GNSS Logger Unit with RTKLIB

Tiphat Areeyapinun

Shizuoka University

Contents

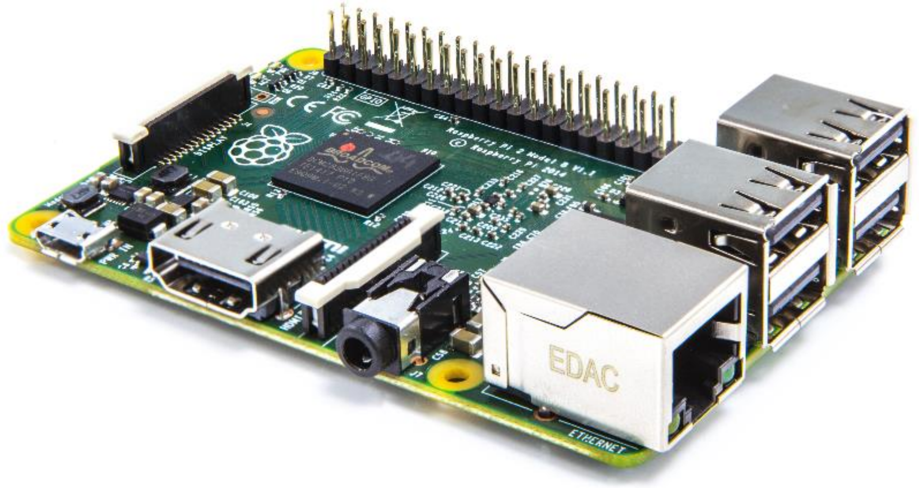
GNSS Logger Unit with RTKLIB

Objective

- To record the GNSS data directly from module for post processing
- Small footprint and easy to use
- Use RTKLIB as base software

Why we choose Raspberry PI?

- Mainstream product
- Easy to use and maintenance
- RTKLIB, Just compile and GO!!
- A lot of integrate module available in Market



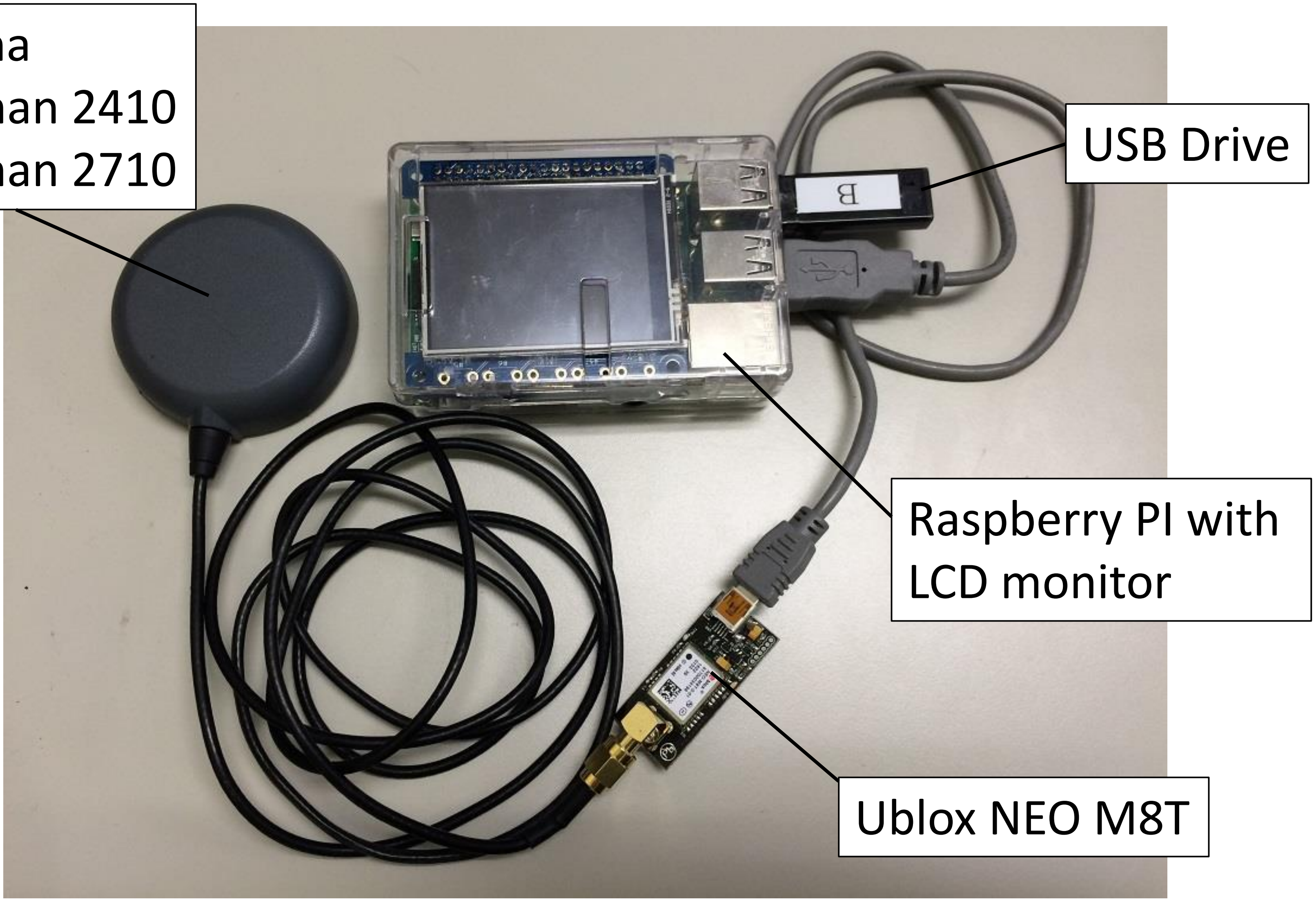
Why we choose Ublox M8T?

- RAW output available (RAWX and SFRBX)
- Reasonable price
- Multiple GNSS support

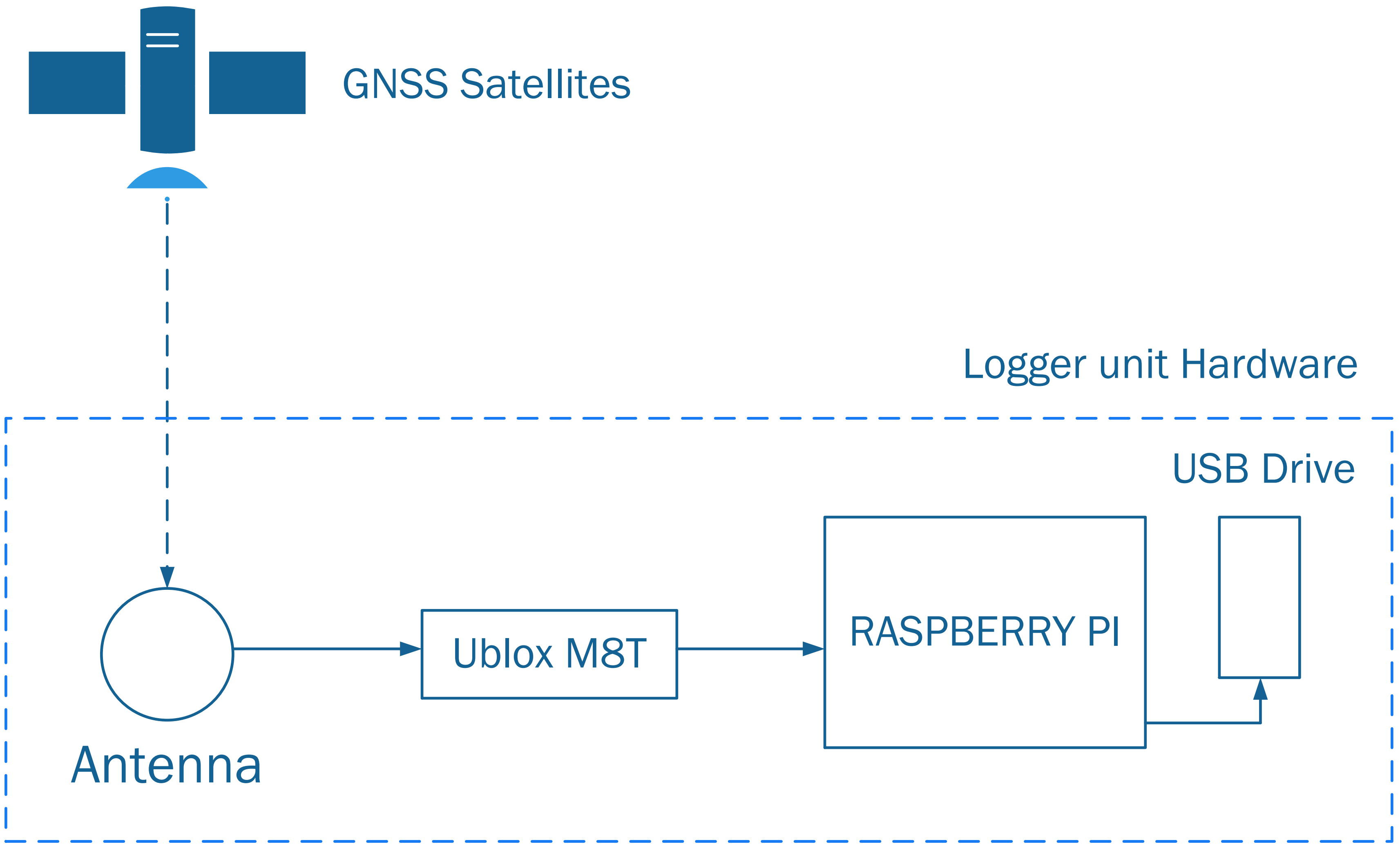
Complement of Logger

- Raspberry Pi (B+, 2 and 3)
- Ublox M8T
- LCD Monitor
- USB cable
- Antenna
- USB Drive (Fat32)

Complement of Logger (2)



Logger Data Flow



Logger function with STR2STR

- This logger unit use STR2STR, one of the application on RTKLIB. Its available on windows as STRSVR
- By the function of STR2STR, we record data directly from receiver to file (store over USB drive)

How to build?

How to build

Narrow down to 3 step

1. Raspberry Pi preparation
2. Receiver preparation
3. Software preparation

Raspberry Pi preparation

- Any version of Raspberry Pi can be use in this project. (B+, 2 and 3)
- In order to gain OS performance, newer version of board is better. (Model B+ use single core 700MHz compare to model 2 and 3 quad core at higher speed)
- But you need to be careful about environment temperature of your experiment, by newer CPU, generate more temperature. This can cause system overheat and auto shutdown
- Comparison of Raspberry Pi available on next slide

Raspberry Pi preparation

Generation	B+	2	3
Release date	July, 2014	February, 2015	February, 2016
Price	25 USD	35 USD	35 USD
Architecture	ARMv6 (32-bit)	ARMv7 (32-bit)	ARMv8 (64/32-bit)
CPU	Single core 700MHz	Quad core 900MHz	Quad core 1200MHz
Memory	512MB	1GB	
USB	4 Ports		
Network	10/100 Lan Port		10/100 Lan Port, Wifi
Power	600mA (3W)	800mA (4W)	1400mA (7W)

https://ja.wikipedia.org/wiki/Raspberry_Pi

- We recommend you to use Raspberry Pi 2 by balance of performance and consumption power
- This model capable to modify to Navigation unit as well

Raspberry Pi preparation

- Tested Operating system that can run RTKLIB are Raspbian and Ubuntu Mate
- Install the OS on SD card (8GB or over)
- Follow the default step of installation

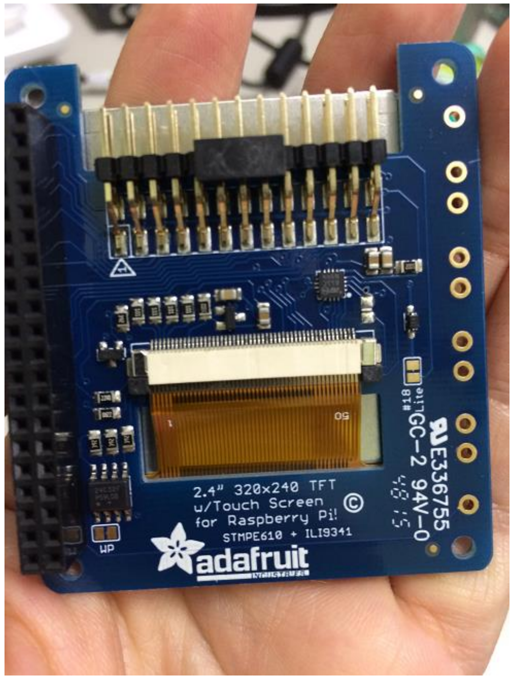
Raspberry Pi preparation

- You can install the board on any case. On this project we used clear case

Case for Raspberry Pi 3 Model B/ Raspberry Pi 2 Model B/ Pi Model B+
https://www.amazon.co.jp/dp/B01CDUM3D6/ref=pe_492632_227730602_TE_item
Price at 790Yen/case

Raspberry Pi preparation

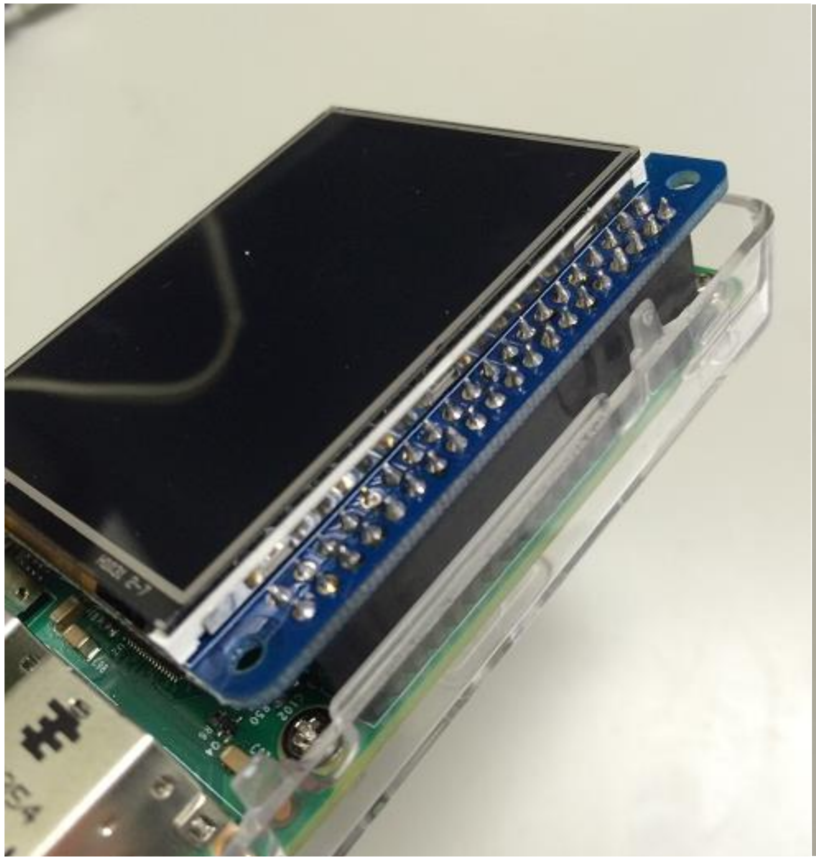
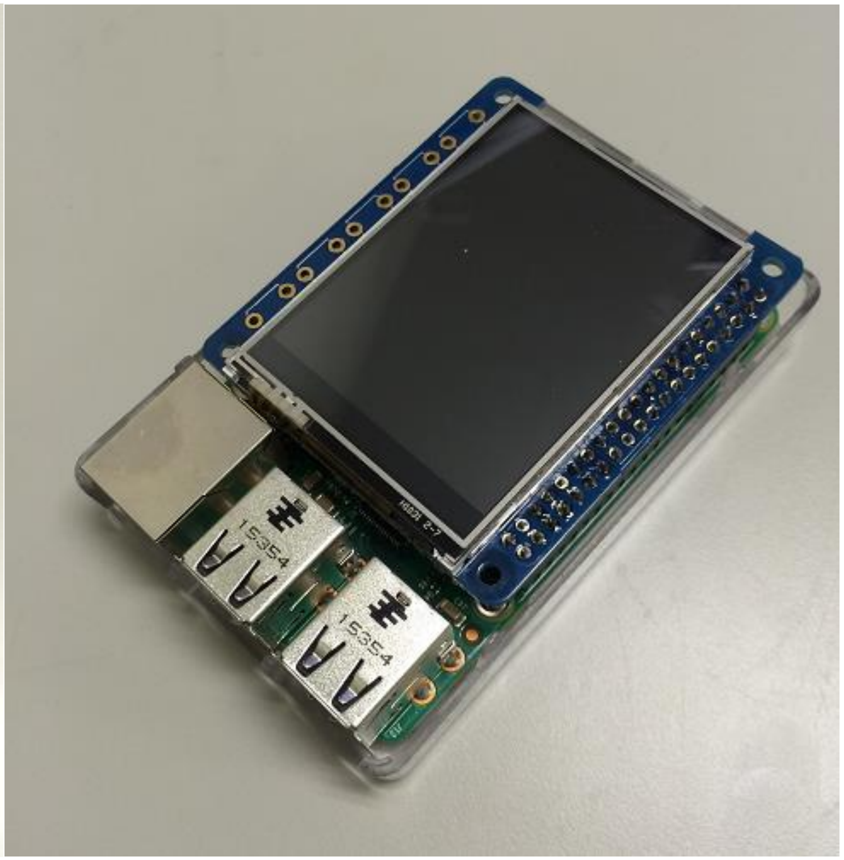
- LCD Screen for display information, we used Adafruit 2.4” 320x240 TFT
- Please soldering 40 female pin to LCD board for connect to Raspberry Pi



https://www.amazon.co.jp/dp/B019IBEMK0
Price at 5527yen/unit

http://www.marutsu.co.jp/pc/i/574345/
Price at 4350yen + tax

Raspberry Pi preparation

- Connect LCD board to Raspberry Pi board via 40 GPIO Pin

Receiver preparation

- Check the firmware of M8T, update to the new version (Current 3.01). For make receiver support Galileo satellite
- In this project we use 4 GNSS satellites. GPS, QZSS, Beidou and Galileo

UBLOX NEO-M8T TIME & RAW receiver board with SMA (RTK ready)
http://www.csgshop.com/product.php?id_product=205 Price at 74.99USD/Unit

Receiver preparation

- The setting in receiver depend on what data you want to output and record
- In our project we use those setting below
  - NMEA and RAW output via USB
  - GPS, Galileo, QZSS and Beidou
  - 5Hz data output

Software preparation

- Pre OS configuration
- RTKLIB software installation
- CLI mode configuration
- LCD monitor setup
- USB drive configuration
- Auto-start script configuration
- Trigger OS shutdown by USB

Software preparation

Pre Operating System configuration

- Expand the disk file system
  - Open Menu → Preferences → Raspberry Pi Configuration
  - Select "Expand Filesystem" then restart the system

Software preparation

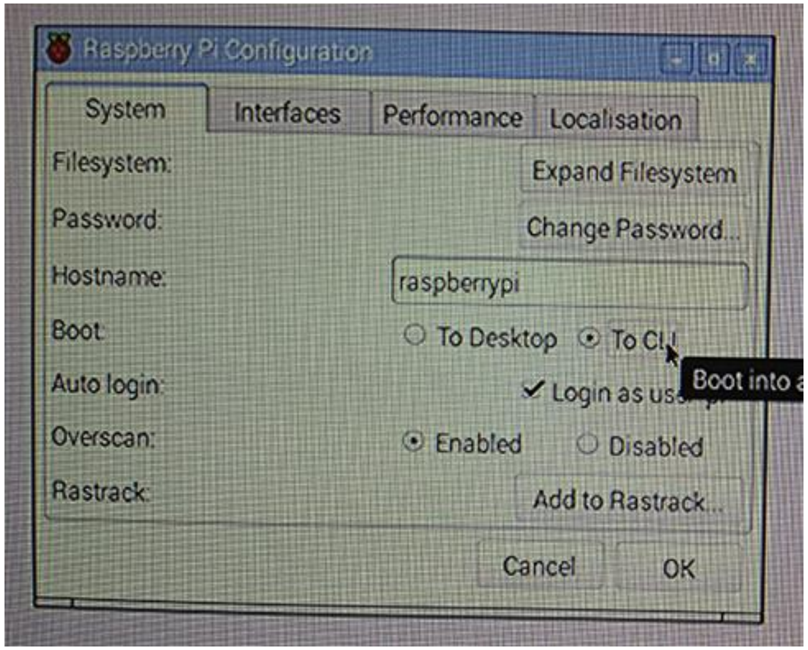
RTKLIB software installation

- Download RTKLIB software from GITHUB (2.4.3 or greater)
- Unzip and go to “RTKLIB-Master¥app” directory
- Execute “sudo chmod 755 makeall.sh” command
- Execute “sudo ./makeall.sh” and wait until it complete

Software preparation

CLI mode configuration

- Open Menu → Preferences → Raspberry Pi Configuration
- From the "System" tab, you can simply click the radio button next to "To CLI" to change the boot preference.
- Then reboot to CLI



Filesystem:
Password:
Hostname: raspberrypi
Boot: To Desktop [✓ To CLI]
Auto login: ✓ Login as us
Overscan: [✓ Enabled] Disabled
Rastrack:

Software preparation

LCD Monitor Setup

- At console prompt, install new kernel by type commands below

```
$curl -SLs https://apt.adafruit.com/add-pin | sudo bash
$sudo apt-get install raspberrypi-bootloader
$sudo apt-get install adafruit-pitft-helper
```

Software preparation

<u>LCD Monitor Setup (2)</u>

- Enable & Configure the PiTFT

```
$sudo adafruit-pitft-helper -t 28r
```

- At the end you will be prompted on whether you want the text console to appear on the LCD Screen. Answer Y to continue.

Software preparation

USB Drive Configuration

- Prepare USB drive using FAT32 file format
- Create mount point on Raspberry Pi by those commands below

```
$ sudo mkdir /media/usb
$ sudo chown -R pi:pi /media/usb
```

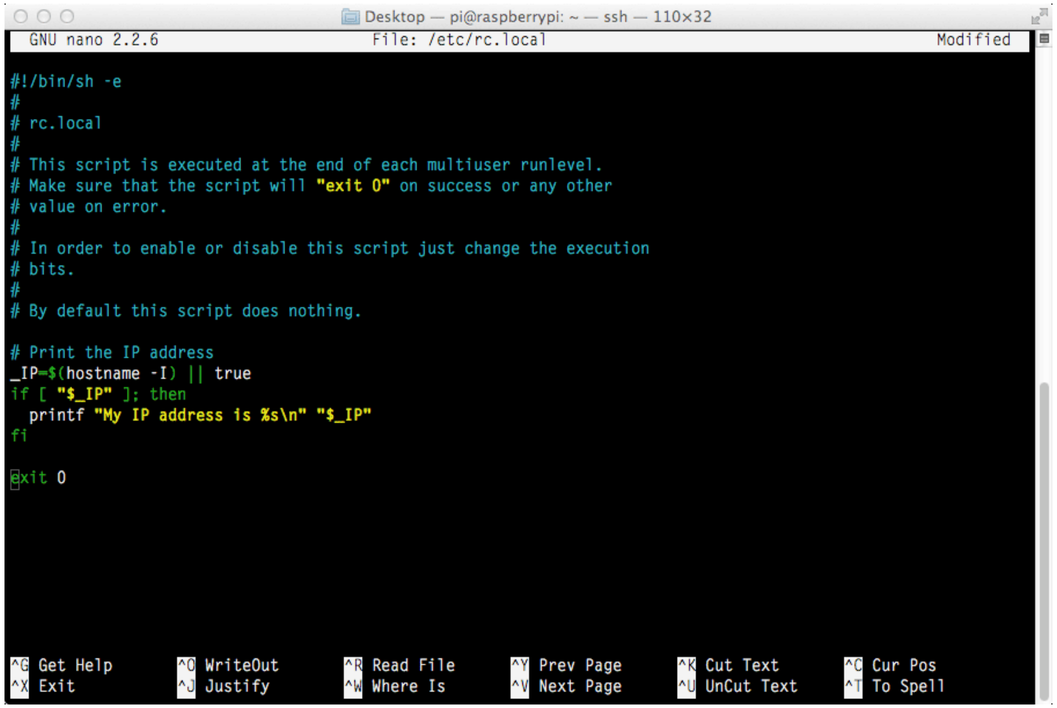
Software preparation

Auto-start script configuration

This allow STR2STR to auto-start when system boot completed

Edit “rc.local” by type this command below

$ sudo nano /etc/rc.local



Desktop — pi@raspberrypi: ~ — ssh — 110×32

```
  GNU nano 2.2.6                      File: /etc/rc.local                                          Modified

#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.

# Print the IP address
_IP=$(hostname -I) || true
if [ "$_IP" ]; then
  printf "My IP address is %s\n" "$_IP"
fi

exit 0

^G Get Help      ^O WriteOut      ^R Read File     ^Y Prev Page     ^K Cut Text      ^C Cur Pos
^X Exit          ^J Justify       ^W Where Is      ^V Next Page     ^U UnCut Text    ^T To Spell
```

Software preparation

Auto-start script configuration (2)

- After the initial comments (lines beginning with '#') add those commands below.

```
sudo mount /dev/sda1 /media/usb -o uid=pi,gid=pi
cd /home/pi/RTKLIB-master/app/str2str/gcc/
sudo ./str2str -in serial://ttyACM0:57600#ubx –out file:///media/usb/$(date +%Y%m%d-%H%M%S)
```

- First command will auto mount USB drive, Second will go to STR2STR directory and the last one will start STR2STR with writing the output to USB drive

Software preparation

<u>Trigger OS shutdown by USB</u>

- As the logger unit has no keyboard when operating on field.
- We add more script to be trigger for shutdown process. By disconnect receiver from Raspberry Pi.
- To protect the output file from EOF problem.

Software preparation

Trigger OS shutdown by USB (2)

1. At console, Get information about the device via "lsusb"

(The third field labelled ID is the vendor and model id separated by a colon)

2. Create a file in /etc/udev/rules.d

(This is the same regardless of distribution. The file must end in .rules and all files in this directory are processed lexicographically. Such as 00-XXX.rules)

Software preparation

Trigger OS shutdown by USB (3)

3. Edit the created file as below

    ACTION=="remove", ENV{ID_VENDOR_ID}=="XXXX", ENV{ID_MODEL_ID}=="XXXX", RUN+="/sbin/shutdown -h now"Create a file in /etc/udev/rules.d

4. Run command "udevadm control --reload-rules“ to take effect.
5. System is now ready to use as Logger unit.

How to use?

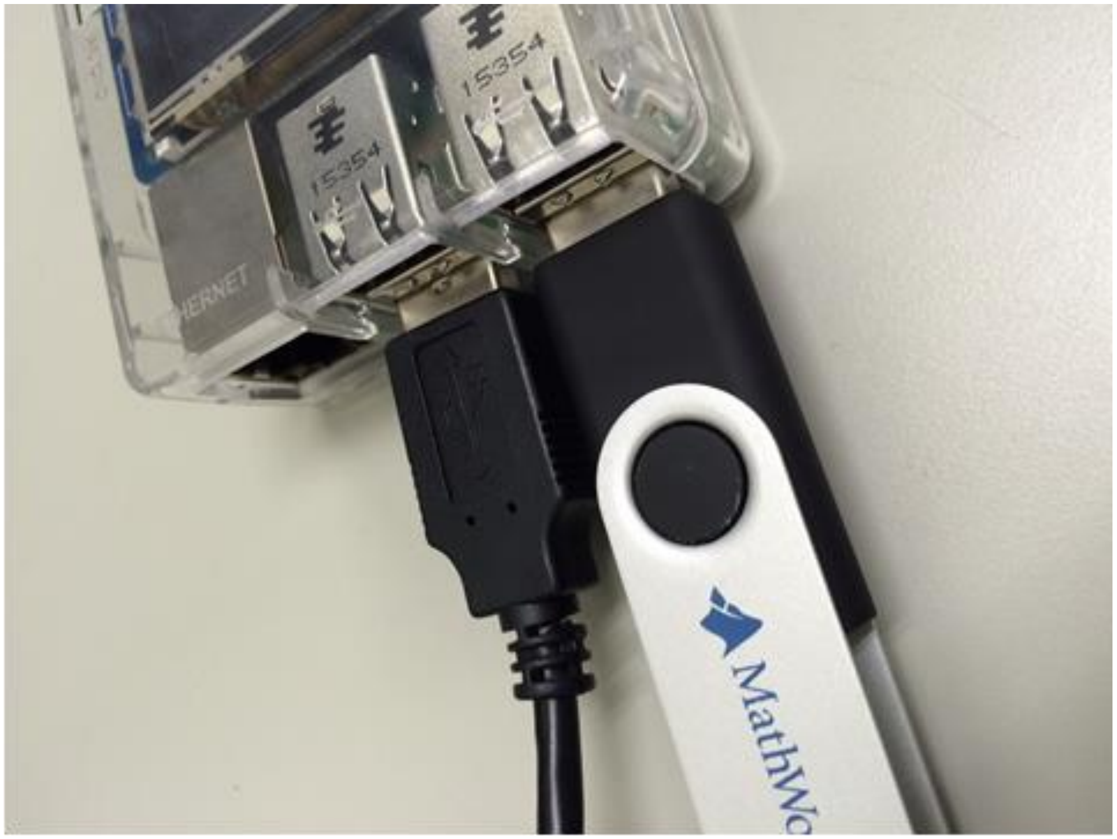
How to use?

We divide into 3 steps

1. Prepare and Start system
2. Shutdown system
3. Access the record file

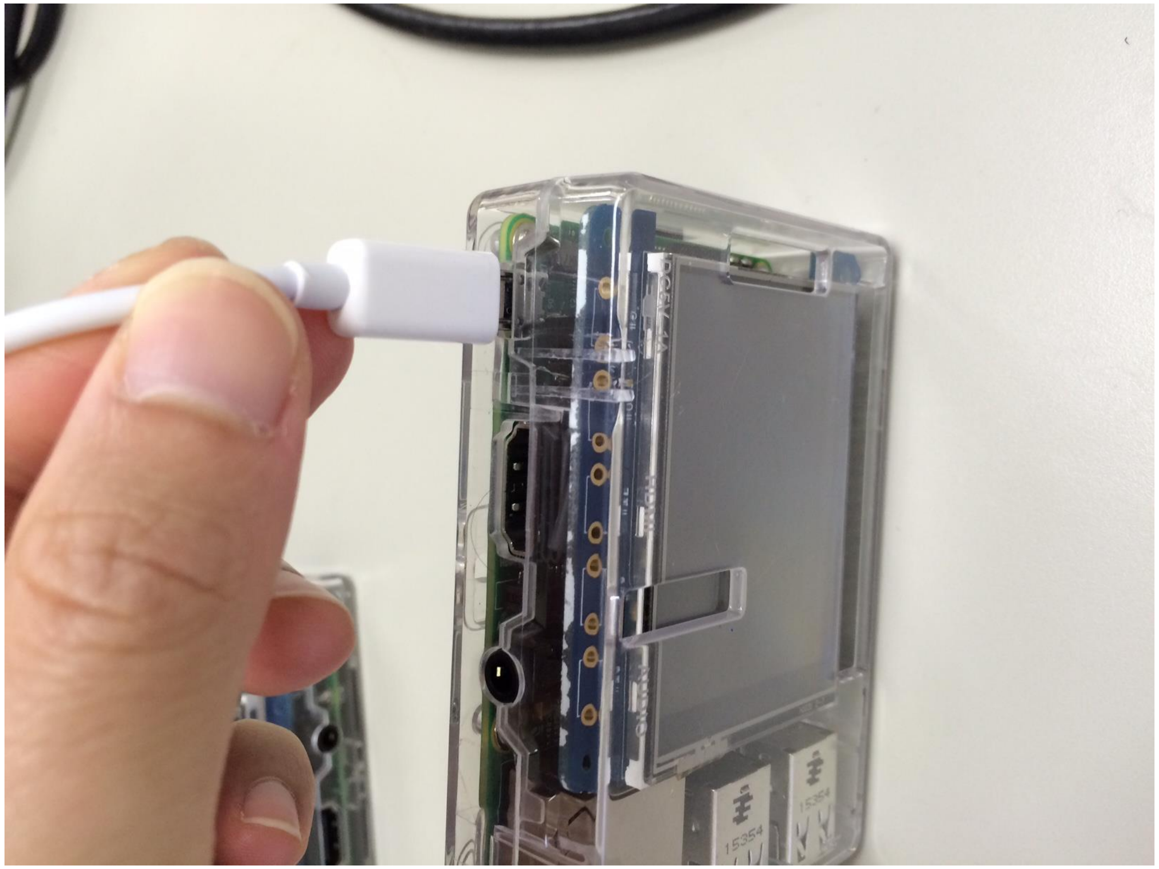
Prepare and Start system

- Plug Ublox Neo-M8T unit (via USB) and USB drive to logger unit



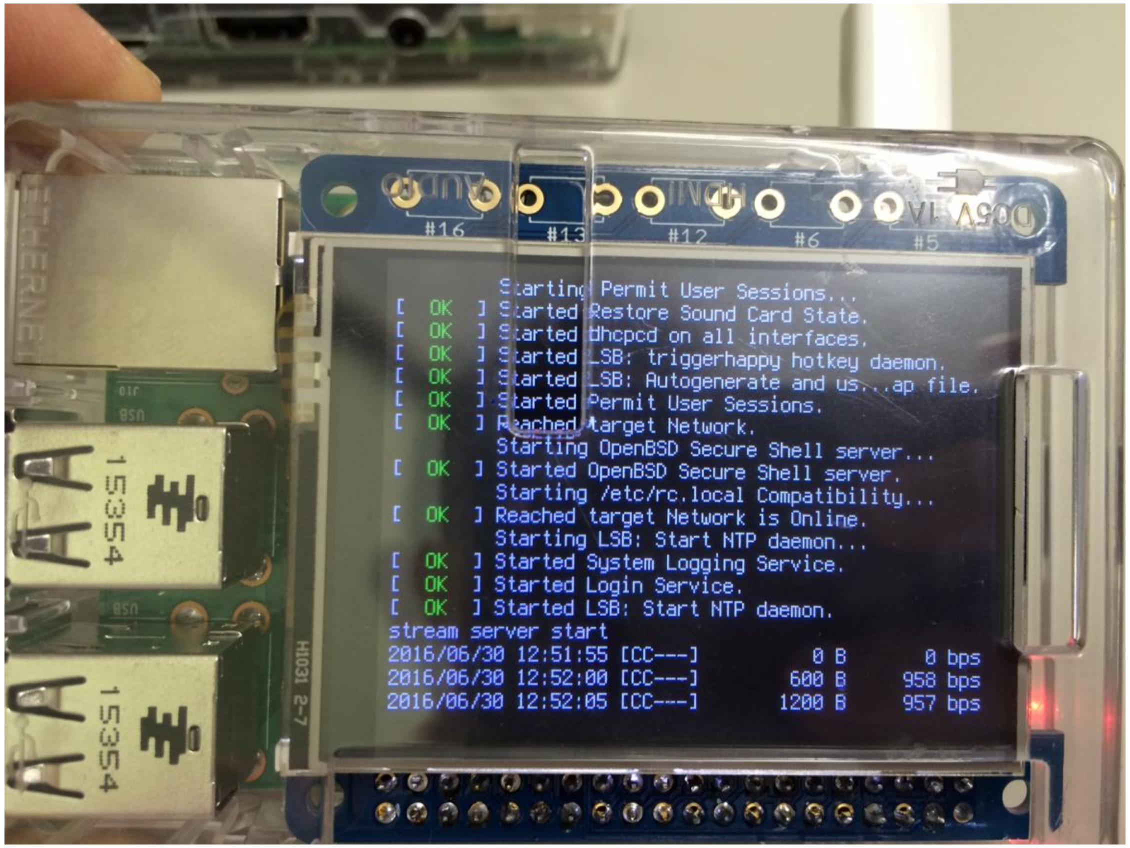
Prepare and Start system

- Plug power cable to logger unit (USB2.0 cable from power adapter or power bank)



Prepare and Start system

- Logger system will start automatically as picture below.

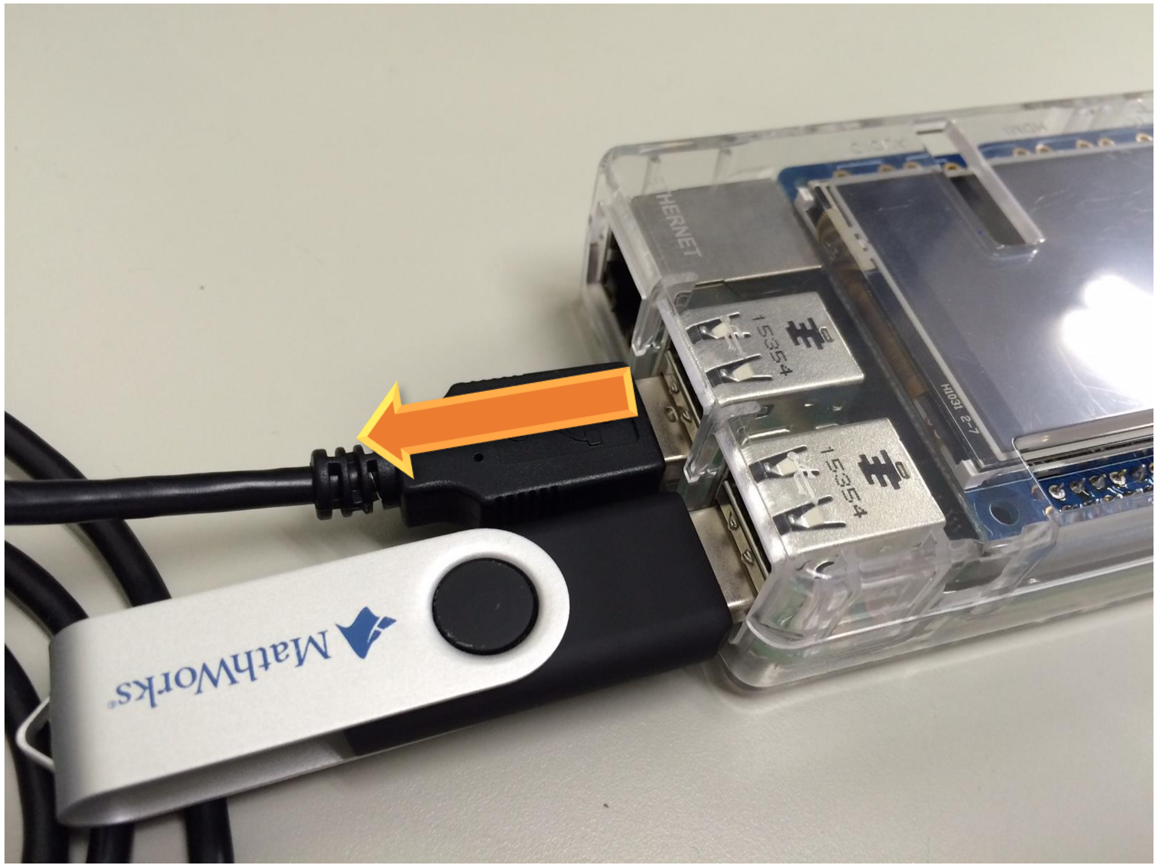


```
         Starting Permit User Sessions...
[  OK  ] Started Restore Sound Card State.
[  OK  ] Started dhcpcd on all interfaces.
[  OK  ] Started LSB: triggerhappy hotkey daemon.
[  OK  ] Started LSB: Autogenerate and us...ap file.
[  OK  ] Started Permit User Sessions.
[  OK  ] Reached target Network.
         Starting OpenBSD Secure Shell server...
[  OK  ] Started OpenBSD Secure Shell server.
         Starting /etc/rc.local Compatibility...
[  OK  ] Reached target Network is Online.
         Starting LSB: Start NTP daemon...
[  OK  ] Started System Logging Service.
[  OK  ] Started Login Service.
[  OK  ] Started LSB: Start NTP daemon.
stream server start
2016/06/30 12:51:55 [CC---]        0 B       0 bps
2016/06/30 12:52:00 [CC---]      600 B     958 bps
2016/06/30 12:52:05 [CC---]     1200 B     957 bps
```

(In case that system cannot start or boot to terminal prompt, please check M8T and USB drive are plug to the system correctly or not, then reboot the system again)

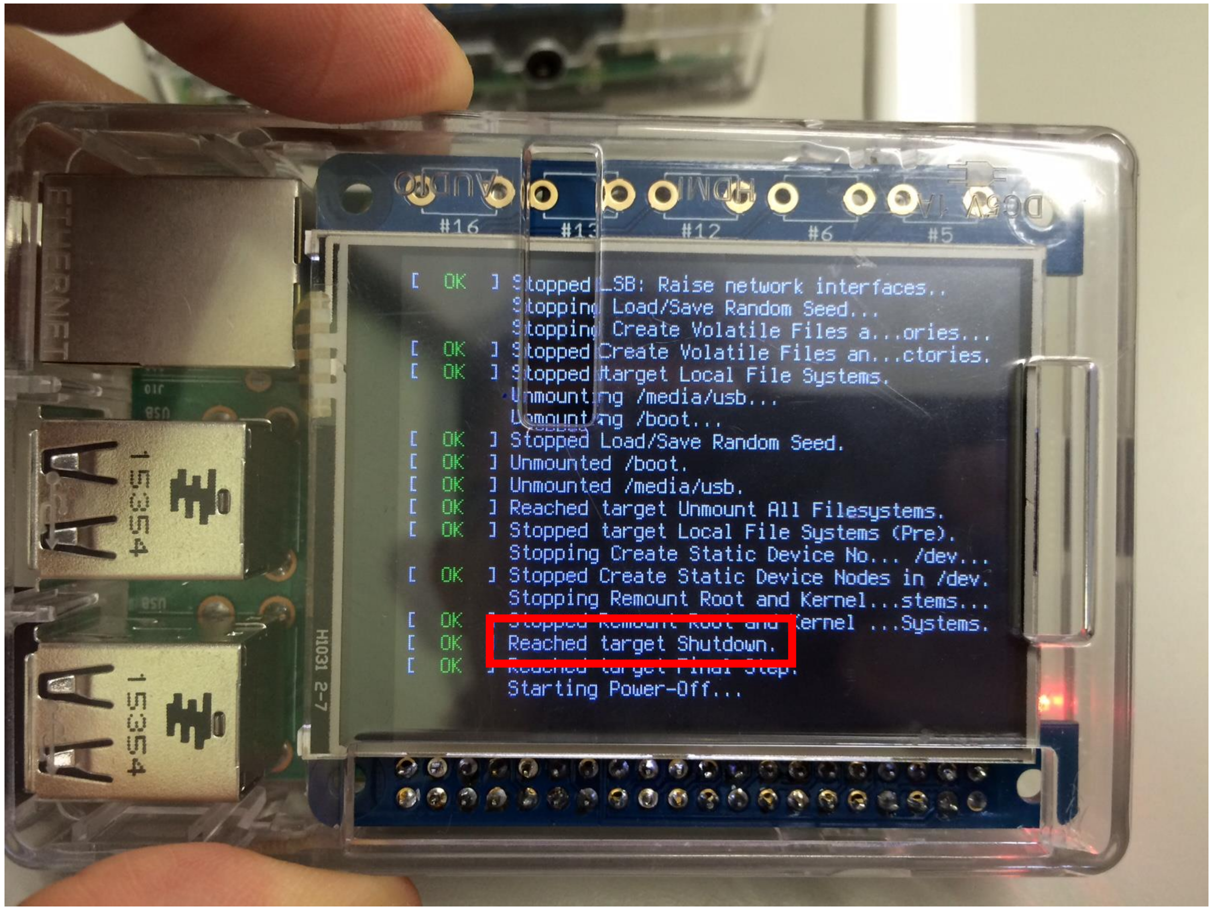
Shutdown system

- When you done the experiment, please unplug the Ublox Neo-M8T from logger unit and system will start to shutdown automatically



Shutdown system

- The last state of shutdown system is “Reached target Shutdown” then you can remove power cable

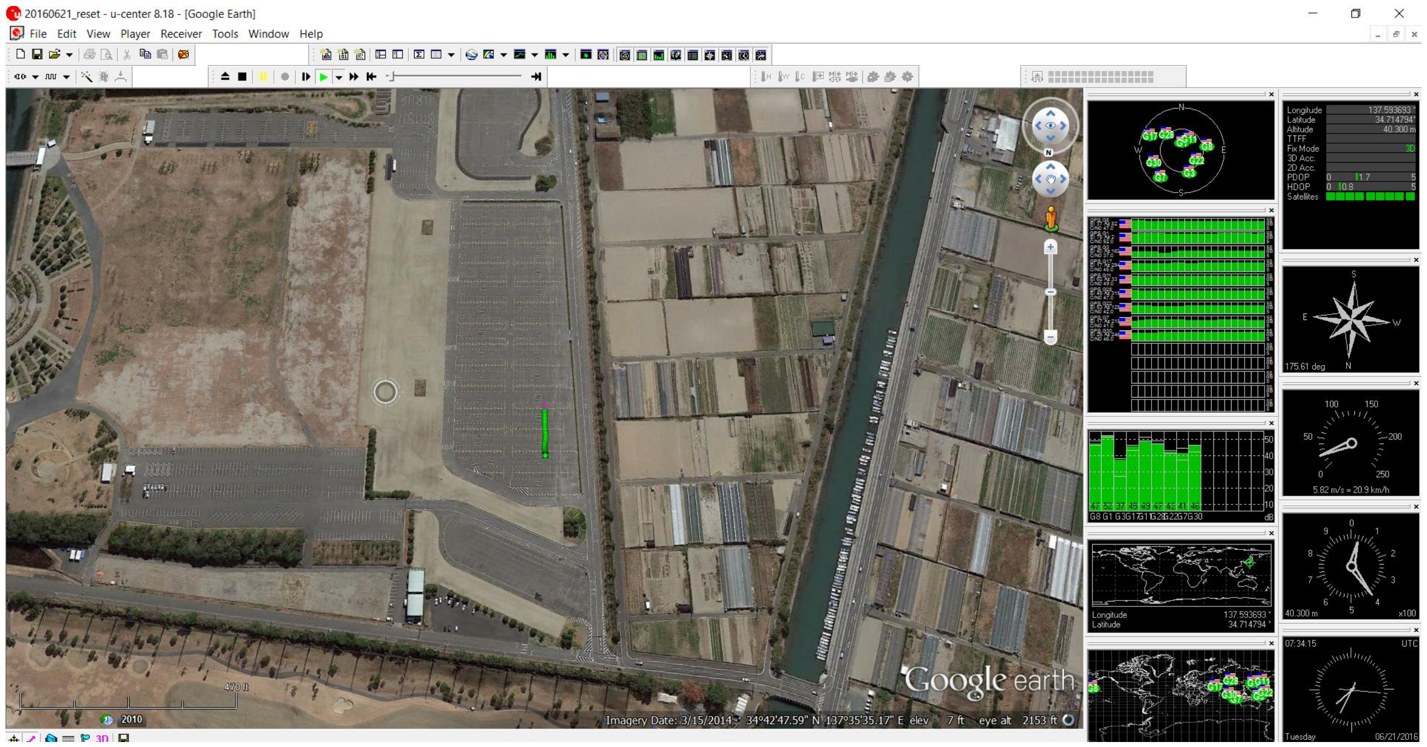


```
[  OK  ] Stopped LSB: Raise network interfaces..
         Stopping Load/Save Random Seed...
         Stopping Create Volatile Files a...ories...
[  OK  ] Stopped Create Volatile Files an...ctories.
[  OK  ] Stopped target Local File Systems.
         Unmounting /media/usb...
         Unmounting /boot...
[  OK  ] Stopped Load/Save Random Seed.
[  OK  ] Unmounted /boot.
[  OK  ] Unmounted /media/usb.
[  OK  ] Reached target Unmount All Filesystems.
[  OK  ] Stopped target Local File Systems (Pre).
         Stopping Create Static Device No... /dev...
[  OK  ] Stopped Create Static Device Nodes in /dev.
         Stopping Remount Root and Kernel...stems...
[  OK  ] Stopped Remount Root and Kernel ...Systems.
[  OK  ] Reached target Shutdown.
[  OK  ] Reached target Final Step.
         Starting Power-Off...
```

Access the record file

- When you shutdown unit, record file will available in the USB drive. Filename will be written by the time of Raspberry PI system, please check the actual time inside the file again (GPST)
- As the script configuration, file will not has the extension

Access the record file


20160621_reset - u-center 8.18 - [Google Earth]
File Edit View Player Receiver Tools Window Help
Google earth
Imagery Date: 3/15/2014 34°42'47.59" N 137°35'35.17" E elev 7 ft eye alt 2153 ft
2010
Longitude 137.593693°
Latitude 34.714794°
Altitude 40.300 m
175.61 deg
5.82 m/s = 20.9 km/h
40.300 m
07:34:15 UTC
Tuesday 06/21/2016

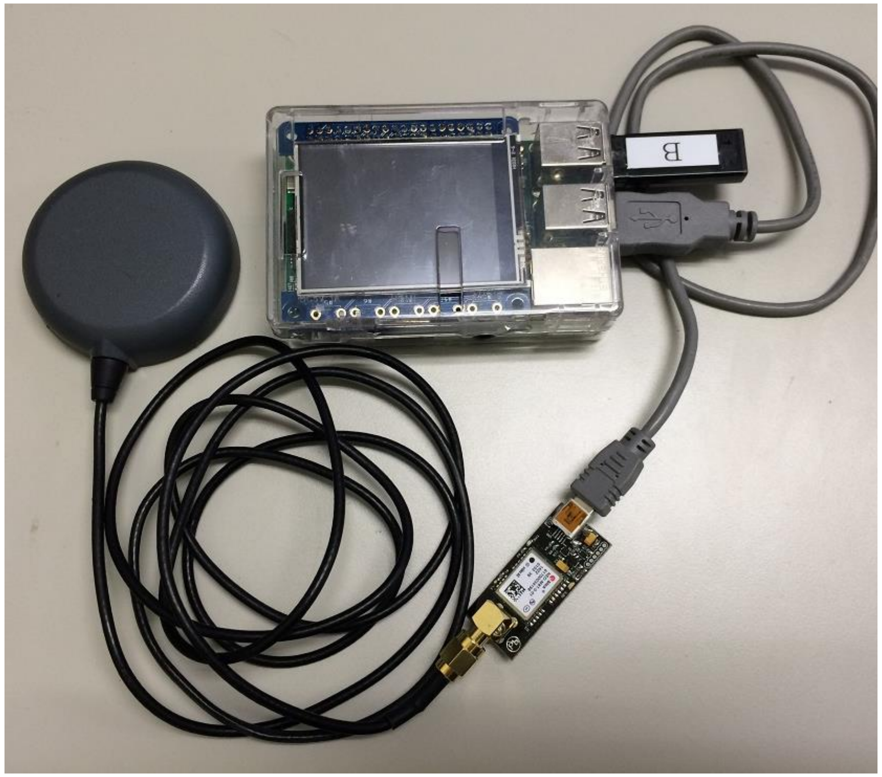
- Recorded file UBX can be review from ucenter application provided by Ublox.
- From here, you can briefly review data and position in map view

Experiment

Experiment with Logger unit

- We conducted an experiment using logger unit in Laboratory work, Hamamatsu and Summer camp in Thailand recently.
- At Summer camp, we collect data and did the post processing by using TUMSAT base station data

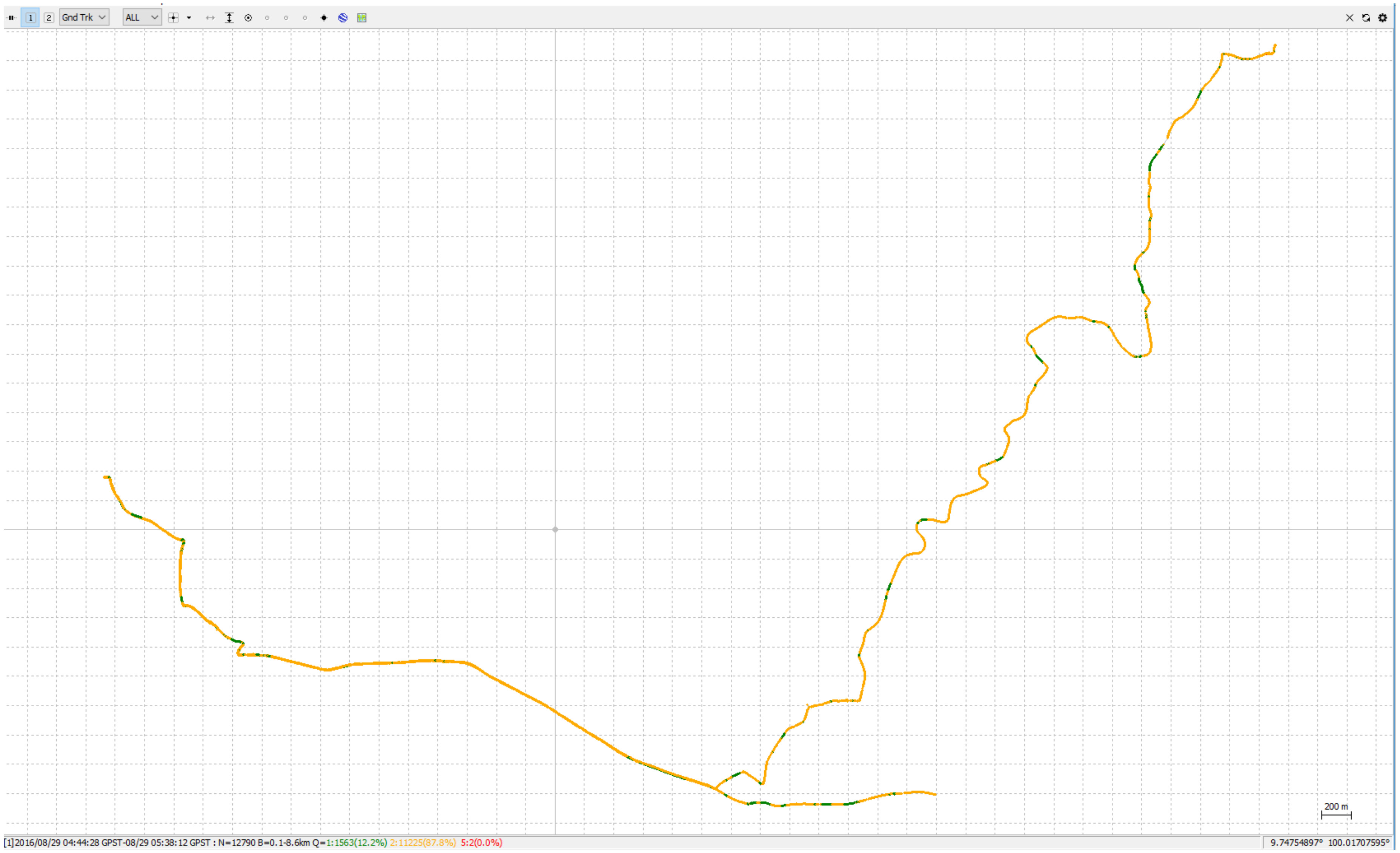
Summer Camp Experiment



Rover (Logger Unit)

Base station setup by TUMSAT

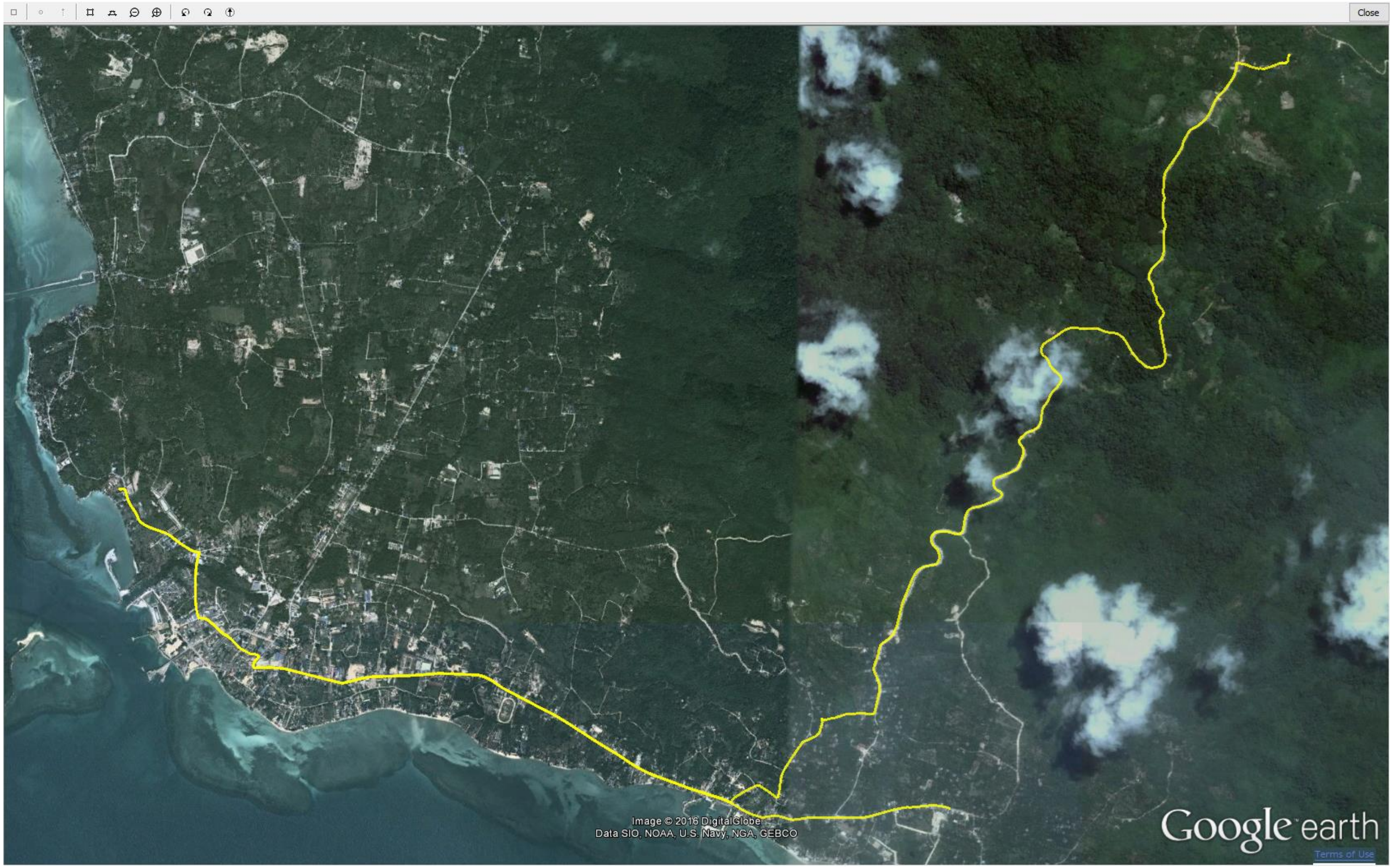
Experiment result (RTK Plot)



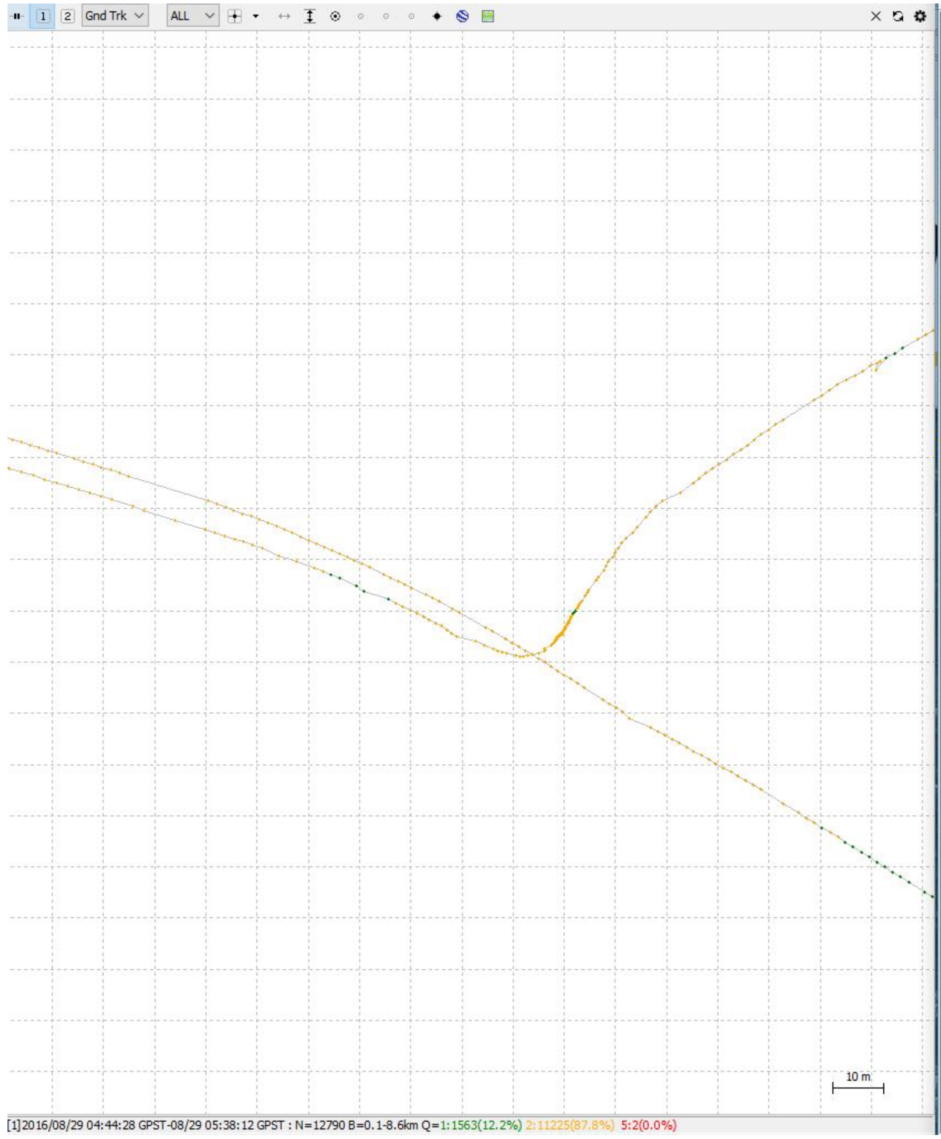
[1]2016/08/29 04:44:28 GPST-08/29 05:38:12 GPST : N=12790 B=0.1-8.6km Q=1:1563(12.2%) 2:11225(87.8%) 5:2(0.0%)

9.74754897° 100.01707595°

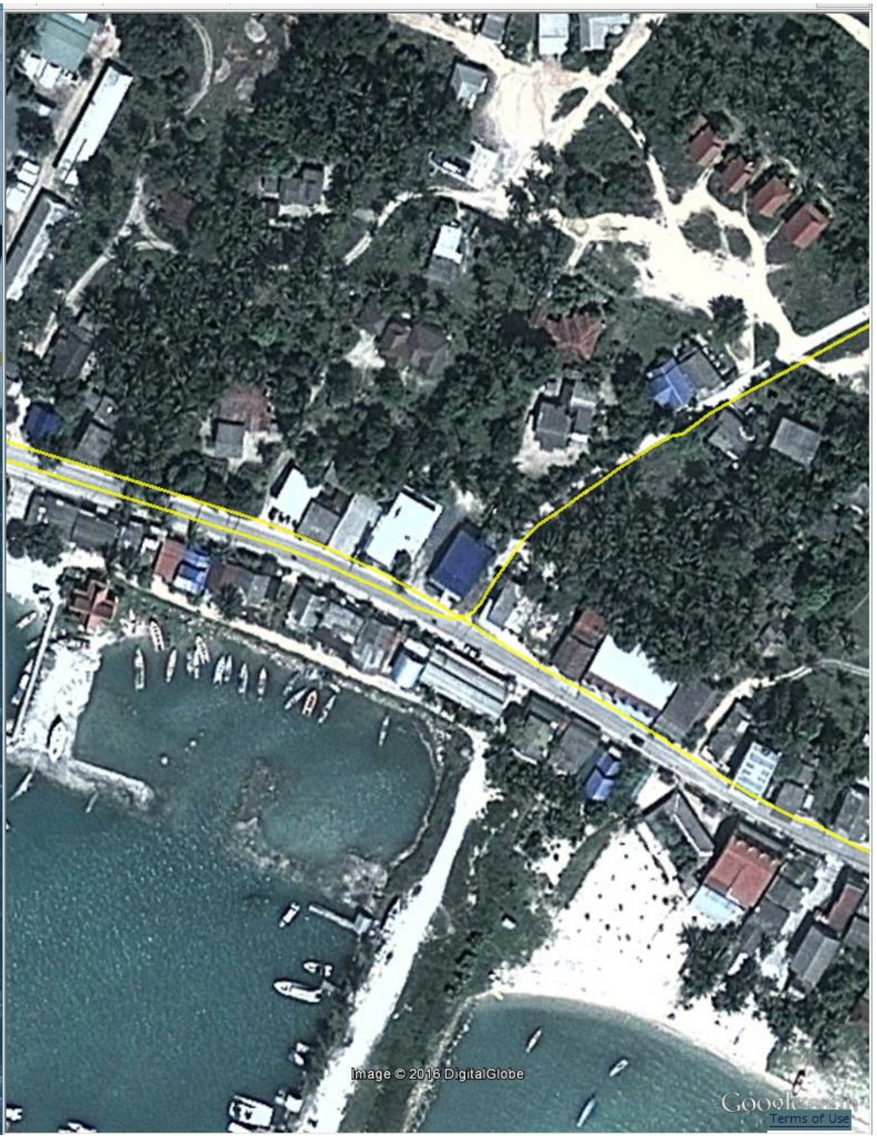
Experiment result (Google Earth)


Close
Image © 2016 DigitalGlobe
Data SIO, NOAA, U.S. Navy, NGA, GEBCO
Google earth
Terms of Use

Experiment result (Crop some area)



[1]2016/08/29 04:44:28 GPST-08/29 05:38:12 GPST : N=12790 B=0.1-8.6km Q=1:1563(12.2%) 2:11225(87.8%) 5:2(0.0%)



Future work

- Study and develop the prototype for more efficiency.
- Gathering the record data for improve my research

Contact point

- Professor Tomoya Kitani

  t-kitani@inf.shizuoka.ac.jp , t-kitani@kitanilab.org

- Tiphat Areeyapinun

  gs15701@s.inf.shizuoka.ac.jp , tiphat-a@kitanilab.org

- Kitani Laboratory, Department of Informatics, Shizuoka University