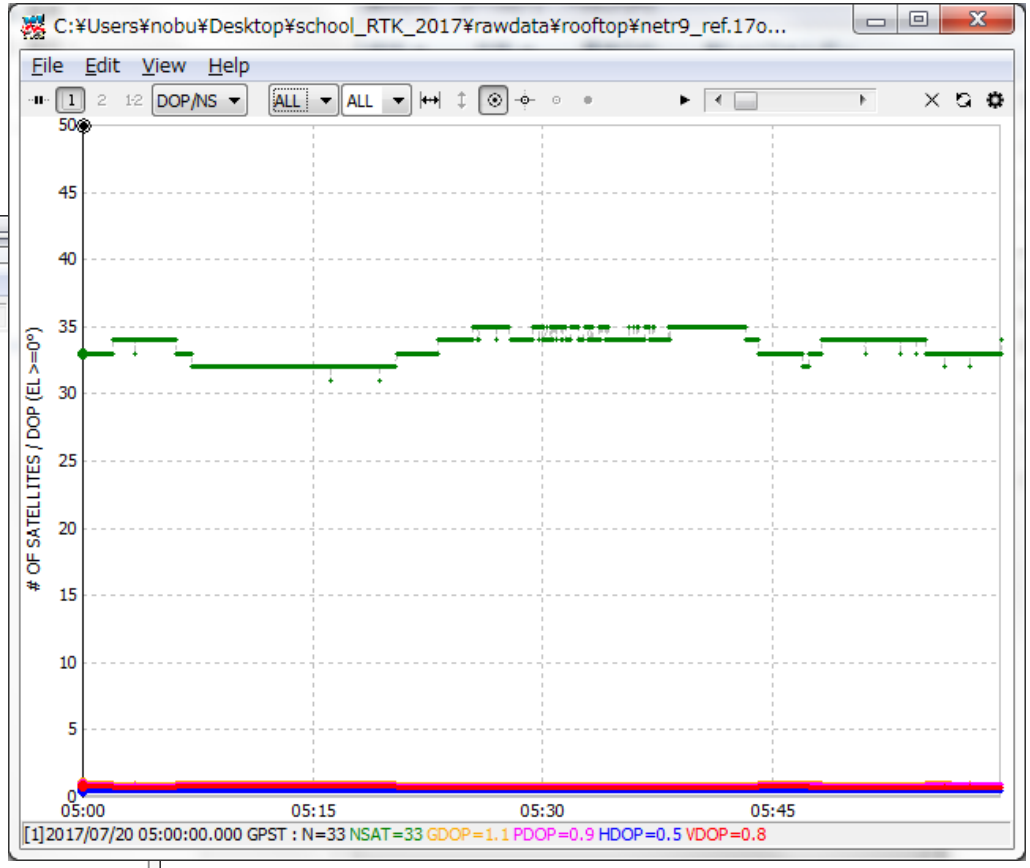
J: QZS

C: BEI

Use RTKLIB (4)

Skyplot

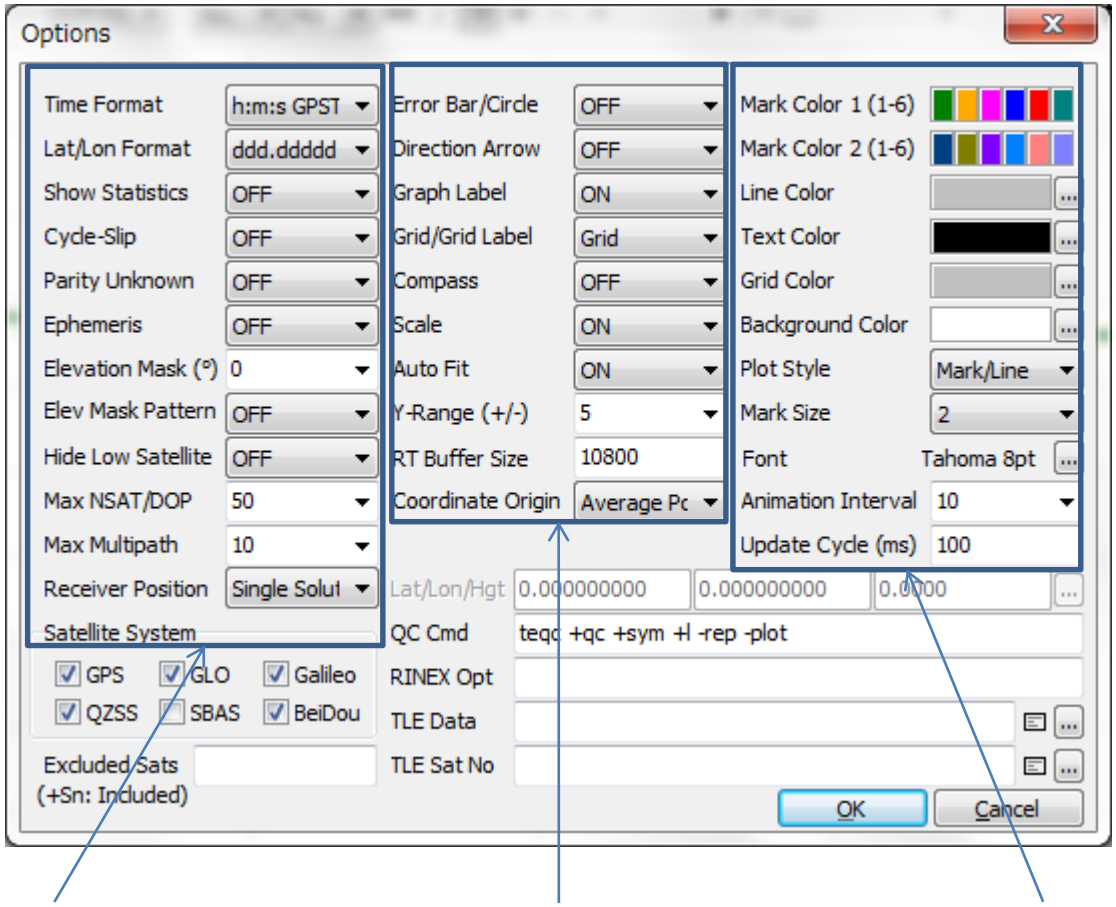




# of Visible Satellites and DOP

Use RTKLIB (5)

RTKPLOT - Options

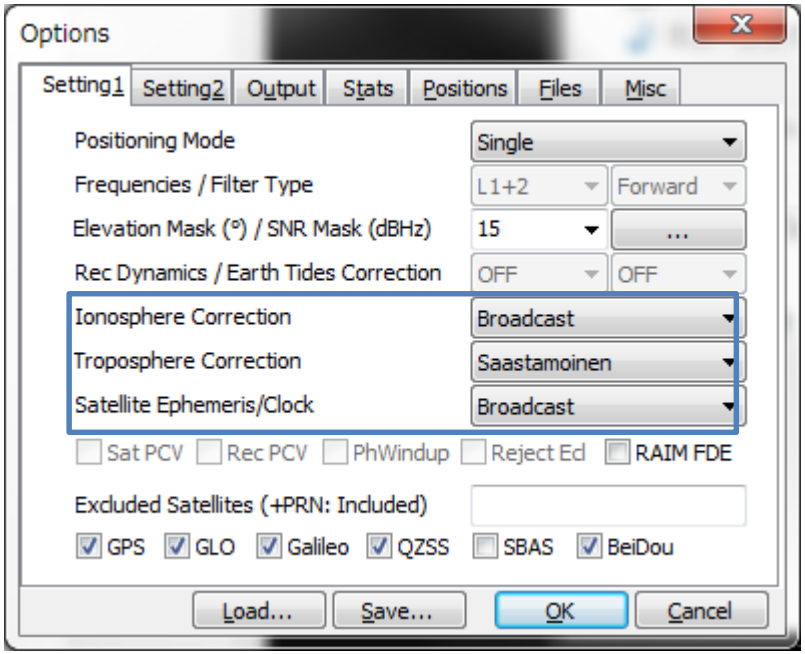


OBS Data Options **Solution Data Options** **Common Options**

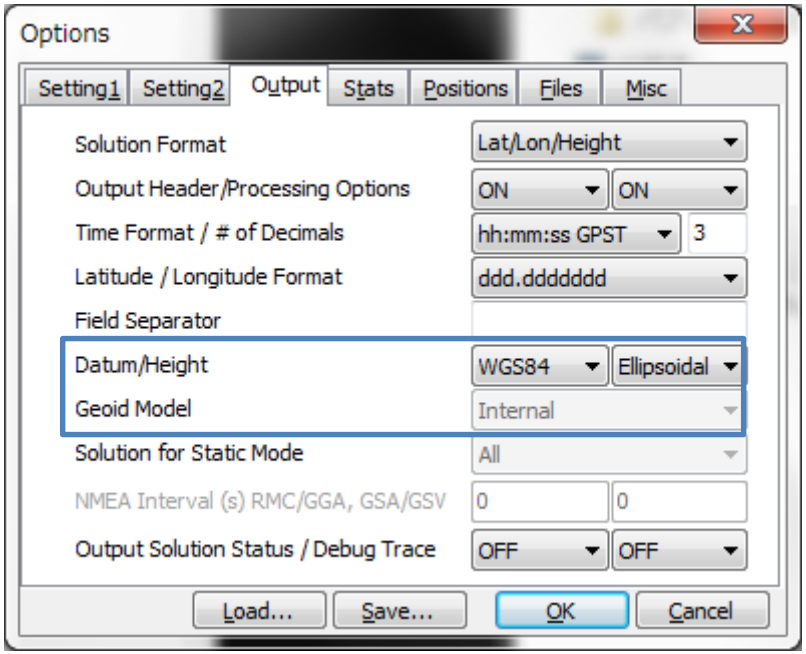
Use RTKLIB (6)

RTKPOST - Options

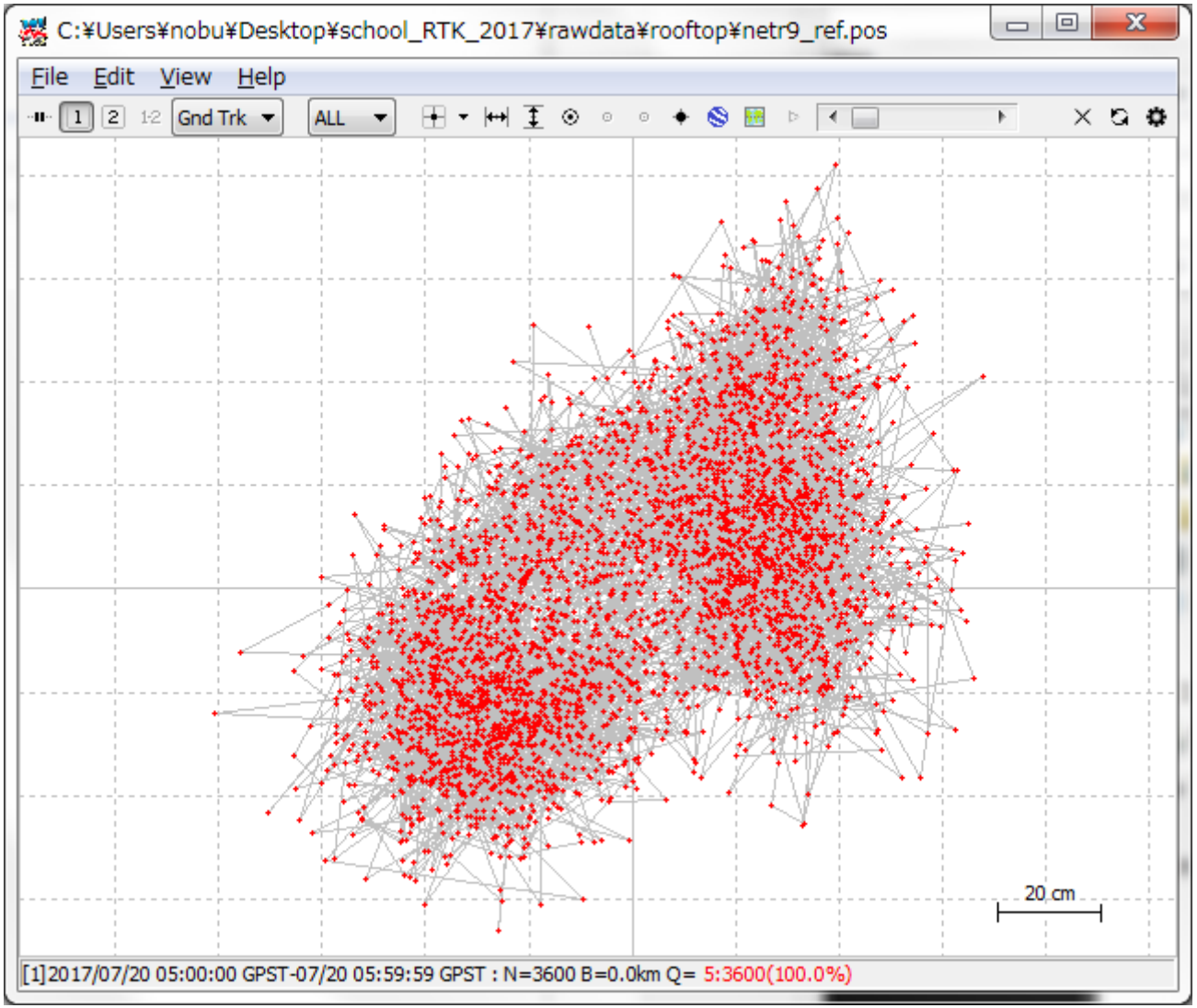
Setting1



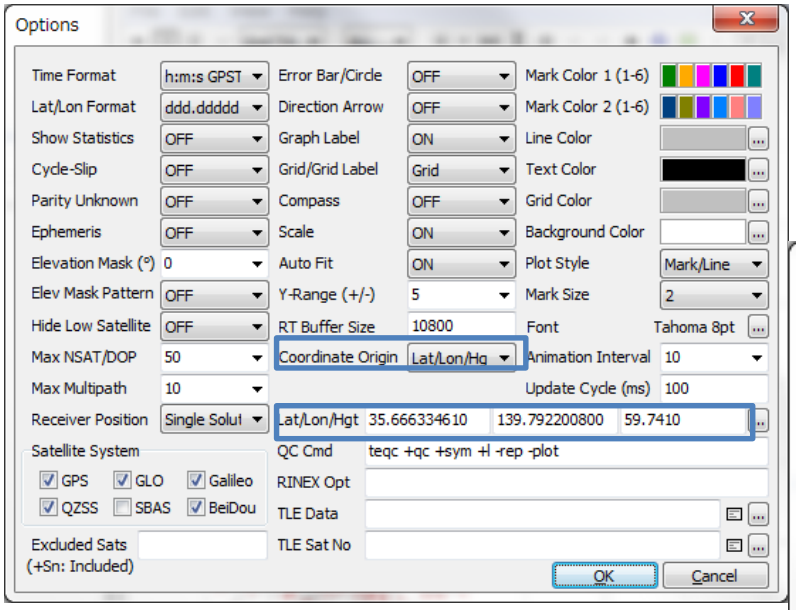
Output



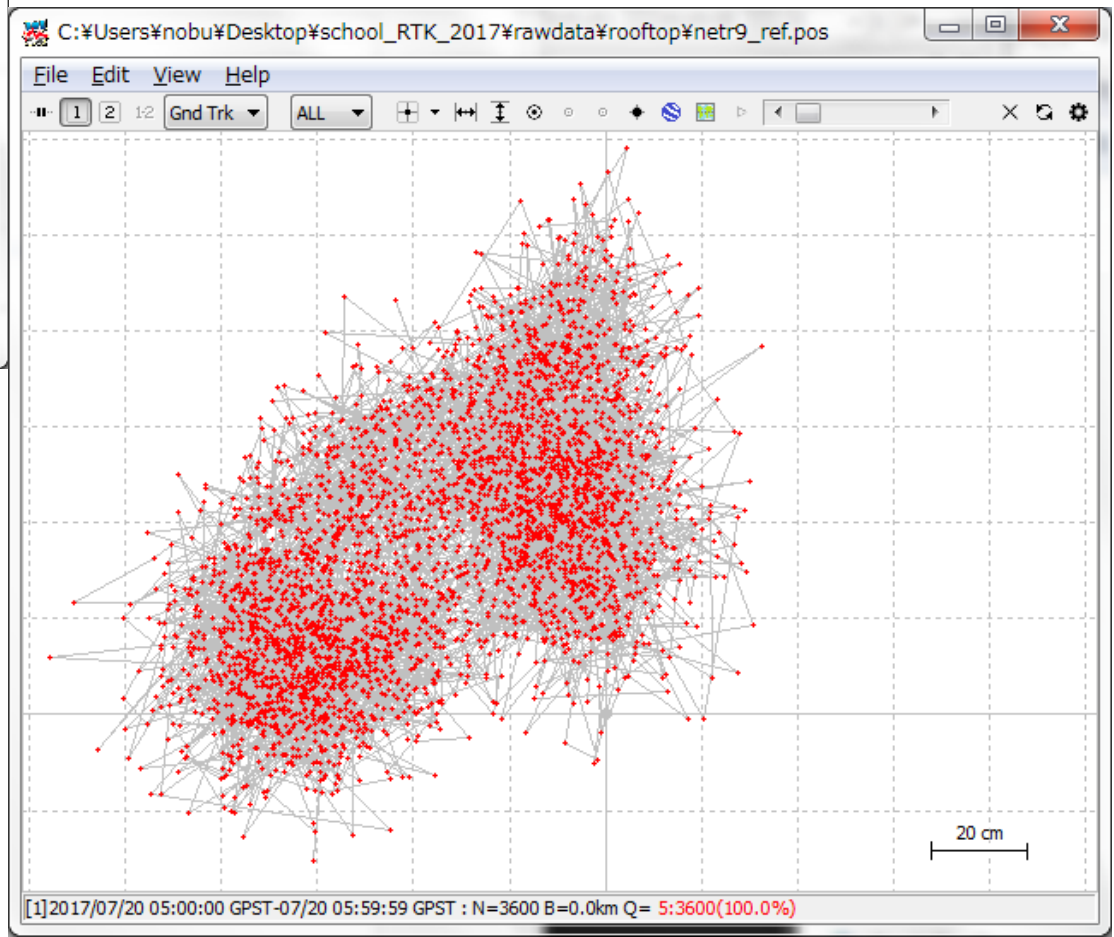
Single Point Positioning



Coordinate Origin



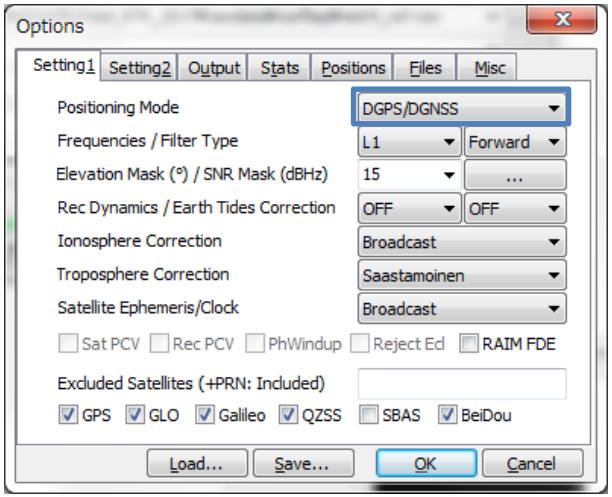
If you change the coordinate origin as a precise reference position, (35.66633461, 139.7922008, 59.741) you see bias like below.

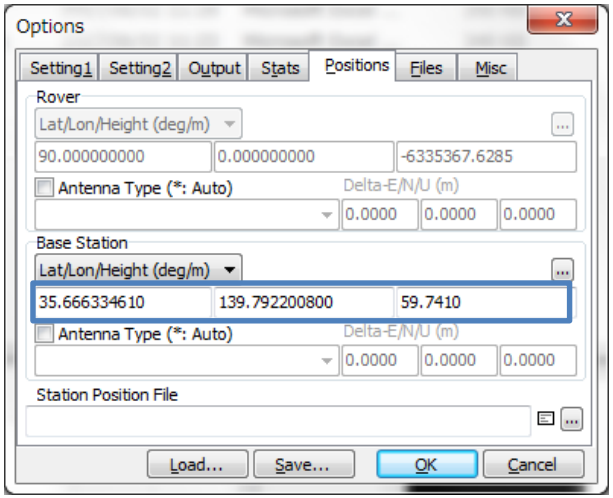


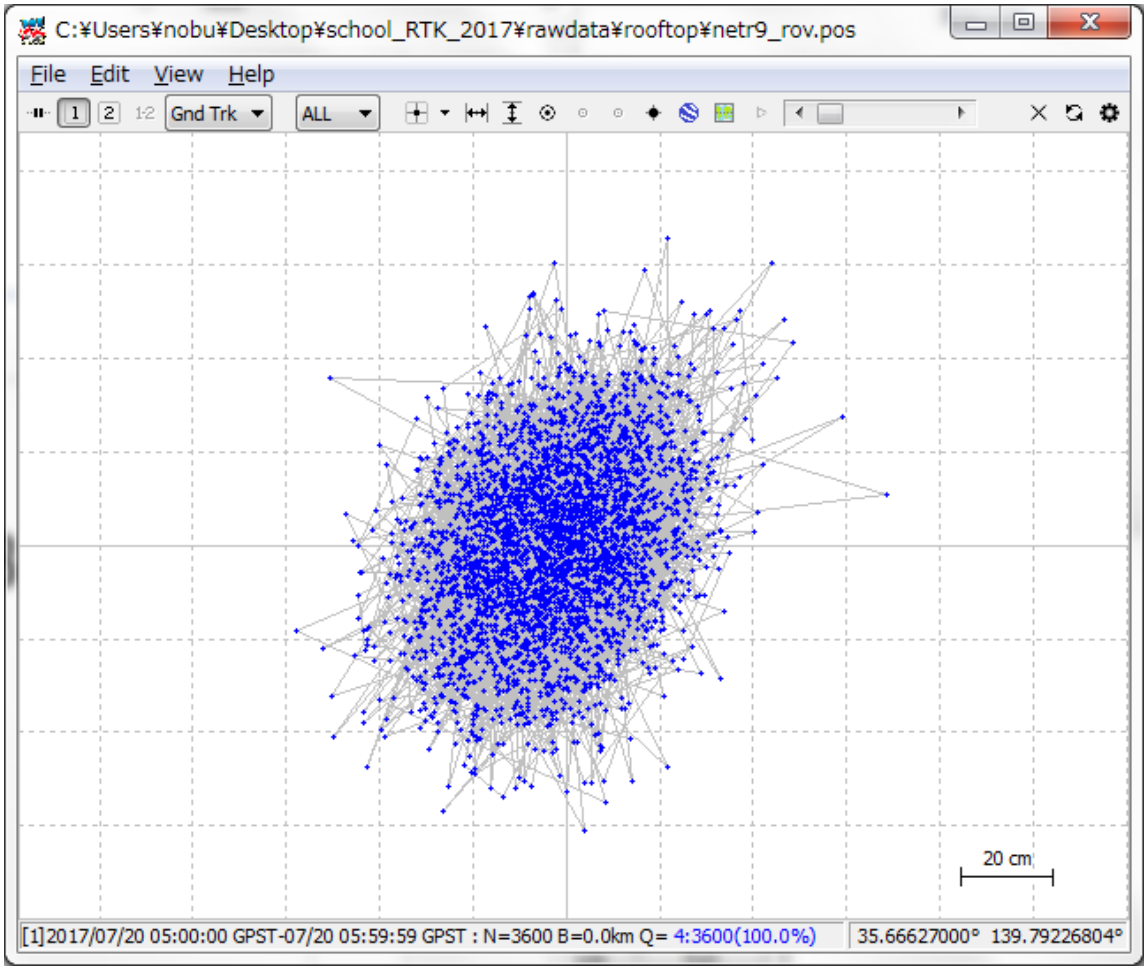
DGNSS

- Precise rover position(LAT/LON/HGT):

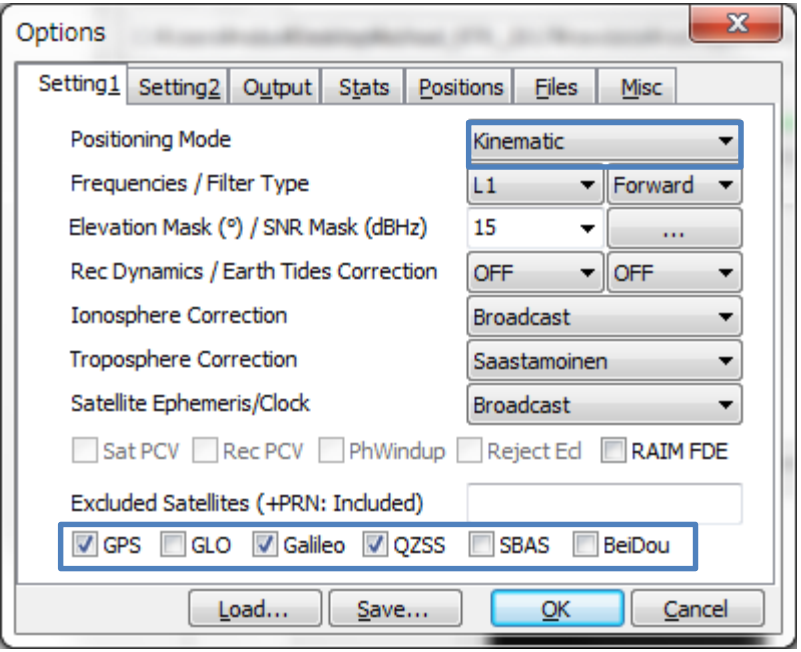
  35.66627025　139.79226723　59.33

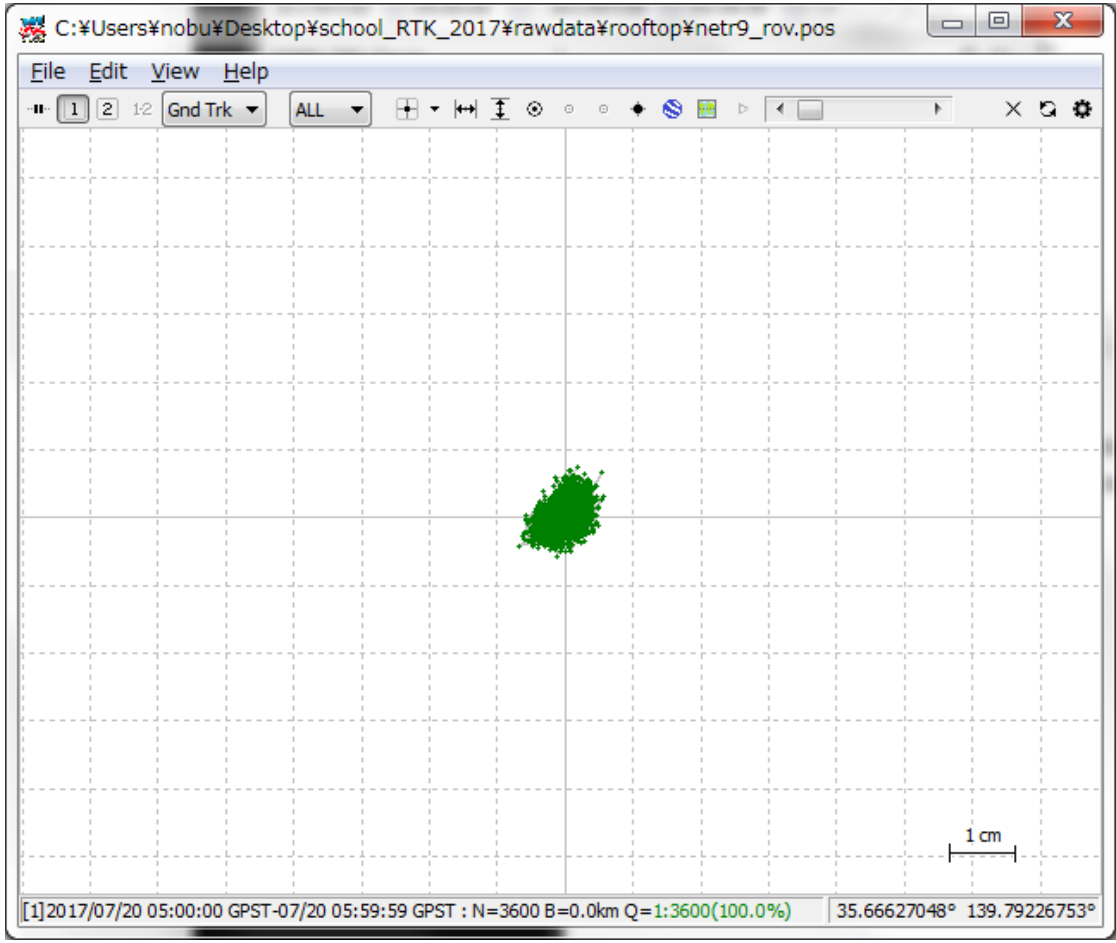






RTK-GNSS


Options
Setting1
Setting2
Output
Stats
Positions
Files
Misc
Positioning Mode
Kinematic
Frequencies / Filter Type
L1
Forward
Elevation Mask (°) / SNR Mask (dBHz)
15
...
Rec Dynamics / Earth Tides Correction
OFF
OFF
Ionosphere Correction
Broadcast
Troposphere Correction
Saastamoinen
Satellite Ephemeris/Clock
Broadcast
Sat PCV
Rec PCV
PhWindup
Reject Ed
RAIM FDE
Excluded Satellites (+PRN: Included)
GPS
GLO
Galileo
QZSS
SBAS
BeiDou
Load...
Save...
OK
Cancel


C:¥Users¥nobu¥Desktop¥school_RTK_2017¥rawdata¥rooftop¥netr9_rov.pos
File
Edit
View
Help
Gnd Trk
ALL
1 cm
[1]2017/07/20 05:00:00 GPST-07/20 05:59:59 GPST : N=3600 B=0.0km Q=1:3600(100.0%)
35.66627048° 139.79226753°