

PythonçijŮćíŇæŮúăĚĹ'ăĚŇ

Release 1.0.0

ăřŔæŸŌăŔŇă■ę

Apr 17, 2019

Contents:

1	A MazeăžŇçž■	1
2	æŮăăžžæIJžçijŮéŸş	1
2.1	çžĎěčĚăŸĂæđŭæžæűşçijŮéŸşéIJĂæśĆçŽĎěčđæIJž:	1
2.2	èįřăžŭăŔĹăŧŇăĚăįjŔçşžçžşéĚ■çįő	4
2.3	çijŮéŸşçĹŇăžŔěğčæđŔ	9
3	ăĹŽæŮřăśŦçđ'ž	12
4	æIJžăŽĹăžžăřijĚĹ	15
4.1	çşžçžşéŦIJăĈŔěğčăŇĚăŸŌæĹ'şăŇĚ	15
4.2	çşžçžşéŦIJăĈŔěčĂăĹĹ	17
4.3	çşžçžşçĈğăŦŦ	20
4.4	ăŸşăŔčĕŕĈĕŦŦ	22
4.5	çşžçžşăĚĚăŸăĹŌăŦž	23
4.6	çşžçžşçŌŕăćĈĚĚ■çįő	25
4.7	Optitrack	30
4.8	EigeniijŇmavros	30
4.9	ăžd'ăŔĹ'çijŮĕŕşăŽĹ	30
4.10	motiveéĚ■çįő	37



1 AMazeäzŃčž■

AMazeæĹŔčŋŃäžŮ2017ážt' 6æIJĹ, AMazeäꞤĹæL' Ÿā■ŌāŃŮčŤtāŁZād' gā■ęčŤtāŁZæŽžèČ;æIJžāŽÍäžžāũ
AMazeāŽcéŸšçŤsāŔt' ā■Ōā■ŽāčŋāŠŃäŸĀçŁd' āŕŃæIJĹ çČ■æČĚæĠČāꞤ ŮāĹZæŃĀāŮĎäžŮāŁZæŮřçŽĎā■ęç'

2 æŮäăžžæIJžçijŮéŸš

2.1 çŽĎèčĚäŸĀæđúæzaæùšçijŮéŸšéĹĀæśČçŽĎéčđæIJž:

éĀŽäŸŸäŸĀäŸĹāđ' ŽæŮŃçŤijçžĎæĹŔāŃĚæŃňäžčäŸŃçāňäžŮ(āŽŽèĵt'):

äŸĀäŸĹāŽŽèĵt' éčđæāŃāŽÍçŽĎæđĎæĹŔĲijŽ

1. āŽŽèĵt' æIJžæđŮ X1ĲijĹāŁĚéąžĲijĹ'
2. āŁĹāŁZéĹ' ñèŁŁ X4 ĲijĹāŁĚéąžĲijĹ'
3. æŮāāĹŮçŤtā■ŔèŕČéĀšāŽÍĲijĹçŤtèŕČ/ESCĲijĹ' X4 ĲijĹāŁĚéąžĲijĹ'
4. PixhACK v3 éčđæŮğ X1 ĲijĹāŁĚéąžĲijĹ'
5. CUAV GPS X1 ĲijĹāŁĚéąžĲijĹ'
6. æŮāçžŁæŤŕāĲjā (CUAV RADIO/XBEE/XTEND/HACKLINK/WP-LINKĲijĹ' X1āŕž ĲijĹāŁĚéąžĲijĹ'
7. āŁĹāŁZçŤtæšāĲijĹāŁĚéąžĲijĹ'
8. RCéĀčæŮğāŽÍāŠŃRCæŮčæŤŮæIJžĲijĹāŁĚéąžĲijĹ'
9. æŮāāĹŮäžŠāŔŕæĹŮèĀĚçŽŸæIJžĲijĹāŔŕéĀĹ'ĲijĹ'
10. èŮĚāčŕæšçæĹŮèĀĚæŁĀĚĹ'āĲjāæĎšāŽÍĲijĹāŔŕéĀĹ'ĲijĹ'
11. āĚĹ'ætĀāōŽçČžāĲjāæĎšāŽÍĲijĹāŔŕéĀĹ'ĲijĹ'

æŤŕæŃĀçŤtæšāçšžāđŃ

äĵŁçŤĹæāĠéĚ■çŽĎCUAV IVæĹāāĲijĹŃæŤŕæŃĀæāĠāĠĚ2-6VāŁĹāŁZçŤtæšā

IVæĹāāĲijĹŃæŤŕæŃĀ2-6VçŤtāŮŃāĀĀ0-60ĀçŤtætĀ āōđæŮŮçŽŠætŃŃ

çŔĚèōžäŸĲijŽ

éĹĀèçĀāōđæŮŮæŮğāĹŮèČ;éčđæāŃçŽĎénŸāžçāŠŃèŮĲçž:èŮšāŁĹāŁZçž■èĹæŮŮéŮt' āĀĀéĀčæŮğāŽĹā

èĠäŸžéčđæāŃçŽĎénŸāžçāŠŃèŮĲçžĲijŽèŮšāŁĹāŁZāĀçž■èĹæŮŮéŮt' æIJĹ'āĚš

四轴飞行器构成：



四轴机架*1架(必须)



PIXHACK-V3飞控*1个(必须)



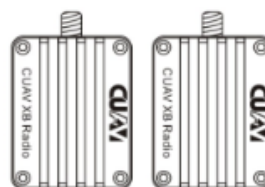
CUAV GPS*1个(必须)



无刷电子调速器*4个(必须)



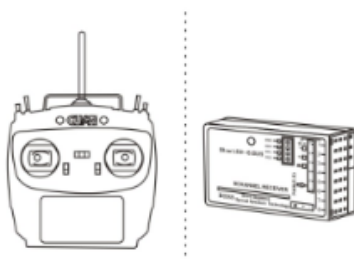
动力马达*4个(必须)



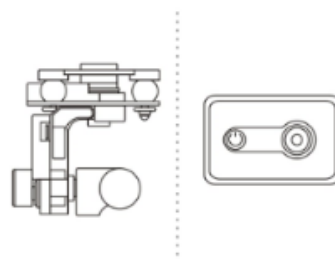
无线数传*1对(可选)



动力电池*1个(必须)



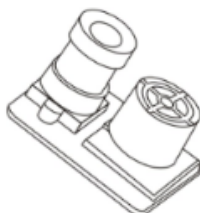
RC遥控器和RC接收机(必须)



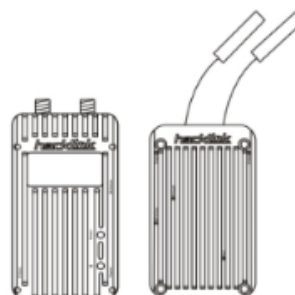
无刷云台/相机(可选)



超声波或激光传感器(可选)

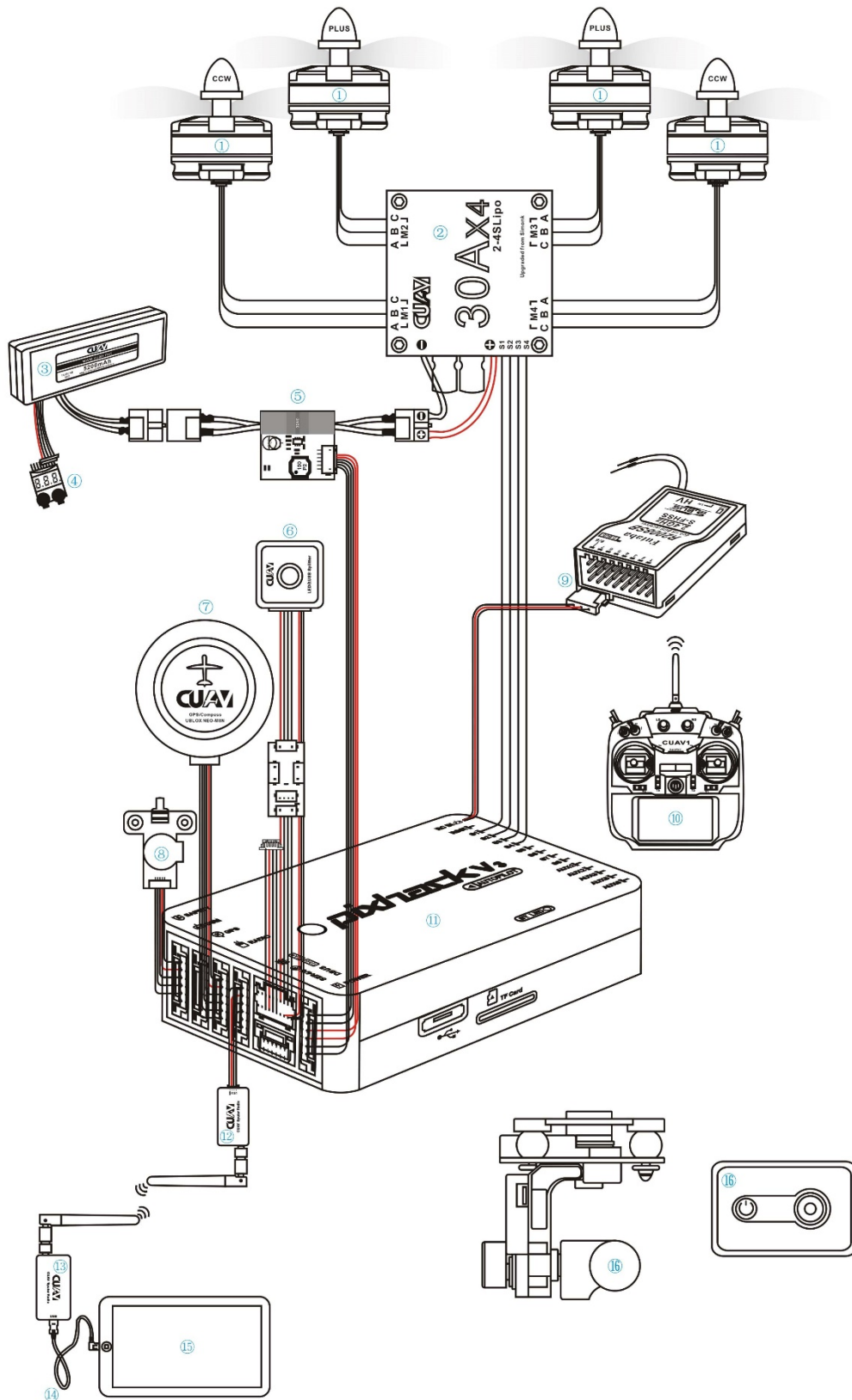


光流定点传感器(可选)

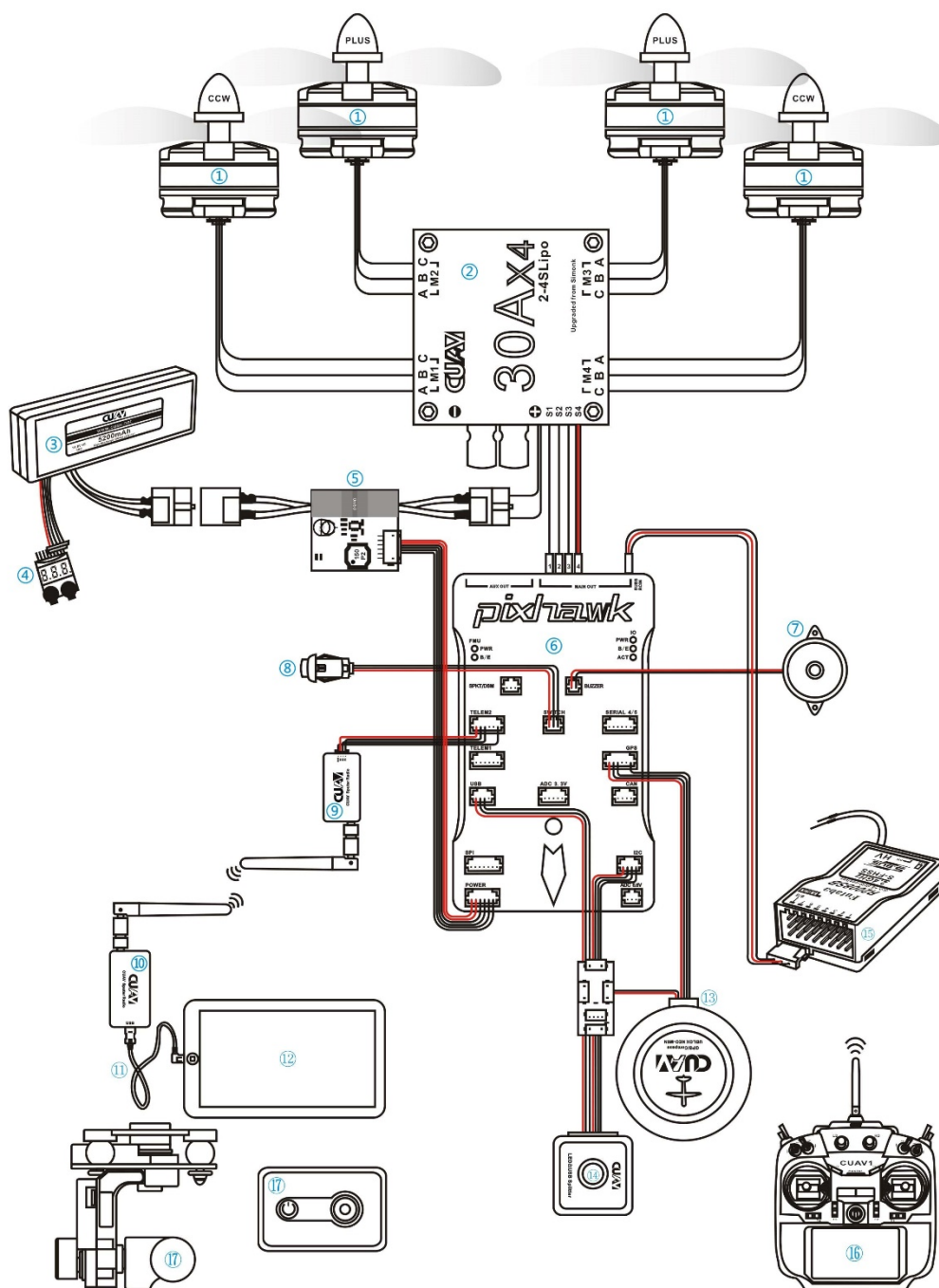


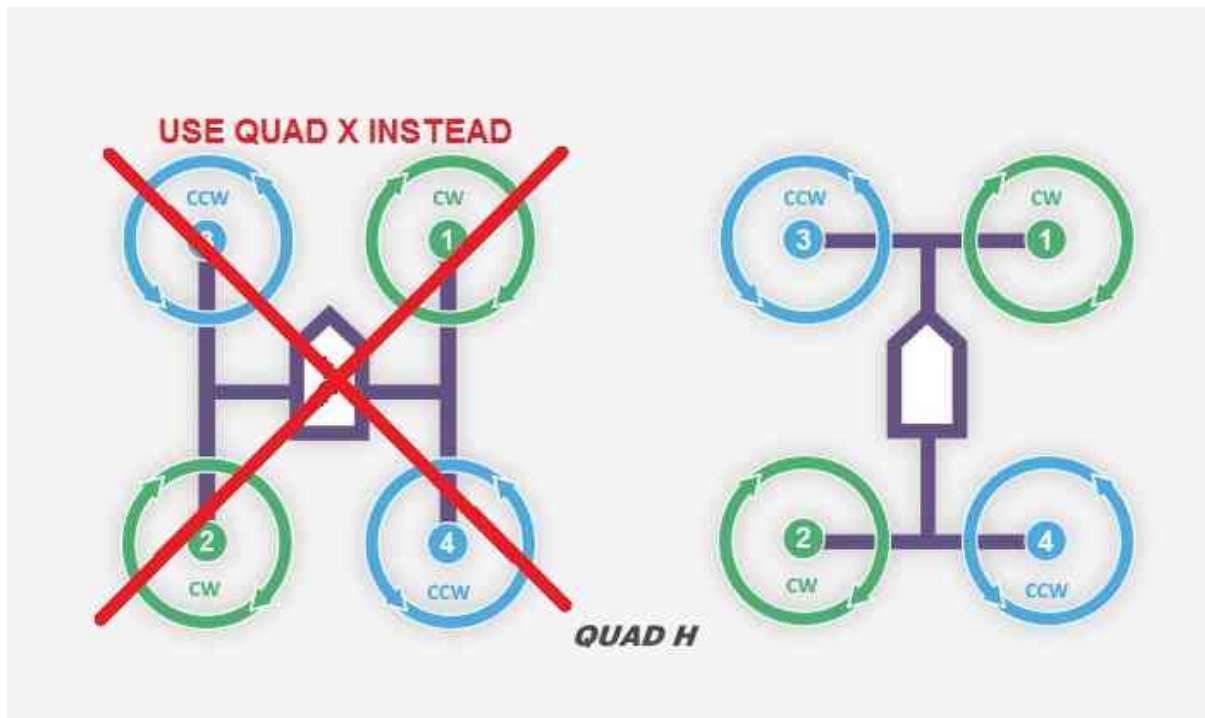
数字链路*1对(可选)

1、动力马达	4、低温报警器 (BB响)	7、NEO GPS	10、RC遥控器	13、无线数传接收端	16、无刷云台/相机
2、电子调速器	5、电压电流模块	8、蜂鸣器	11、PIXHACK主控	14、OTG线	
3、动力电池	6、LED及扩展板	9、RC接收机	12、无线数传发射端	15、手机/平板显示	



1、动力马达	4、低温报警器（BB响）	7、喇叭蜂鸣器	10、无线数传接收端	13、NEO GPS	16、RC遥控器
2、电子调速器	5、电压电流模块	8、安全开关	11、OTG线	14、LED及扩展板	17、无刷云台/相机
3、动力电池	6、PIXHAWK主控	9、无线数传发射端	12、手机/平板显示	15、RC接收机	





- 2) `çşżçżşéTİJâČŘæŮĞäzűâIJÍUbuntu mateãŏŸç;ŚäyŁäyNè;ııijNâEŻâĖĖSDâ■açŽĐè;řäzűWin32 Disk Imager`
- 3) `ãŏL'èčĖæŞ■ä;IJçşżçżşæŮııijNâNĭéĂL'âijĂæIJžèĞtâŁĭçŽzâ;TăĂĆ(ãŏđçŎřauto-login)`
- 4) `æŞ■ä;IJçşżçżşãŏL'èčĖãŏNæĹŘäzěâŘŎııijNäyNè;ııæŮĞæIJñcijŮè;ŚâŽĭgeditııijNæŮzä;ŁăŘŎçz■æŽt' æT' sudo apt install gedit`

2. âřĖUARTâŘřçŤĭäyžcommunication interfaceãĂĆ

- 1) `sudo gedit /boot/cmdline.text`
- 2) `âĹăéŽd'èř■âŘěâĂŸconsole=serial0,115200âĂŽăĹâ■ŸăzűéĂăĂĞžăĂĆ ##### 3. äŁŏæŤžUARTéĖ■ç;ŏııijNâĖşéŮ■èŞĭçL'ŽııijLæşĭ:mateçşżçżşéçŸŸèŏd'èŞĭçL'Žă■ăçŤĭUARTııijL'`
- 3) `sudo gedit /boot/config.txt`
- 4) `ăĹŏæŤžinit_uart_clock to 16MHzăžěâŘĹinit_uart_baudrate to 921600ăžűâĹăéŽd'èř■âŘěâĹ■éĭçŽĐæşĭéĞĹçņăŘû#ăĂĆııijLæşĭ:éčđæŎğTelem2æşççL'žçŎĞăžşèçAç`
- 5) `âIJĭæŮĞæIJnäyNéĭçæűzâĹăér■âŘěâĂŸcore_freq = 250âĂŽ`
- 6) `âIJĭæŮĞæIJnäyNéĭçæűzâĹăér■âŘěâĂŸdtoverlay=pi3-disable-btâĂŽăĹâ■ŸăzűéĂăĂĞž`
- 7) `sudo systemctl disable hciuart sudo reboot`
`éĞ■âŘřăŘŎUARTéĖ■ç;ŏãŏNæĹŘ`

4. **Python na Linuxe: mateczszechy python 2.7 Orac Cijl'**

- 1) `sudo apt install python-pip python-dev sudo pip install pip dronekit xbee numpy gps pyzmq --upgrade`
- 2) `sudo apt install python-serial`

5. **ajArsshAC**

- 1) `sudo apt install openssh-server openssh-client`
- 2) `sudo raspi-config`
- 3) `gZOrE[ç]oçTNeIcijNeALæNI'âYnetworkâZãAC`
- 4) `éALæNI'âYssh enabledâZãAC ##### 6. èç;ç;oiJæIJzèGlâRçlNâžRãAC(çãöäIauto-loginâušãóçÖř)`
- 5) `sudo gedit /etc/rc.local`
- 6) `âIJæÜGäzũäy■æũzãŁă cd /AmazeFly python onboard.py -xbee /dev/ttyUSB -pix /dev/ttyAMA0ijNäfiâ■YéÄãGž`

XBeeé[ç]oâRCèAC

1. **âZzäzüçGgäEZ**

- (1) `äyNèj;âũčäEüXCTU`
- (2) `çÇgäEZDigiMeshâZzäzü(æšl:æÖlè■R8073 – Xbee DiGiMesh 2.4)`

2. **ârEé[ç]öæÜGäzũârijäEëXBee**

2.3 **çijÚéYšçlNâžRègčædŘ**

Introduction

Welcome to the AmazeFly project! âIJAmazeFlyâI is a multi-copter drone test platform originally designed by Amazegroup, NCEPU.

The âI AmazeFly âI drones take the [Pixhawk](#) and the [ArduPilot](#) stack as their low-level flight controller and use [dronekit-python](#) as the high-level application control. So far there is no modification at the Pixhawk and the ArduPilot level, so this project is purely written in Python running on a Linux companion computer (e.g. Raspberry Pi).

The âIJAmazeFlyâI drones use [XBee](#) modules to establish a high-level communication network between the drones and the ground control station.

License

AmazeFly Project is made available under the permissive open source Apache 2.0 License.

Coding convention

This project is entirely written in Python 2.7, under the convention of [Google Python Style Guide](#).

System requirement

Drone

1. A multi-copter drone using Pixhawk as its flight controller.
2. A mini onboard Linux companion computer. (e.g. Raspberry Pi)
3. An XBee module with a USB adapter. (e.g. XBee S1, XBee S2C, etc.)

Note: Zigbee's are not recommended as they are relatively slow and have small data throughput volume. Zigbee modules tend to get stuck often. The XBee Pro S1 with DII Mesh firmware is tested to be working very well. A new hardware upgrade by DII unifies XBee and Zigbee to a S2C version, which are now compatible across all the DII product lines.

4. (Optional) A USB-TTL adapter for debugging (e.g. FT232, CP2102/CP2104, do not use PL2303)

Connection example: The XBee modules is connected to the RPi via USB, and the Rpi is connected to the Pixhawk via UART. Configure the hardware UART on the Rpi to run at 921600bps baudrate, consistent with the setup on Pixhawk's serials. See [Companion Computers](#) page in the ArduPilot dev wiki.

Ground control station

1. A Linux computer. (Virtual machines are okay)
2. An XBee module with a USB adapter. (e.g. XBee S1, XBee S2C, etc.)

Packages

1. Linux packages: python-pip python-dev python-serial python-gps gpssd gpssd-clients
2. Python packages: dronekit dronekit-sitl xbee numpy pyzmq

The main composition of the project

gcs.py: The ground control station script for the quadcopter flocking control experiment.

onboard.py: Onboard main script for the multi quadcopter flocking control experiment.

comm.py: Communication classes and functions.

mas.py; Multi-agent system control algorithm module.

nav.py: Navigation functions.

shared.py: Attributes shared across the files.

util.py: Utility functions.

missionparser.py: Upload the missions.

PWMController.py: Turn on or turn off the LED on the drones.

mission_txt: The directory to place some TXT files of the missions.

How to use

1. Plan a mission:

Software: APM Planner 2.0

Steps:

```
(1) Run the software.
(2) Click 'FLIGHT PLAN'.
(3) Click on a point on the map (with number of 0) as the 'Home_
    ↳ Origin'.
(4) Click the 2nd point (with number of 1) as the 'Takeoff' point.
(5) Continue to click some points as the waypoints.
(6) Altitudes, accuracies and headings of the points can be changed_
    ↳ in the lower part of the pages.
(7) Copy the generated TXT files to the mission_txt directory and_
    ↳ divide them by the serial numbers of the drones. (The TXT files 's_
    ↳ names must be 'FIRST.txt', 'SECOND.txt' or 'THIRD.txt'.)
```

Reference: <http://ardupilot.org/planner2/>

2. Run onboard.py :

The onboard.py script will run automatically when the power of the drone is switched on.

(6)The console will start prompting some information. Input keys accordingly to execute certain operations.

```
'x': 0, # switch the mode to auto
'1': 1, # the first mission
'2': 2, # the second mission
'3': 3, # the third mission
'1': 4, # landed
'd': 5, # light-high level
'k': 6, # kill thread and restart
```

- 1). press ``1'/'2'/'3'`` to choose the first/second/third mission.
- 2). press ``x'`` to switch the mode to auto and take them off.
- 3). press ``d'`` to turn on the LED.
- 4). press ``l'`` to directly land the drones.
- 5). press ``k'`` to kill thread and restart.

3 ĀŁŽæŮřǎśȚçd'ž

-2018åzt'10æIJL27æUëæŽŽiijŇçT'sæŎğèőaã■ęéŻcãRt'ã■ŎèĀAąŸŁŁäyęćEçŽŽDAMazeiijŁamazething.ic
æUăăžžæIJzãIJăd'ŀ'çł'žçžDăĹŖĪăîd' N C E P UçŽŽă■ŮăũũiijŇęŁŎĬĂăd'Uçł'zã■ĞëtũũijŇăIJçł'žäy■æTčĂŤ



-èŁŽăŽăd' IJçl' žçŽDçşçAŧæŮüèĀŇçl' Łæć■āRŸæ■ćæĹŖâĀIJâŮāâĀIéŸŧăđŇă;ćæĹŖâĠièAŽăĹZăđAăijžçŽ
 -ăIJlăd' IJçl' žăy■rijŇŇ C E P UèŁŽăžŤăyĹă■Ůæŕ■èćŋçşçAŧăžŋăŽŧ' çžŤăIJçl' žăy■éĀăăĠžăyĀăyĹăŭlăd' ġç
 -æŮăăžžæIJžæĖćæĖćçžĎæĹŖâĀIJăĖ■ă■AăĀiçŽĎă■ŮæăŭăĀĈăE■æŋăæĹĹæŤŧ' äyĹăăĹæijŤæŌĹăŖŖăžE



-AmazeǎŽćéŸšè£ŸǎřEǎijǞæžŘæIJǎæñaçijŮéŸşçŽǾL'ǞæIJL'æŮĞæaçètǾDæŮŽǎŠŇăžčçǎAǎǞĆæñćè£ŌǎĚ



thingčŽDă;ăăŁăăĚčăŽcéŸšřijA

4 æIJžăŽĺăžžăríjèŁ†

4.1 çşžçžšéŤIJăČŘèğčăŇĚäyŎœL'šăŇĚ

æIJñéąžçŽŏæIJĂăĹiă;ŁçŤĺçŽDæŸřăďŇăŔŭăyžMIQIřijŁSBCřijL'çŽDæĹă■ăřijŇăď'ĐçŘĚăŽĺăyžRK328

1.èŎŭăŁŮăžŘčăAřijŽ

```
1. $mkdir miqi_root
2. $cd miqi_root
3. $git clone https://github.com/mqmaker/miqi-linux-build.git build
4. $git checkout -b rklinux remotes/origin/miqi/v3.10
```

(continues on next page)

(continued from previous page)

```
5. $git clone https://github.com/mqmaker/miqi-linux-kernel.git
   ↪ rockchip-kernel
6. $git checkout -b linux remotes/origin/miqi/linux
```

2.ářĚǎǒŸç;ŠäyÑè;çŽǾlubuntuèğčǎŇĚiijŽ

```
1. ./ rkImageMaker -unpack lubuntuÉŤIJăĈŔăĬŸæŤ;çŽĎă;■ç;ő
↪èèAă■ŸæŤ;ăĽŕçŽĎă;■ç;ő
ii jĽæIJŋéazçŽōæŸŕăĽăžăžăžĚăŸĂăŸŷăŸŌăŔŌéÍcăŔŔăĽŕçŽĎăĚĚăăŸæžŔçăĂăđ' ĎăžŌăŔŇăŸĂçžğçŽ
2. ./afptool -unpack build/firmware.img buid/unpack
3.ăĤŏăŤzăŔŌăĽ'ŠăŇĚ
ăĤŏăŤză■sărzăĚĚăăŸăĽŮăĚăŸŕăŮĜăžăűçşzçzŞşĚŹăăŇăĤŏăŤzăŔŌii jŇcŤĽăŮŕçŽĎăĚĚăăŸăĽŮăĚă
```

æslæĐRiijŽpackage-fileéGŇéIccŽĐlinuxrootèurá;ĐæYréTŽčŽĐiijŇaIJllinuxrootaL■éIcāŁääyŁImagei.

```
# NAME          Relative path
#
#HWDEF          HWDEF
package-file    package-file
bootloader      RKLoader.bin
parameter       parameter
# boot to linux
recovery        Image\recovery.img
#cache          Image\small_ext4.img
#data           Image\small_ext4.img
backup          RESERVED
linuxroot       Image\linuxroot.img
```

äŁæTžēĠăŘŌiijŃă;ĤçŤlăžěäyNēĎŽæIJñēĤZēąŃæL'ŞăŃĚiijŁæşlæĎŔăL■ēĬăŔYéĠŔēő;ç;őiiijŃăũēă

```
#!/bin/bash
SWD=$(cd "$(dirname "$0")"; pwd)
AFPTOOL=${SWD}/../afptool
IMGMAKER=${SWD}/../rkImageMaker
#GEN_DIR=${SWD}/gen
#OUT_DIR=${SWD}/out
TMP_IMG=firmware.img
UPDATE_IMG=ubuntul404-sfs.img
LOADER=RKLoader.bin

pause()
{
echo "Press any key to quit:"
read -n1 -s key
exit 1
}

echo "start to make update.img..."
# read pack-file packing to update.img
```

(continues on next page)

(continued from previous page)

```
#mkdir -p ${GEN_DIR}
#mkdir -p ${OUT_DIR}
echo ${SWD} $TMP_IMG
$AFPTOOL -pack ${SWD} $TMP_IMG || pause
# merge uboot and firmware
rm -f ${UPDATE_IMG}
$IMGMAKER -RK32 $LOADER $TMP_IMG ${UPDATE_IMG} -os_type:androidos_
↪ || pause
#rm -r ${GEN_DIR}
echo "Making lubuntul404.img OK."
#echo "Press any key to quit:"
#read -n1 -s key
exit 0
```

æʃlæDŕiijŽăĂŖæȚrçnnăZZèaÑäy■echo âĂIJMaking lubuntu1404.img OK.âĂİ
lubuntu1404.imgäyžäjäèeAăŁœĤzăŖŎcŽĎETIJăČŖăŖăăĂĈ

4.2 część 1 Ja i świat

1.èÕůǎꞤŮubuntuæĲĂăŕĚčÿæŮĞăžúçșçżš

iiijL1iiijL'äzÖubuntuaoŸæÚzç;ŚcńZäŸNè;jaĖŮæRŔä;ZçŽDæJĀaŕRèçŸæŮĠgäzúçşzczşşăĂĆ

```
mkdir~/ubuntu-rootfs
cd ubuntu-rootfs
```

Ubuntu Core rootfs

```
wget http://cdimage.ubuntu.com/ubuntu-core/releases/14.04.1/release/  
↳ ubuntu-core-14.04-core-armhf.tar.gz
```

ǽŁŨëĂĚçŽ' æŎěăŎžç;ŚäýŁæŁ'ăŁřăřăžăŤçŽĐçŁ'ŁæIJňăýŇè;; #####
 ĩĭĹ2ĩĭĹ'èğčăŎŇUbuntu Core rootfs tarăŇĚ

```
sudo tar -xzvf ubuntu-core-14.04-core-armhf.tar.gz
```

iijL3iijL'čTšāzŌæĹSāzñēēAčhrootiijNæĹ'ÄäzēēĴÄēēAēČ;ād'šāĴIchrootčŌráčČæĹ'gēāNarmht
 staticēğčēGŁăZĭăĂĆ

```
sudo apt-getinstall qemu-user-static
sudo cp/usr/bin/qemu-arm-static ubuntu-rootfs /usr/bin/
```

```
sudo cp -b/etc/resolv.conf ~/ubuntu-rootfs/etc/resolv.conf
```

```
sudo mount -t proc /proc ~/ubuntu-rootfs/proc
sudo mount -t sysfs /sys ~/ubuntu-rootfs/sys
sudo mount -o bind /dev ~/ubuntu-rootfs/dev
sudo mount -o bind /dev/pts ~/ubuntu-rootfs/dev/pts
```

```
sudo chroot ~/ubuntu-rootfs/
```

```
vim /etc/apt/sources.list
ævi /etc/apt/sources.list
```

```
sudo apt-get update
```

```
SSHii jÑè£IJçlńÇžžéžĚçşžçžšÉIJĂèęAçŤlálřãĀĆ
Networkmanager,ç;ŚçžIJçõaçŘĚçŽĎè;řřäúũăĀĆ
wireless-toolsăĀĆ
ç■L' ç■L'ăĀĆ
```

```
apt-get install è;rážúåŘ■
```

6. èŕç;õçŦlæŁuçŽÿăĚş

adduser UbuntuüijŇçĐúăŔŎæăžæ■ŏæŔŔçd'žèŕç;ŏârĚçăAăĂĆ
èŕç;ŏăÿžæIJžăŔ■çğŕijŽ

```
echo "ubuntu-arm" > /etc/hostname
```

èŕç;ŏæIJŇæIJžăĚăŔçipijŽ

```
echo "127.0.0.1 localhost" >> /etc/hosts  
echo "127.0.1.1 ubuntu-arm" >> /etc/hosts
```

ăĚĂèŕçèĠăĹlæŽt'æŰŕdnsüijŽ

```
dpkg-reconfigure resolvconf
```

èŕç;ŏæŰŭăŇžüijŽ

```
dpkg-reconfigure tzdata
```

7. éĚ■ç;ŏăÿşăŔçèŕČèŕŦ

æŭžăĹăÿŦĂăÿŦ/etc/init/ttyS2.confæŰĠăžŭüijĹæşăă;ŰăŏŸæŰžèŦĐæŰŽă;ŰçşçèŕČèŕŦăÿşăŔçăŔŭăÿžttyS

```
cp tty1.conf ttyS2.conf  
vi ttyS2.conf
```

ăĚŏæŦžăĚŭăĚăŕăŕăçăĈăÿŇüijŽ

```
start on stoppedrc or RUNLEVEL=[12345]  
stop on runlevel[!12345]  
respawn  
exec /sbin/getty-L 115200 ttyS2 vt102
```

8. éĂăĠăĠchrootüijŇçŽt'æŎëexit

9. â■ÿè;çŽÿăĚşæŰĠăžŭçşžçş

```
sudo umount ~/ubuntu-rootfs/proc  
sudo umount ~/ubuntu-rootfs/sys  
sudo umount ~/ubuntu-rootfs/dev  
sudo umount ~/ubuntu-rootfs/dev/pts
```

10. âĹă;ĬçşžçşşéŦĬăČŔ

ăŔČèĂČăŔçăÿĂçŕĠæŰĠăçéŦĬăČŔæĹŦşăŇĚăÿŎëğçăŇĚüijŇăĚĹăŕĚăŎşăğŇéŦĬăČŔëğçăŇĚüijŇăŕĚ
çŽĐæŰĠăžŭçşžçşşüijŇçĐúăŔŎăĚ■æĹŦşăŇĚüijŇăŕşăĂžăă;ăžĚăĂĆ

æşlæĐRiijŽçČġā;TçşzçzşşāRŌiijNæRŔçd'žæŰĞäzŭçşzçzşçl'žéŰt'äy■ād'şiiijNéCčázĹāřséĠæŰřresize
āLEāNžāR■āĀČāLEāNžāR■āRřazēçTĹāçCäyNāŚ;āzd'ēāNŕiijŽ

```
cat /proc/partition
æĹŰ df
```

èNěāĞžçŌřæŰāæşTäyŁç;ŚçŽĐéŰóécŸiijNāRřèČ;éIJĀèçAæL'NāLĹéŌŭāRŰIPāIJřāĹiijNā;EāçCædIJā
dhclient eth0

4.3 çşzçzşçČġā;T

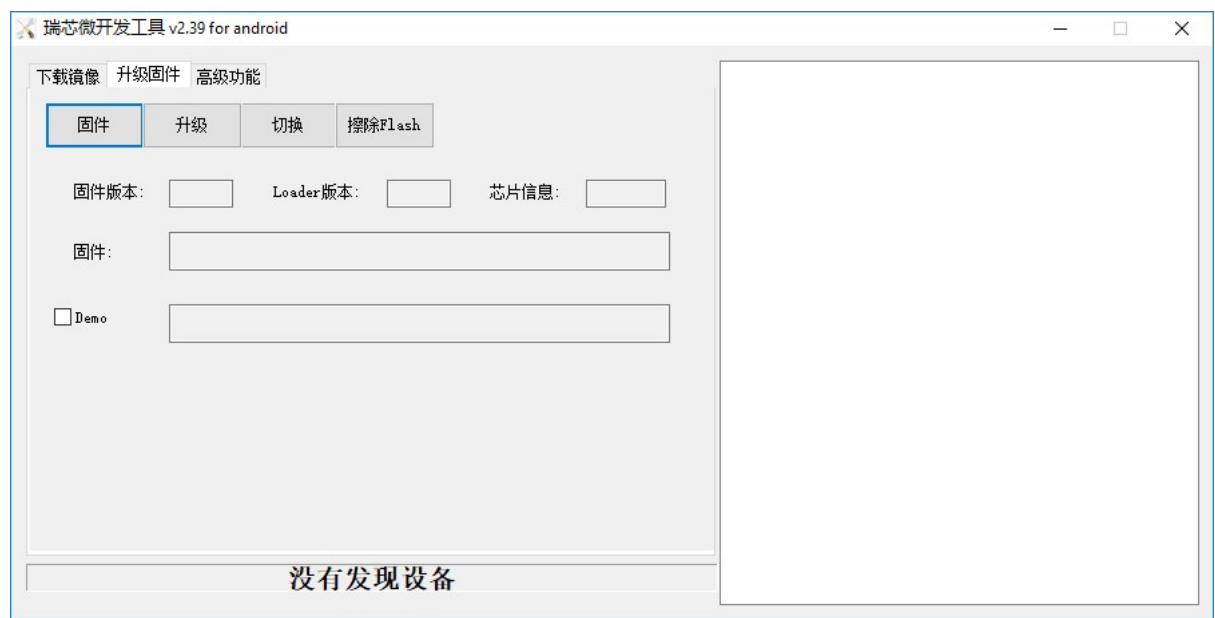
çşzçzşçČġā;TāIJĹWindowsiijNŮbuntuäyđ'äyĹæŞ■ā;IJçşzçzşçéČ;āRřazēāóđçŌřiijNāřsāzNāL■æRŔāĹřçŽ

1. äyNè;Ĳ Release_DriverAssistant.zipiijNèġçāŌNŕiijNçĐŭāRŌèεRēāNéĠNélcçŽĐ DriverInstall.exe āĀČ

äyžāžEæL'ĀæIJĹ'èō;āđ'ĠéČ;ā;ŁçTĹæŽt'æŰřçŽĐēl'sāĹĹiijNèřŭāĒĹéĀL'æNĹ'āĀĹēl'sāĹĹā■yè;ĲāĀiijNçĐ

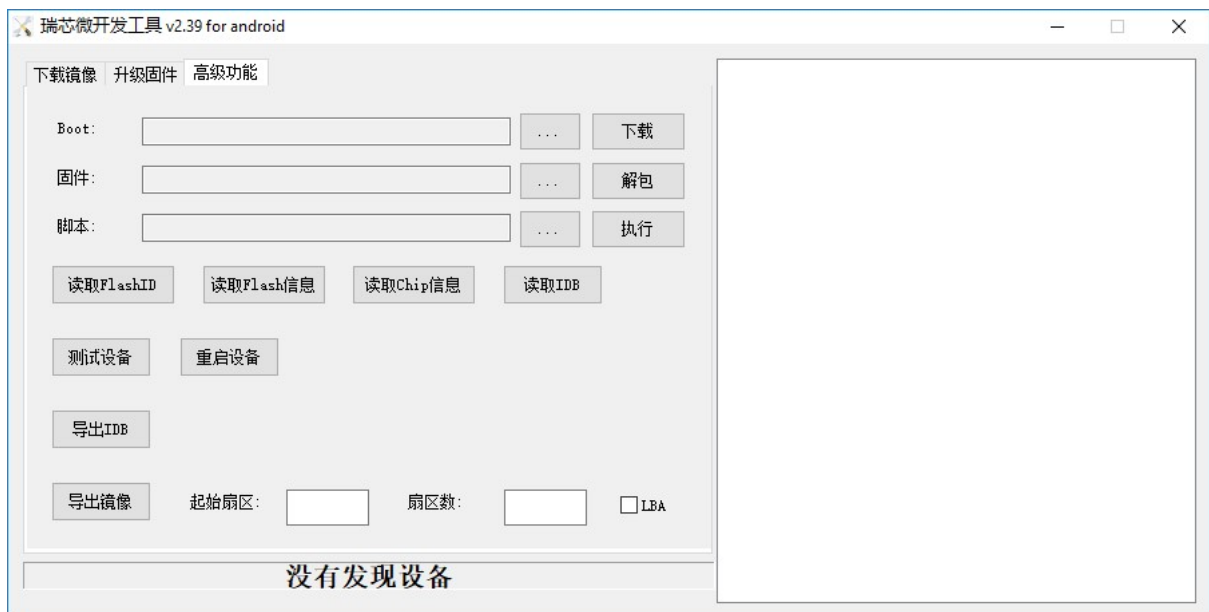
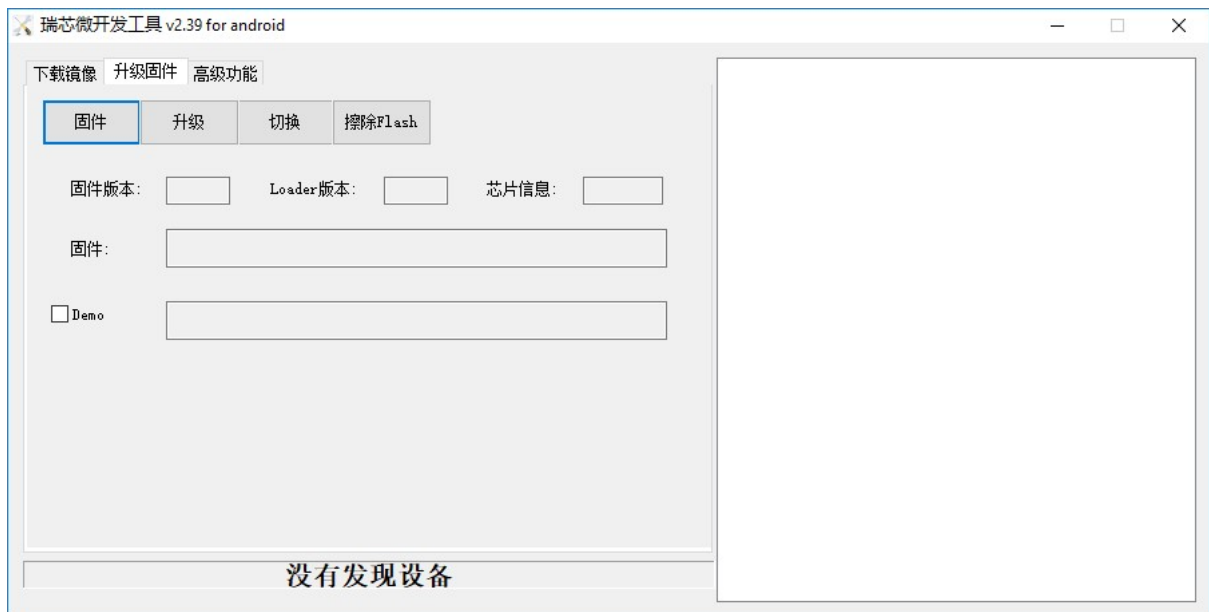
1.1 äyNè;Ĳè;řāzŭāŭēāĒŭiijŽ

AndroidTool_Release_v2.39iijNāIJĹMIQIāōŸæŰžæŰĞæāçéĠNāřsāijŽæRŔä;ŽiijNæĹŰèĀĒçŽt'æŌēāz
1.2 èŁŽāĒēè;řāzŭāRřazēçIJNāĹRāçCäyNāyLçTŕNélcçijŽ



äyL'çġ■æŰžæşTēČ;āRřazēāóđçŌřçČġā;TāLşèČ;iijNā;EæŸřæIJnéāzçŽōäyçTĹçŽĐæŸřçñnāžNçġ■iijNā

1. çČzāĞžāĀIJāŽžāzŭāĀĹæNĹ'éŠō
éĀL'æNĹ'āřzāžTā;āèçAçČġā;TçŽĐéTĹĹāČRæŰĞäzŭ
2. æĹā■āiijŽèŁŽāĒēāĀŸrockusbāĀŽæĹāiijŘ



- iiijL1iiijL æŮ■āijÅUSBçTtæžŘ
 - iiijL2iiijL ä;£çTléTŁā■ŘæLŮāZđā;céŠLæNL'ä;RæAçåd'■éTō
 - iiijL3iiijL éĜ■æŮřèfđæŌěUSBçTtæžŘ
 - iiijL4iiijL ç■L'ā;Ěād'ğçžę3çğŠiiijNçĎūāŘŌéĜLæTŁæAçåd'■éTōāĀĆ
 - iiijL5iiijL PCçnräyŁäijäë;řäzūāZTæčĀætNāLřāLäë;ç;èö;āđ'Ĝ
3. æL'ğëāNāĀIJā■ĜçžğāZžāzūāĀĪæ■ééld'
- ā■şæNL'äyNāĀIJā■ĜçžğāĀĪæNL'éŠōāĀĆāIJā■ĜçžğāŘŌiiijNë;řäzūāRşçnrè£YäijZæYŁçd'žāĀIJæčĀæş

4.4 äÿşāRčërČërT

äÿşāRčërČërTā■şæYřāIJPCäyŁçŽzéZEæĪēā■açşçzçşiiijNçĎūāŘŌāřzāĚūè£ZEaÑèrČërTāĀĆæIJñéazçŽ

1. çañāzūè£đæŌě

ä;£çTlé;ñäÿşāRčëriiijNāřEPCäyŌāijĀāRŚæĪē£ZEaÑè£đæŌěiiijZ

- iiijL1iiijL è;ñäÿşāRčëGNDāIJřçž£äyŌāijĀāRŚæĪēGNDéŠLçŽyè£điiijZ
- iiijL2iiijL è;ñäÿşāRčëTXDè;ŞāĜžçž£äyŌāijĀāRŚæĪēRXéŠLçŽyè£điiijZ
- iiijL3iiijL è;ñäÿşāRčëRXDè;ŞāĚëçž£äyŌāijĀāRŚæĪēTXéŠLçŽyè£điiijZ

2. è£đæŌěāRČæTř

- iiijL1iiijL æşççL'zçŌĜiiijZ115200
 - iiijL2iiijL æTřæ■ōä;■iiijZ8
 - iiijL3iiijL āAĪæ■cā;■iiijZ1
 - iiijL4iiijL āčĜāAŮæāāēĪNā;■iiijZæŮā
 - iiijL5iiijL ætAæŌĝiiijZæŮā
- è£ŽāžZāRČæTřæYřéIJĀèëAāIJĪminicomäyŁéĪcèö;ç;řçŽĎiiijNāşžæIJñäyŁāRĪēIJĀèëAèö;ç;řçññäyĀéa

3. äÿşāRčërČërT

- æ■d'ēāžā■şæYřçşçzçşçŽzéZEçŽĎéĜ■èëAæ■ééld'iiijNéZđ'āžEæNL'çĚğçññāžNçĆzæRŘāLřçŽĎéIJĀèëèö;ç;řçminicomāRČæTřçŽĎāŚ;āzd'ēāNāëCäyNiiijZ
- sudo minicom –s
- æL'ŞāijĀèrëçTÑéĪcāŘŌéĀL'æNĪ'çññäyL'ēāžiiijNçĎūāŘŌè£ZEaÑèö;ç;řç
- æşĪæĎRèö;ç;řçōāōNærTřāŘŌéIJĀèëAäĪā■YiiijNéĀL'æNĪ'çññāžTēāžiiijNāĪā■Yèö;ç;řç

- èø¿ç;ïðáõÑæŕŦäŖÖïjÑéCčázĹåršårEæÍfå■äyŁçŦïijŦçŦĹázŦåĹ■æŖŖåĹŕçŽDè;ñäyšåŖčçŽDåŖæyÄ

```
sudo minicom
```

- çĎúåŖÖåršåŖŕázèèŁŽåĖëåĹŕæÍfå■açŽDçşçzçşéĠŦžEïijŦæşĹæĎŖåĹĹæñaçŽzéŽEçŽDårEçåAäyžåÄĹ

éĹJĀèèAæşĹæĎŖçŽDåĹJŕæŦzïijŽ ##### 1. èŦèPCäyŁæşæĹJĹæŦŦçd'žæÍfå■æĹŦşåŖåĠžæĹèçŽDåŁæ
2. äy■åE■èŕČèŕŦæŦüïijŦéĹJĀèèAåĖĹéĀĀĠžminicomïijĹctrl A +
XïijĹïijŦåE■åĹJæ■čåržæÍfå■açŽDä;ŽçŦïijŦèŦëäy'd æ■èåððæŦŦ;éazžŖæŦžåŖŦïijŦåĹŽäyŦæñä;ŁçŦĹmi
äyšåŖčèènéŦĀïijĹDevice /dev/ttyS0 is locked ##### èğčåEşæŦzæşŦïijŽ

```
ls /var/lock
åĠžçŦŕLCK..ttyS0 subsysèŁŽäyŁæŦŦžäzŦ
kill 0
```

äzŦåŖÖåršåŖŕázèèĠæŦŕèŁŽåĖminicomžEåĀČ

4.5 çşçzçşåEĖæäyæŁóæŦž

æĹJñæççŽDåržåEĖæäyæŁóæŦžçŽDæş■;ĹJåd'ğèĠŦåĹæyžäy'd éČĹåĹEïijŦäyĀéČĹåĹæŦŕåşžæĹJñæş■
1.1çijŦŕšåEĖæäy åĹJĹåEĖæäyæžŖçåAäyŦæĹğèāŦmake menucon-
figžŦåŖÖïijŦåršåŖŕázèçĹJŦåĹŕåržåžŦçŽDåEĖæäyèè■ç;ŦçŦŦéĹçïijŦçŦĹæĹååŖŕázèāŦéĀĹéĹJĀèèAçijŦŕ
åĹJĹçŦĹæĹåèè■ç;ðáõŦåEĖæäyåŖÖïijŦäijŽçŦŦşæĹŖäyĀäyĹéè■ç;ðæŦŦžäzŦïijŦçŦĹèèèèè■ç;ðæŦŦžäzŦåEĖæäy

```
#!/bin/bash

#####
##
## You need to change ANDROID_ROOT to real Android SDK path !!!!!
##
#####

MIQI_BUILD_DIR=${PWD}
MIQI_KERNEL_DIR=${PWD}/../rockchip-kernel
MIQI_ROOTFS_IMG_FULLPATH=

ARCH=arm
CROSS_COMPILE=arm-eabi-
export ARCH CROSS_COMPILE
export PATH=${MIQI_BUILD_DIR}/prebuilts/gcc/linux-x86/arm/arm-eabi-
→4.6/bin:$PATH

build_kernel()
{
(
cd $MIQI_KERNEL_DIR
make ARCH=arm xxxx_defconfig
make ARCH=arm -j8 rk3288-MiQi.img
```

(continues on next page)

(continued from previous page)

```
./mkbootimg --kernel arch/arm/boot/zImage \
--ramdisk ${MIQI_BUILD_DIR}/images/rootfs.cpio.gz \
--second resource.img \
--output recovery-linux.img
)
}
build_kernel
```

æşlæĐRrijŽ

1. build_kerneaĠ;æTřäy■ārzážTčŽĐmake ARCH=arm xxxx_defconfigèŁŽäyĀèaŇäy■ijŇxxxx_defconfi
2. èĐŽæIJŇæIJĀāL■élcçŽĐāRŸéGRèøç;đāĀĆæIJŇäžžæŸřāEāEĒæäyæžŘçāAžžēāRLbuildæŮĠžüzāđ'

1.2çijŮèřSmodes

āIJlāEĒæäyæžŘçāAārzážTčŽĐæŮĠžüzçŽōā;TäyŇæL'gèaŇāçCäyŇāŚ;äzd'èaŇijŽ

èŇæLééTŽ āIJçijŮèřSéSç;ä;■ç;óéTŽèřāĀiijŇāLŽārEbuildæŮĠžüzāđ'žäyŇçŽĐprebuildèĀČèŁGāŮŽ

```
make modules
mkdir modules_install
make INSTALL_MOD_PATH=./modules_install modules_install
```

æŇuāLřāijĀāRŚçL'LāRŮrijŇæyĒæčŽælaāIŮāōL'èčĒçŽōā;T(èřçŽōā;TāRŇæIJL'éSç;æŮèijŇāijŽā;śāS■

1.3 éĒ■çijŮuart3,ā■şārzážTçşçzçşäy■ttys3äyşāRçiiŇNéIJĀèeAèŁdæŮèéçdæŮgä;ŁçTlāĀĆ

1.3.1éĒ■çijŮ DTS èŁĆçĆz

æŮĠžüzū kernel/arch/arm/boot/dts/rk3288.dtsi äy■āušçzŘæIJL' uart
çŽyāĒşèŁĆçĆzāōŽāzL'rijŇ

āçCäyŇæL'Āçđ'žiiŇŽuart_gps: serial@ff1b0000 { compatible = āĀIJrockchip,serialāĀĪ;
reg = <0xff1b0000 0x100>; interrupts = ; clock-frequency = <24000000>; clocks =
<&clk_uart3>, <&clk_gates6 11>; clock-names = āĀIJsclk_uartāĀĪ, āĀIJpclk_uartāĀĪ;
current-speed = <115200>; reg-shift = <2>; reg-io-width = <4>; dmas = <&pdma1 7>,
<&pdma1 8>;#dma-cells = <2>; pinctrl-names = āĀIJdefaultāĀĪ; pinctrl-0 = <&uart3_xfer
&uart3_cts &uart3_rts>; status = āĀIJdisabledāĀĪ; };

æşliijŽuart_gps āIJlèrèæŮĠžüzçŽĐ aliases èŁĆçĆzäy■ècnāōŽāzL'äyžiiŇŽserial3 =
&uart_gps;

çTlāēLūāRlēIJĀāIJĪ **kernel/arch/arm/boot/dts/firefly-rk3288.dts
**æŮĠžüzüāy■æL'ŞāijĀæL'ĀèçAā;ŁçTlçŽĐèŁĆçĆzā■şāRřiiŇ

```
æCäyNæL'Äçd'žiiž&uart_gps { status = âÄJokayâÄI; dma-names = âÄIJ!txâÄI,
âÄIJ!rxâÄI;pinctrl-0 = <&uart3_xfer &uart3_cts>; };
```

1.4 äçdæûzél'sälíijl5Gíijl'

æUäzæIJzéIJÄèeAç;ŠçzIJäijæ;ŠäLšèÇ;iiijNèÄNäóðetNäód'çÓráçCäy■áRÜäLrè;Čäd'Ž2.4GHZ
wifiäfaaRûçŽDäzšæL'riijNéCčázLèÄČèŽSéGčTí5GHZ wifi
äfaaRûäÄČäIJç;ŠäyLèt'■äzr5Gç;Šä■aaRÖriijNäRŠçÖréIJÄèeA8811AUél'sälíijNèÄNäIJlâEËæäyæžRčäAa
menuconfig,ázúæIJlâRŠçÖr8811AUél'sälíijNéCčázLéIJÄèeAèGlaúsæûzälææZäÖžriijNèfGçlNæCäyNíijL

1äÄAärE8811AU.tar.gzriijLäR■aa■UäRräzèèGtèaNäfóæTžriijL'æÜGäzúègčäÖNèGšâÄlâEËæäyæžRčäAa

2äÄAmake menuconfigriijNæL'çälíDevice drivers->network device support->wireless
lan->usb zd1201 based wireless device support**éÄL'äyž*

3äÄAäfóæTžäÄlâEËæäyæžRčäAaNĚ/drivers/net/wirelessâÄlèûrâ;DäyNçŽDKconfigriijNäžTèaNäçdäl
source âÄIJdrivers/net/wireless/8811AU/KconfigâÄI

4äÄAäfóæTžäÄlâEËæäyæžRčäAaNĚ/drivers/net/wirelessâÄlèûrâ;DäyNçŽDmakefileriijNäžTèaNäçdäl
obj-\$(CONFIG_8811AU) +=8811AU/

5äÄAäLrâEËæäyæžRčäAaNĚä;■ç;ômake menuconfigriijNéÄL'äy■Device drivers-
>network device support->wireless lan->8811AU USB wifiÄL'äyž*

6äÄAäflâ■ÿ makeázNäRÖäÄC æšlæDRriijŽäfóæTžäóNäEËæäyärÖéIJÄèeAèG■æÜçijÜerSâEËæäyriij

4.6 çšžçžšçÓráçČéĚ■ç;ó

æ■d'äd'ĐçÓráçČéĚ■ç;óæNĚçŽDæYræIJnéazçŽóæL'ÄéIJÄèeAçTíälíçŽDäRĐäyè;rázúçÓráçČäzèäRĚ
OpencvriijNrosriijNOptitrackriijNEigenriijNmaivosç■L'äÄC

Opencv

riijS.áoL'èçĚäzd'äRL'çijÜerSâuèäĚuèSç;riijNáoL'èçĚèGčlNæCäyNíijŽ

ä;NíijŽáoL'èçĚçL'LæIJnäyžgcc-linaro-arm-linux-gnueabi-hf-4.8-
2014.04_linux

- ç;ŠäyLèÖüäRÜgcc-linaro-arm-linux-gnueabi-hf-4.8-2014.04_linux.tarriijNäIJlæL'ÄäIJlçŽóä;TèfZèaNè

```
sudo tar -xvf gcc-linaro-arm-linux-gnueabi-hf-4.8-2014.04_linux.  
→tar
```

- äIJl/usr/localäyNæÜräzžäyÄäyæÜGäzúäd'žriijNä;ççTíäS;äzd'èaŇ

```
sudo mkdir arm-toolchain
```

- `arm-linux-gnueabihf-4.8-2014.04_linux` `arm-toolchain`

```
sudo cp -r /home/wl/arm-toolchain/gcc-linaro-arm-linux-gnueabihf-4.8-2014.04_linux /usr/local/arm-toolchain
```

- `export PATH=$PATH:/usr/local/arm-toolchain/gcc-linaro-arm-linux-gnueabihf-4.8-2014.04_linux/bin`

```
source /etc/profile
```

- `arm-linux-gnueabi-gcc -v`

```
source /etc/profile
```

- `arm-linux-gnueabi-gcc -v`

2. `arm-linux-gnueabi-gcc`

`arm-linux-gnueabi-gcc`

```
sudo apt-get install cmake
```

3. `OpenCV`

(1) `arm-linux-gnueabi-gcc`

- `sudo cmake-gui`
- Specify the generator for this project: `Unix Makefile`
- `Target System: linux, Compiler: arm-linux-gnueabi-gcc`
- `Config`

- `çDúãRÕãrsæYřäfôæTzçzçèL'sãNzãşşçZĐëCíáLeãĀijĭijNãŌZæŌL'æşŘäzZãĀijĭijZ
WITH_OPENCLãĀĀWITH_TIFFãĀĀBUILD_OPENEXRãĀĀWITH_OPENEXRãĀĀWITH_CUDA
ãzúãfôæTzCMAKE_INSTALL_PREFIXçZĐãĀijäyžèGłãũsæCşèçAçZĐèũrãĭDĭijNè£ŽéGÑæĹSèöçç;
opencv/ãĀC`
- `çDúãRÕæL'SãijĀ/home/wl/arm-linux-opencv-binariesçZŌãĭTäyŊçZĐCMakeCache.txtæŪGãzũãfôæTz
CMAKE_EXE_LINKER_FLAGSãŌŌşæİëäyžçł'žĭijNãĹãäyŁ-
lpthread -lrt -ldl CMAKE_INSTALL_PREFIX:PATH=
XXXXXX(ãŌL'èçĒèũrãĭD) èğĀäyŁãĀC çDúãRÕãrsãRřäzææL'gèãŊsudo
make,æşlæDŘèfŽéGÑèçAãĬĬçĭjŪèrŞçTşæĹRæŪGãzũçZŌãĭTäyŊæŞ■ãĭĬĭijNè£ŽéGÑæĹSçZĐèũrãĭD
linux-opencv-binaries`

- gnuabihi/bin/ld:~/.3rdparty/lib/libzlib.a(crc32.obj):~ relocation~ARM_THM_MOVW_fPIC
- æslæDRålRåEüäy■ÇDlibzlib.aæÜGäzüüijNæLSäznéIJæeAålIJlä;æNĞáoŽçŽDäzNëfZålÚæÜGäzüe, fsigned-char~W~Wall~WerroræŽæ■cäyž C_FLAGS = -fsigned-char -O3 -fPIC -W -Wall -Werror årseąNäzErijNålIJlëfZäzNåRŌæLŪëÄËäzNål■Éç;årRëÇ;åĞžçŌråRÑæäüæCĚ
- årŌÉícéIJæeAéG■æŪrålRrconfigæ■éld~ijNåfóæTžCMaKeCaChe.txtæÜGäzüüäfóæTžéGÑë;žçŽDäyd

```
sudo make install
```

åIJæ■d'åĖŁåAŽèrt'æÿŎiijŽ

```
/usr/local/arm-opencv/lib
```

```
sudo gedit /etc/bash.bashrc
```

```
PKG_CONFIG_PATH=$PKG_CONFIG_PATH:/usr/local/opencv-arm/lib/
↪pkgconfig
export PKG_CONFIG_PATH
```

- cijŮerSætNerTijŽaAĞaĆä;äüüščzRæIJL'äyÄäyImain.cppâĜ;æTřæŮĞäzûäzErijNëŁZâEëäEüæL'ÄâIJćZDæŮĞäzûäd'zæL'ğëaNäyNéIććZDâŚ;äzd'èŁZëaNcijŮerŚtijŽ

```
arm-linux-gnueabihf-g++ main.cpp -o hh -lpthread -lrt `pkg-
config --libs --cflags opencv`

arm-linux-gnueabihf-g++ -c main.cpp -o main.o
main.o
cp main.o /usr/lib/
cp main.o /usr/lib/
```

-çijÛerŚæĹŖăĹşăzŃăŖŌiijŃăŕśăiijZăŖŚçŌŕçŤşæĹŖăzEăyĀăyĹăŖ■ăyžhhçŽĎăžŃeŧZăĹuăŪĠăzŭiijŃe
5.çğzæd'■ăĹŕarmăžşăŖăyĹ

```
-æŁŁarm-opencvæŃuèt'İâĹĹARMäy■çŽĎ/usr/localæŰĞäzûäd'zäyŃ,æŁŁarm-linux-  
opencv-binariesæŃuèt'İâĹĹARMäy■çŽĎ/home/wlæŰĞäzûäd'zäyŃăĀĈăzŃăĀŔŎétŃērAçŽĎæŰzæşŦăŠŃăyŁ  
æĪĴăăŔŎæŦŦ'äyĴOpencvçŽĎçğzæd'■ăŕşăŏŃæĹŔăžEĭĭjAĭĭjAĭĭjA
```

ROS

1. ééŨáĚŁćŦĩminicomèŁđæŦěaijĀăŦŚſæłſiiĴŦæſæłJL'ėŨŦěćŸăĂĆ

ä;£çŤlçŽĐăŚ;äzd'èąŃii;Ž

```
sudo minicom
```

2. æŌēçİÄaİJÍaijÄaRSæİĖäyŁēçŞăĖĕăŞjăzd'eaŊæŭăŁăē:řăžŭăžŘăĹsources.list

ä;£çŤlçŽĐăŚ;äzd'èąŃiijŽ

```
$sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu trusty_↵
↵main" > /etc/apt/sources.list.d/ros-latest.list'
```

3. ĆĐŰǎŘŎèó¿ċj'óǎřĚéŠě

ä;£çŤlçŽĐđŚ;äzd'èąŃ:

```
$ wget http://packages.ros.org/ros.key -O - | sudo apt-key add -
```

4. æŌěäÿŊælěèŁŞăĚě

```
$ sudo apt-get update
```

5. çĐúăŔŎè¿ŞăĚě

```
$ sudo apt-get install ros-indigo-base
```

6. æŌěäyŊælēăĹiăġŊăŊŮrosiijŽăĹlæñəēĹŞăĔĕ

```
$ sudo rosdep init
$ rosdep update
```

7. ælJÅåŘŮéĚ■çj"çŮřácČiijŽ

è;ŞăĚěïjŽ

```
$ echo "source /opt/ros/indigo/setup.bash" >> ~/.bashrc
```

ä;ŁçŒřăĈĈăRŸéĜŔèő;ç;őĉńŃă■șĉŦșæŦĹè;ȘăĚëiijŽ

```
$ source ~/.bashrc
```

ǎŘřèČĵéAĞǎŁřçŽĎéUőécŸ

1. åŠ;äzd'èaÑéTfåžęŻĐéUóécŸiijŽăIJlæûzâŁăe;făzúæžRăŠÑěŮ;ç;őârEęŚěe;ŚăĚěăŠ;äzd'èaÑæŮiijŃŃ
2. UpdateæŮüäijŽăGžçŮřăŸĂžŽç;ŚçńŽhităŸ■ăžEřijŃăŔléIJĂèęAăd'ŽèřTăGăæñăęĜşăĹŔăĹşă■şăŔřăĂ
3. InstallæŮüăęĆăđIJăŮĹ'èçĚăIJĹ'fullçĹ'ĹăIJňăŔřèÇ;äijŽæŮăęşTlocateřijŃăĹ'ĂăžčăŸ■čęAăŮĹ'èçĚfull
4. InstallæŮüèęAęşlăĐŔrosæŤŕăŃAçŽĐubuntuçĹ'ĹăIJňiijŃăęĆindigoæŮü14.04řijŃěĂŃ16.04ăĹŽéIJĂ
5. InstallæŮüăŔřèÇ;äijŽæIJĹ'ăŸĂăžŽçŽŮăăĜgetăŸ■ăĹřijŃăIJlinstallăŸĂæñăă■şăŔřăĂĆ

6. `apt-get install ros-noetic-catkin`

4.7 Optitrack

az Ögithubäy Läy Nè; ; a ÖNçijl' a NĖijŽ

1. `cd /home mkdir catkin_make`

```
cd /home mkdir catkin_make
cd /catkin_make mkdir src
```

2. `cp -r /home/src catkin_make`

```
cp -r /home/src catkin_make
cd .. catkin_make
```

4.8 EigeniiNmavros

1. `make`

2. `make install`

```
make
make install
```

4.9 äzd'a RL'çij Üer SáŽÍ

1. `make`

äJläy Äçg èðaçð ÜaIJç Öřác Čäy èĚŘea NçŽDçij Üer Šćl Náz Rñj Nè Čçij Üer SáĞž äJlä Rēad' Üäy Äçg èĚŘea


```
PATH=$PATH:/home /äÿÑè; ;/gcc-linaro-arm-linux-gnueabihf-4.8-2013.10_  
↳linux/bin  
export PATH
```

åĚüä;ŞæäijäijRä;Iæ■ōiijŽ PATH=\$PATH:/iijLä;äæL'ÄèğčăŌŇázŇăŘŎ
çŽĎăüăĀĚüăŞ;æL'ĀăIJçŽĎëŭă;ĎiijL/bin export PATH

æŸŞéŤŽçĆziiž

çňňäŸĀăŇçŽĎPATHHPATHăzŇăL■çŽĎç■L'ăŔŭăŭăăŔşăſl'ă;ğăŸ■èçAă■ŸăIJçl'žæäijiiijŇäŸ■çĎŭăĚ

3.1.4ă;ſçŎŕăçĈăŔŸéĜŔ çŤşæŤĹ

```
~$ source .bashrc
```

3.1.5 æŧŇërŤ

```
~$ arm-linux-gnueabihf-gcc -v
```

æĹëéŤŽiiž

```
arm-linux-gnueabihf-gcc: error while loading shared libraries:_  
↳libstdc++.so.6: cannot open shared object file: No such file or_  
↳directory
```

ăŎşăŽăiiijŽ64ă;■çŽĎçşçzşşiiijŇçijžăŕŤ lib32stdc++6èĚŽăŸlăŇĚ

èğčăĚşiiijŽ~\$ sudo apt-get install lib32stdc++6

ăĚ■æŧŇërŤăĀĀ~\$ arm-linux-gnueabihf-gcc -v äĜžçŎŕçŽŸăžŤçL'ĹæIJňăŔŭ

3.2ézŸèöd'çL'ĹæIJňăŸŇè; ;

éĜĜçŤĹăŇĜăzd'ézŸèöd'ăŸŇè; ;çŽŸăžŤăŭăăĚüăŞ;

```
sudo apt-get install gcc-arm-linux-gnueabihf g++-arm-linux-gnueabihf
```



```
arm-linux-gnueabi-g++ examples/cl_convolution.cpp test_helpers/Utils.cpp -I. -include -std=c++11 -mfpu=neon -L. -larm_compute -lOpenCL -o cl_convolution ##### 4.2.1cl_convolution ~$arm-linux-gnueabi-g++ examples/cl_convolution.cpp test_helpers/Utils.cpp -I. -include -std=c++11 -mfpu=neon -L/home/dengkai/äÿÑè;;/arm_compute-v17.03.1-bin/lib/linux-armv7a-neon-cl -larm_compute -L/home/dengkai/äÿÑè;;/arm_compute-v17.03.1-bin/-lOpenCL -o cl_convolution
```

æLëéŦŽ:arm_compute not found opencv not found

ãŒšãŽäiijŽarmcomputerãŠNopencläfl'äÿlãžŞëûrã;Đäÿ■ãrż

ëğçãEşiiijZæL;ãLrëfZäfl'äÿlãžŞçŽĐä;■ç;örijNãEüã;ŞãRĆçEğäzëäÿNãzççãA

```
arm-linux-gnueabi-g++ examples/neon_convolution.cpp test_helpers/Utils.cpp -I. -include -std=c++11 -mfpu=neon -L/home/dengkai/äÿÑè;;/arm_compute-v17.03.1-bin/lib/linux-armv7a-neon -larm_compute -L/home/dengkai/äÿÑè;;/arm_compute-v17.03.1-bin/-lOpenCL -o neon_convolution
```

äzşãRãzëãIJlarm_compute-v17.03.1-binçZöã;ŦäÿÑèŒüã;ŦçijŦëŦSãë;çŽĐneon_convolutionæŦĞäzŦ

```
gcc examples/neon_convolution.cpp utils/Utils.cpp -I. -include -std=c++11 -mfpu=neon -larm_compute -larm_compute_core -o neon_convolution
```

æşlæĐRiijŽ linux-armv7a-neon-clãŠN linux-armv7a-neonçŦëæIJL'äÿ■ãRÑèŒrã;ŦüãŁŒãŦž

5. äijäë;ŞëĞşæİ£ã■æè£RëãŦ

5.1çŦŦèĐŞpcçžŁçñräÿŒæİ£ã■äiijLrk3288iijLminicomè£đæŒŒ

```
~$sudo minicom äzNãRŒŒè;ŞãEëæIJñæIJžãŦEçãAäŁŒãŦž
~$minicom -s
```

5.1.1Serial port setupéÄL'éäzéĚ■ç;Œ

éÄL'æŦŦ'ëĚ■ç;ŒéäziiijNë£ŽéĞŦæLŠäzñäÿzèçAéĚ■ç;ŒSerial port se-
tupéÄL'éäziiijNäÿNéİçæŦŦãžæIJñëĚ■ç;Œ

```
âL'ă A - Serial Device : /dev/ttyS
âL'ă B - Lockfile Location : /var/lock
âL'ă C - Callin Program :
âL'ă D - Callout Program :
âL'ă E - Bps/Par/Bits : 115200 8N1
âL'ă F - Hardware Flow Control : No
âL'ă G - Software Flow Control : No
```

(continues on next page)

(continued from previous page)

```
âL'ă      Change which setting?
```

```
+-----+  
↩-----+
```

AéĀL'éąŻSerial Device æāzæ■ōāĹSāznçŽDāyśāRćæİēijŃāęĆæđIæŸr-
COM1æĹSāznārséĀĹæŃI'ttyS0iijĹäy■ēĹGāRŌēİcèrt'çŽDāGžéTŽāzşæIJĹäĭŃād'ŪiijĹiijŃCOM2āRćéĀĹa
FéĀĹ'éązäyĀāōŽèęAæTžäyžNOiijŃäy■çDūçzĹçnrāRİèÇ;æĹŞā■rāzŌäyŃä;■æIJzāRŚèĹGæİēçŽDāĹæ.

5.1.2ä;ŁçTĪminicom

æĹēēTŽāRĹaRrèÇ;āGžçŌřçŽDēUōécŸiijŽ

äyśāRćècñéTĀiijĹDevice /dev/ttyS0 is lockediijĹ èğčāEşiijŽ

```
~$ ls /var/lock  
LCK..ttyS0 subsys  
:~$ kill 0  
~$ ls /var/lock  
subsys  
~$ sudo minicom  
Welcome to minicom 2.3
```

5.2çTtèDŞpcçzĹçnräyŌarmāzşāRrsshèŁđæŌë

5.2.1æĹĹā■āzTèrēäyŌpcāĹĹāRŃäyĀäyĹāsĀāşşç;SiiijŃæçĀæşşæŸrāRçèAŹéĀŽiijŃāRrāzè

```
~$ ssh root@IP
```

äĭŃāęĆ

```
~$ ssh root@192.168.1.153
```

5.2.2æĹēēTŽiijŽ

```
ssh: connect to host 192.168.1.153 port 22: No route to host
```

āŌşāZāiijŽipāĹrāĹĀäy■ārż

èğčāEşiijŽ

a.æĹŞāijĀminicomçnrīijŃèĭŞāĒë

```
~$ ifconfig
```

èŌuā; ŪipāIJrāĀĀijĹāŁāāŁĒāIJlēŁZāĒēminicomāy■è; ŠāĒēēēēāŠ; āzd'ĀijNāIJlčTtēĎŠpcāyNē; ŠāĒēēēēāŠ
b.āIJlčTtēĎŠčZĹčnřē; ŠāĒē ~\$ ssh root@IP āzNāŘŌæŘŘčd'žē; ŠāĒēāřEčăĀijNæ■d'ād'ĎārEčăĀäyžæĹ.

5.2.3āŁŌæTzārEčăĀ:ēŁZāĒēminicom(sudo minicom)

è; ŠāĒēāzēäyNæNĠāzd'ĀijŽ

```
root@mqmaker:~# passwd root
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
```

5.3 sshéŠ;æŌēāzšāŘrāzēāŘŌāijāè; ŠæŪĠāzŪ

```
scp /home/dengkai/lingd/upload/cl_convolution.o root@192.168.1.
↪153:/home/dk
scp /home/dengkai/lingd/upload/neon_convolution.o root@192.168.1.
↪153:/home/dk
```

6. æĹā■āæŁ'gēāNāzd'āŘĹ'čijŪērŠāŏNčŽĎæŪĠāzŪ

6.1ēŁŘēāNāŘræŁ'gēāNæŪĠāzŪ

```
~#./cl_convolution
```

6.1.1æŁēēTŽĀijŽ

```
~#./cl_convolution
terminate called after throwing an instance of 'cl::Error'
what(): empty
libarm_compute.so not found
```

ēġčāEšĀijŽ

```
scp -r /home/āyNē; ;/arm_compute-v17.03.1-bin/lib root@192.168.1.
↪153:/usr/local/lib
```

ārEēIJĀēēAčTĹāĹčŽĎāžŠæT;ēĠšarmārāzāžTæŪĠāzŪād'žāyN


```
cl_convolution: ELF 32-bit LSB executable, ARM, EABI5 version 1
↳ (SYSV), dynamically linked (uses shared libs), for GNU/Linux 3.2.
↳ 0, BuildID[sha1]=e2d66d68a52b4d09a0a934980b934c3a79f0b8c7, not
↳ stripped
./cl_convolution: /usr/lib/arm-linux-gnueabi/libstdc++.so.6:
↳ version `GLIBCXX_3.4.20' not found (required by ./cl_convolution)
```

6.2 ÆRèaŃçz\$ædlJ

äĴŃăŔèŔèaŃăççaoăĴZèŦăZđ

```
"Test passed"
```

7. ælJňăĴŕçijŮerŚ

7.1 éĚ■çĳoăũeăĚũgităĂAg++ăĂAsconsĳijŽ

```
sudo apt-get install g++ git scons
```

7.2 äŷŃèĳComputeLibrary

```
git clone https://github.com/Arm-software/ComputeLibrary.git
```

7.3 ælJňăĴŕçijŮerŚĳijŽ

```
cd ComputeLibrary
scons Werror=1 debug=0 asserts=0 neon=1 opencl=0 examples=1
↳ build=native -j`
```

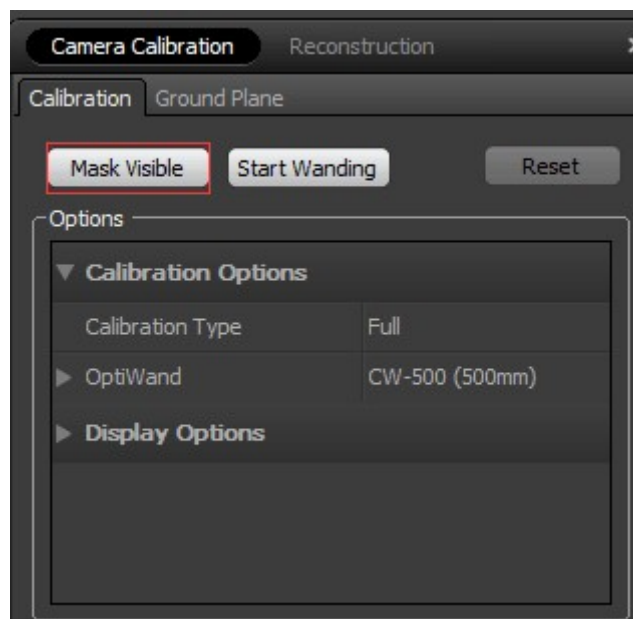
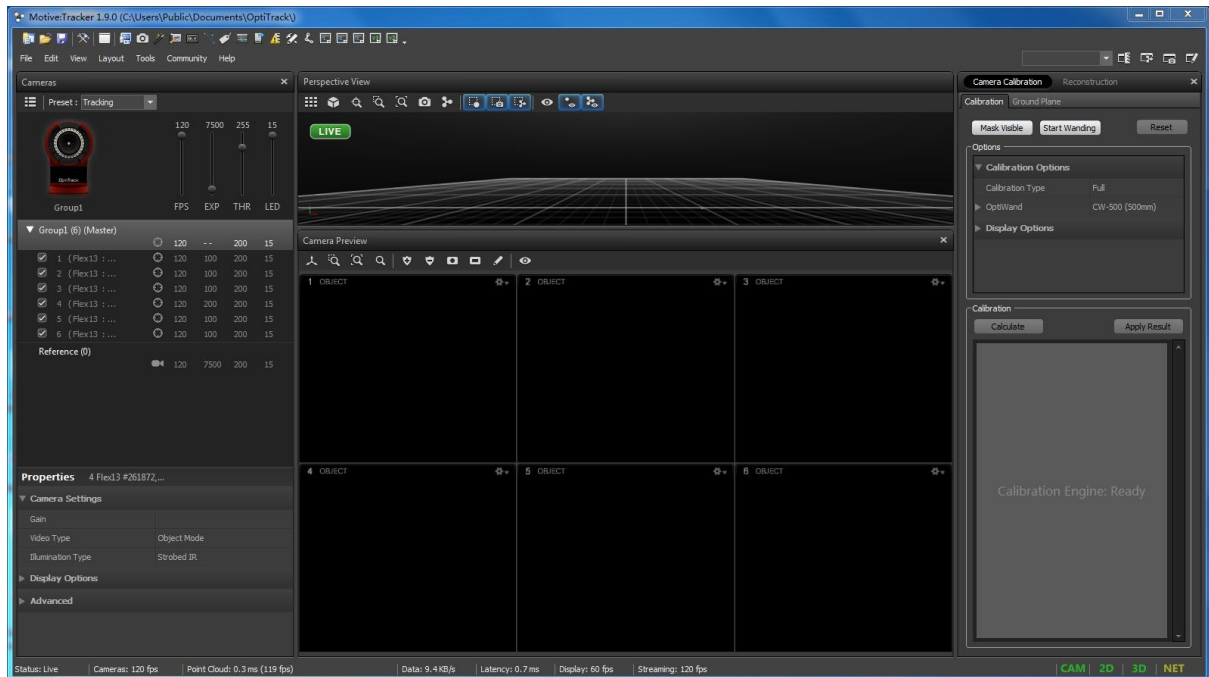
4.10 motiveéĚ■çĳo

1. æăGăoŽçŽŷăĚşæŮĜæaç

1.1 äŦă■ŶăũeçĴŃ

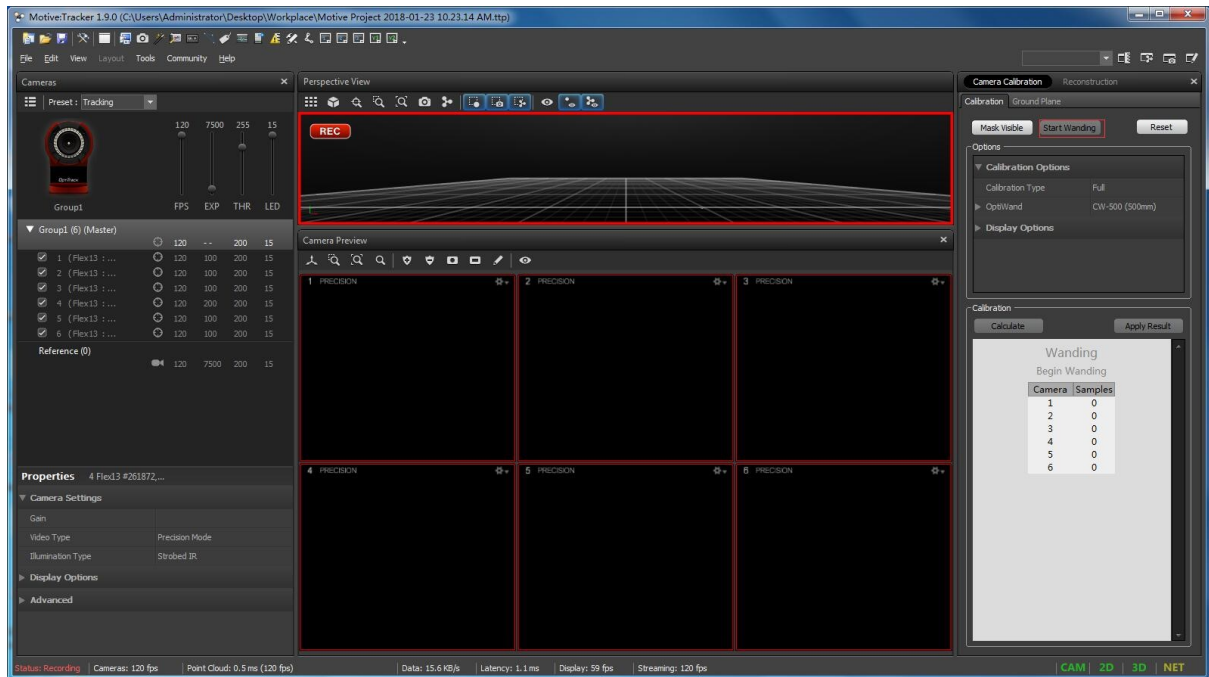
æĴŖăĳĂMotiveèĳfăzũĳŃçĴŮăŔçæŶĴçđ'žăeĆăŷŃçŦŃeĴc
éĂĴæŦŦăŔşăŷĴæŮžăŃžăşşCalibrationçŽoăĴŦăŷŃçŽĐMask
bleĳĳŃăĜžçŮŕăŦă■ŶăŕžerĴăeĴĳŃeĂĴæŦŦŦYesăŦă■ŶăĴŕçŽŷăžŦăĳ■çĳoăĂĆ

Visi-

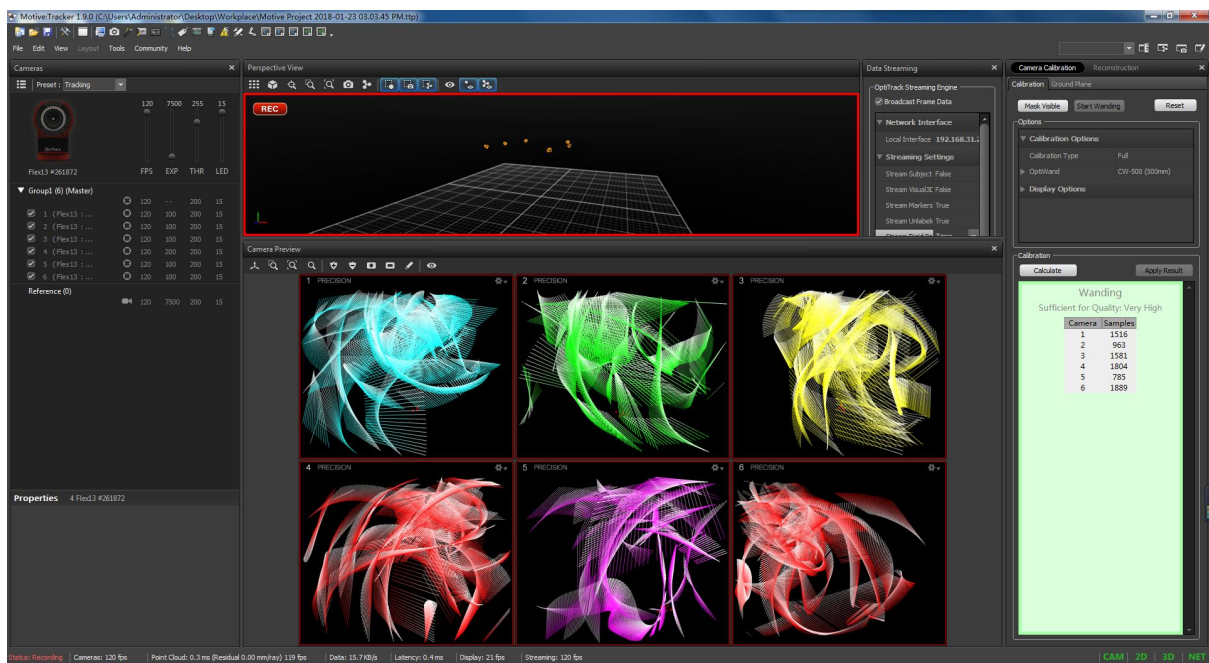


1.2 3D Calibration

3D Calibration is a process used to calibrate multiple cameras in a 3D environment. It involves capturing data from multiple cameras and using it to create a 3D model of the environment. This process is essential for applications such as motion capture, robotics, and virtual reality.



3D Calibration is a process used to calibrate multiple cameras in a 3D environment. It involves capturing data from multiple cameras and using it to create a 3D model of the environment. This process is essential for applications such as motion capture, robotics, and virtual reality.



1.3 3D Calibration

3D Calibration is a process used to calibrate multiple cameras in a 3D environment. It involves capturing data from multiple cameras and using it to create a 3D model of the environment. This process is essential for applications such as motion capture, robotics, and virtual reality.

Camera Calibration

Reconstruction

Calibration

Ground Plane

Ground Plane Calibration Square

Vertical Offset (mm)0

Set Ground Plane

Ground Plane Refinement

Vertical Offset (mm)20

Refine Ground Plane

Capture Volume Translation

▼ Capture Volume Translation (mm)

X	0
Y	0
Z	0

Apply Translation

Capture Volume Rotation

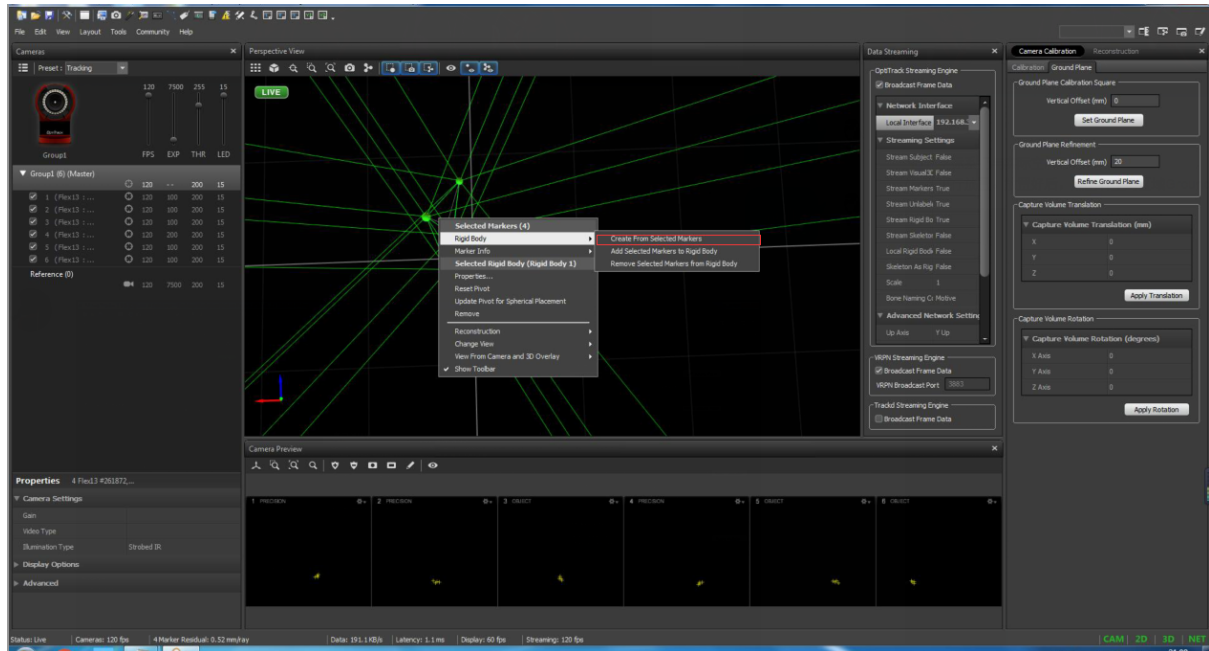
▼ Capture Volume Rotation (degrees)

X Axis	0
Y Axis	0
Z Axis	0

Apply Rotation

1.4žžčŇŇĹŽä;Š

ăĖĖčđæĬžæŤ;ăĖĖăăă;ĬĬăŇžăššăĂĈéĀĹ'ăy■æŤđăĈŖăd't'æŇ■æŤđăĹŕčđæĬžăyĹčŽđæĹ'ĂæĬĹ'æăĈ
BodyăyŇčŽđCreate From Selected MarkersiijŇăžžčŇŇĹŽä;ŠăĂĈ



1.5ěőĹč;őMotive

éĀĹ'æŇĬ' ViewăyŇčŽđRigid Body Properties

čđŮăŖŖŖăĖ■éĀĹ'æŇĬ' ViewăyŇčŽđData Streaming

ăĬĬĬData Streamingăy■éĀĹ'æŇĬ' Local InterfaceăyŇæŇĹ' éăžăy■čŽđæĬŇăĬĬŖăőăŖĈ

ăĖ■ăĖStream Rigid BodiesěőĹč;őăĹŖTrue

æĬĬăŖŖŖăĖMulticast InterfaceăŤžăĹŖ224.0.0.1

2.éĖ■č;ő

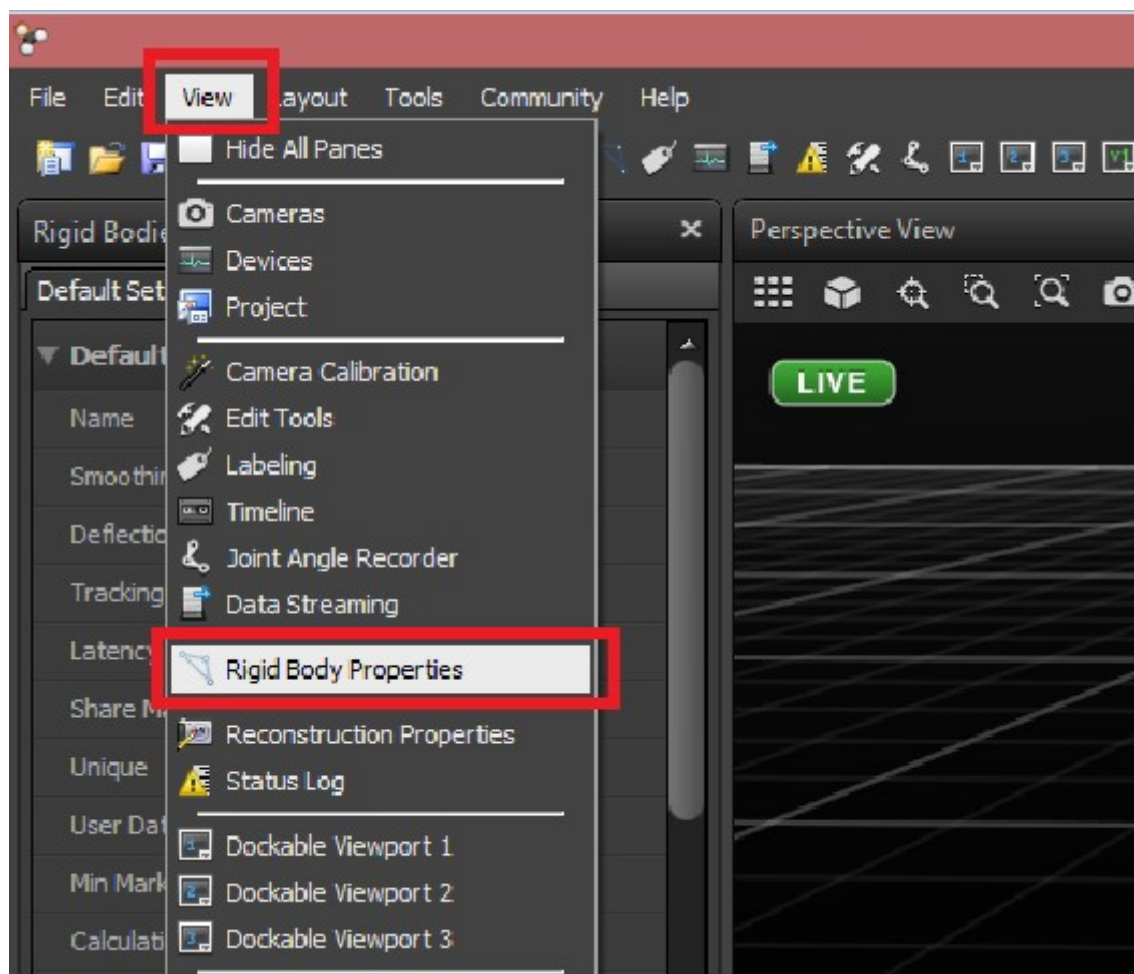
(1)ăĬĬčŇŇăyĂăyĹăŤ;ăžd'ăqŇčĬŮăŖĈčĹšăĖăžăyŇăŤ;ăžd'ăqŇiijŽ

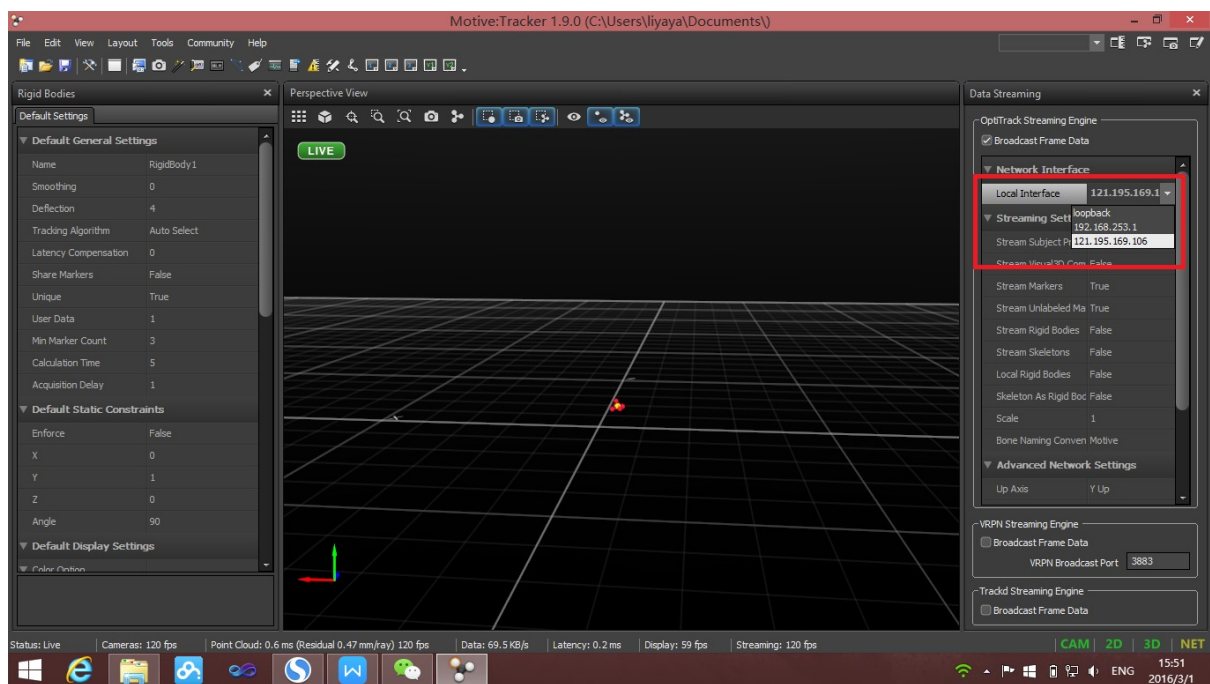
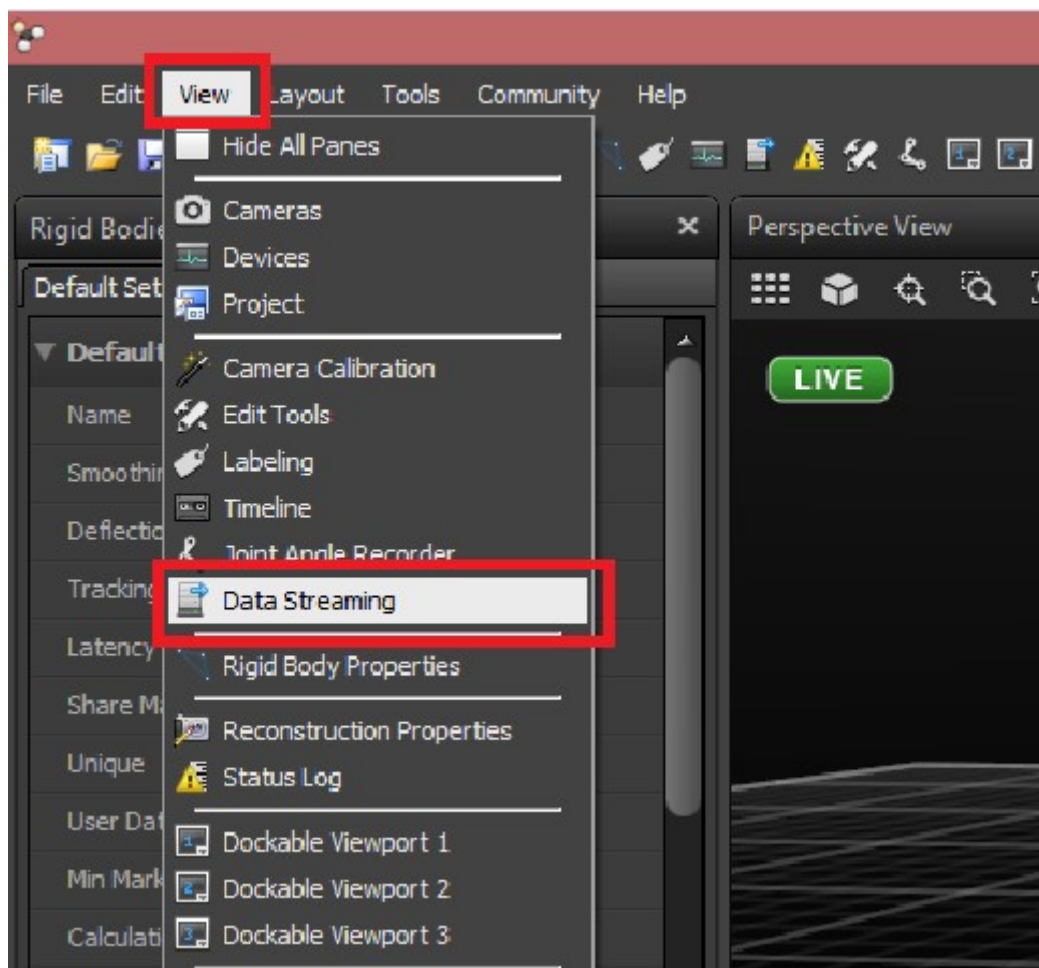
```
ssh root@192.168.31.34
```

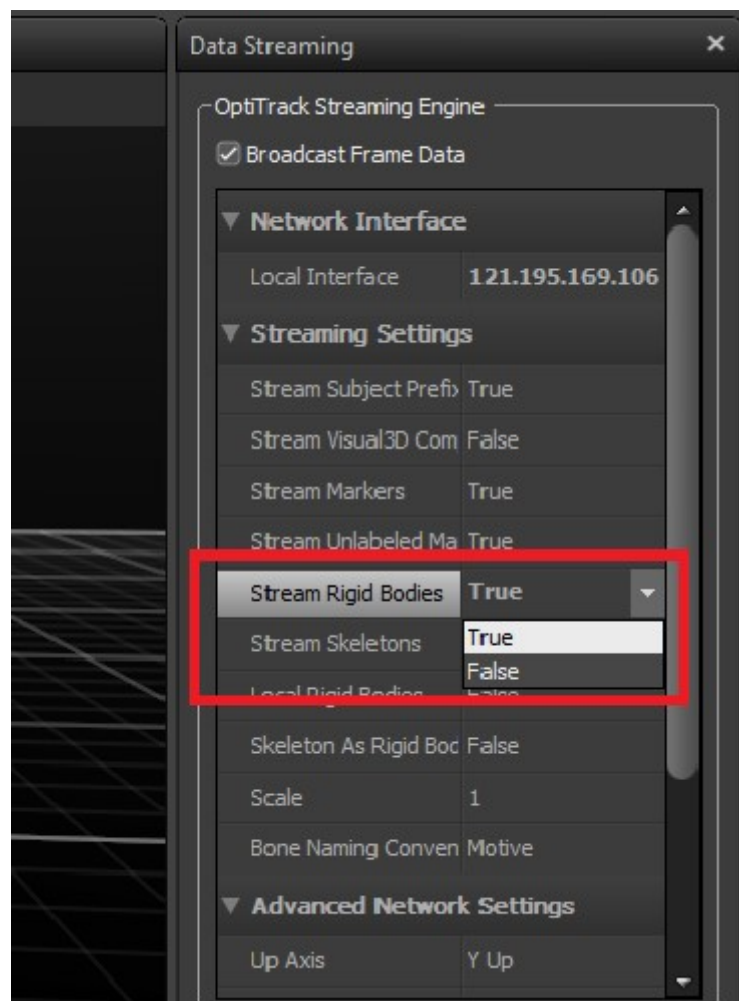
```
ěĹšăĖĖăŖĖčăĬiijĹăčĬmqmaker) :
```

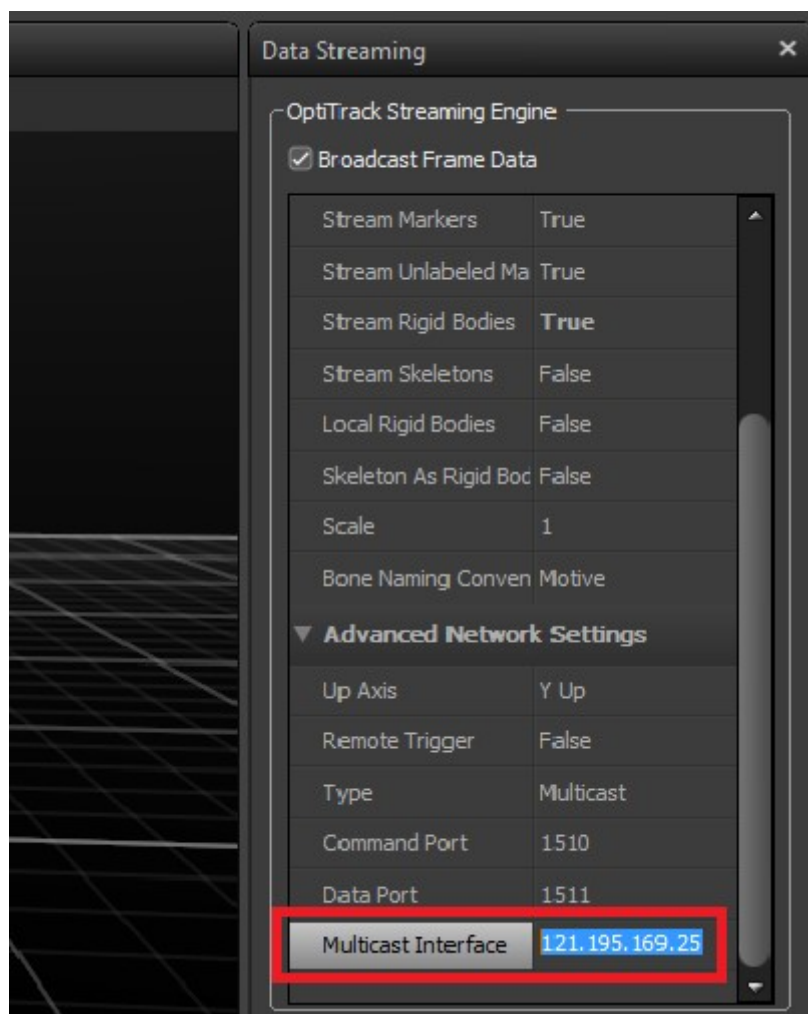
```
mqmaker
```

```
ěĹšăĖĖăžăyŇăŤ;ăžd' iijŽ
```









```
roslaunch mavros px4.launch
```

(2) æLŠaijÄäyÄäyŁæŮřŠ;äzd'èaŇčĽŮřčrijŇèŁŠăĚäzëäyŇăŠ;äzd'èaŇrijŽ

```
ssh root@192.168.31.34
```

```
èŁŠăĚëărĚçăÄïijŁăĚĆmqmaker) :
```

```
mqmaker
```

```
èŁŠăĚëäzëäyŇăŠ;äzd'èaŇrijŽ ``ls`` ``cd /home/`` ``cd /home/catkin_  
→make/``
```

```
source devel/setup.bash      roslaunch mocap optitrack mocap  
launch
```

(3) æŸçd'žăŽĽăĽčĽŮřč rviz rviz

```
(4) æšëçIJŇăŮčæŤrijŁăĚĆLPE_PN_V)      rosrun mavros mavparam get  
LPE_PN_V
```

```
(5) äřőæŤžăŮčæŤrijŁăĚĆLPE_PN_V)      rosrun mavros mavparam set  
LPE_PN_V number
```