

R70GT & R62GT GNSS Receivers

User Manual

Shanghai AllyNav Technology Co.,Ltd.

September 2026

Revision History

Version	Revised by	Revision	Date
V1.0	Hong Junhui	Initial release	

Disclaimer

This document provides information about products of Shanghai AllyNav Technology Co.,Ltd. No license or right to any patent, trademark, copyright or other proprietary right of the Company or any third party is granted by implication, estoppel or otherwise.

Except as stated in its terms and conditions of sale, Shanghai AllyNav Technology Co.,Ltd. assumes no other liability. The Company makes no express or implied warranties regarding the sale and/or use of its products, including fitness for a particular purpose, merchantability or non-infringement of patents, copyrights or other intellectual property rights. The Company is not liable for problems caused by connections or operation that do not comply with this manual. Product specifications and descriptions may change at any time without notice.

Products may contain design defects or errors. Once identified, these will be listed in errata and may cause products to differ from published specifications. The latest errata are available on request.

Before ordering, contact the Company or your local distributor for the latest specifications.

Preface

This User Manual describes the hardware features, performance specifications and operation of ALLYNNAV R70GT and R62GT receivers.

Intended Readers

This User Manual is intended for technical personnel with a basic understanding of Android systems and GNSS receivers.

Manual Structure

This User Manual contains the following chapters:

- 1 Product Overview: Product functions and features
- 2 Technical Features: Receiver technology and capabilities
- 3 Product Appearance: External views and dimensions
- 4 Technical Specifications: Equipment specifications
- 5 Operating Instructions: Basic equipment operation

Contents

1 Product Overview	1
2 Technical Features	1
3 Product Appearance	2
3.1 Product Appearance	2
3.2 Product Dimensions	3
3.3 Product Components	3
3.4 Connector	4
3.4.1 Connector Appearance	4
3.4.2 Connector Pinout	4
3.5 LED Indicators	5
4 Technical Specifications	5
5 Operating Instructions	8
5.1 Introduction	8
5.2 Functional Modules	8
5.2.1 Ethernet	8
5.2.2 CAN	10
5.2.3 Wi-Fi	11
5.2.3.1 STA Mode	12
5.2.3.2 AP Mode	14
5.2.4 RS232	15
5.2.5 Radio	16
5.2.6 GNSS Board	16

5.2.7 Gyroscope	16
5.2.8 SIM Card	16
5.3 System Flashing and Development Setup	20
5.3.1 System Flashing via OTG	20
5.3.1.1 OTG Driver Installation	20
5.3.1.2 System Flashing via OTG	21
5.3.1.2.1 Flashing a Complete Image	22
5.3.1.2.2 Flashing Individual Images	23
5.3.2 Development Environment Setup	24
5.3.2.1 Development Environment	24
5.3.2.2 Development VM Image	24
5.3.2.3 Source Code and Compilation	25
5.3.2.3.1 Full Build	26
5.3.2.3.2 Individual Build	26
5.3.2.3.3 Cleaning Build Files	26
5.3.3 Compiling and Running Applications	26

1 Product Overview

The R70GT GNSS receiver is designed for advanced agricultural vehicles. It integrates high-precision positioning and heading, attitude sensing, wireless communication and edge computing. Built on the high-performance RK3588 platform, it supports Linux/Android and ROS deployment, providing reliable support for precision navigation, travel over complex terrain and intelligent operation control. High-precision positioning and edge computing enable precision navigation and operator-assisted operation in complex environments for advanced agricultural vehicles.

2 Technical Features

- a) A full-constellation, full-frequency BDS/GPS/GLONASS/Galileo/QZSS RTK module maintains positioning accuracy in complex environments. Compatible with multiple high-precision positioning and heading boards; supports BeiDou-only positioning.
- b) An eight-core high-performance CPU and 6 TOPS NPU run Linux and support ROS, meeting edge-computing needs for agricultural robot perception, navigation and control. Wide-range 9 – 36 V DC input with reverse-polarity protection.
- c) Built-in data radio supports TRIMATLK, TRIMMARK3, TRANSEOT, SP, CSS and other protocols.
- d) Wide-range 9 – 36 V DC input with reverse-polarity protection.
- e) Remote diagnostics, remote configuration and multiple screen-mirroring methods simplify debugging and maintenance.
- f) Integrated design for easy installation, with IP67-rated dust and water protection.
- g) A built-in high-precision gyroscope provides high-precision attitude output.

h) L-BAND satellite augmentation enables stable centimeter-level positioning without terrestrial networks or CORS base stations once the service is activated, including operation in areas without network access.

3 Product Appearance

3.1 Product Appearance



Figure 1 R70GT Overview



Figure 2 R62GT Overview

3.2 Product Dimensions

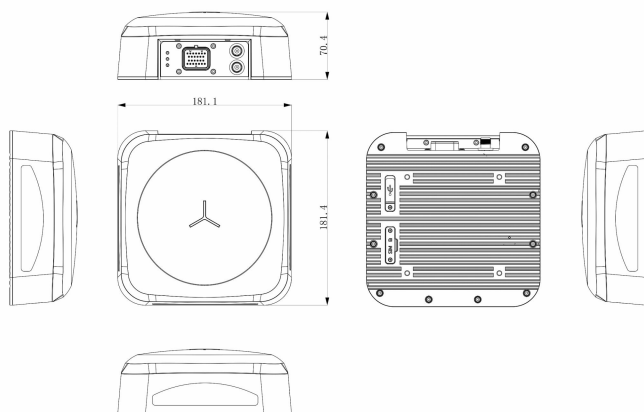


Figure 3 R70GT Dimensions

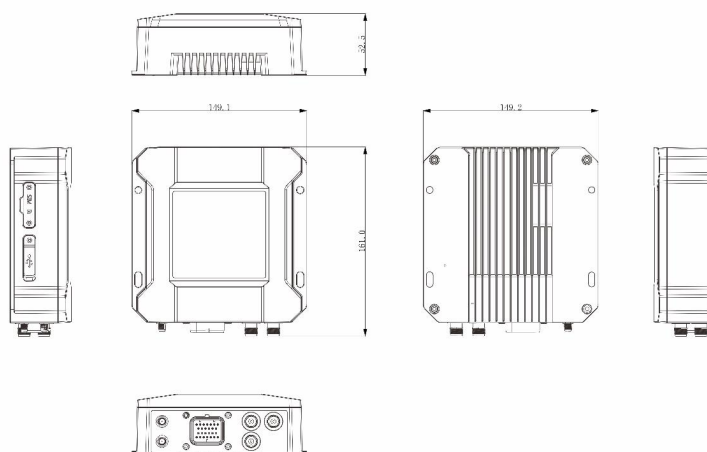


Figure 3 R62GT Dimensions

3.3 Product Components

No.	Component	Color	Qty.	Model	Remarks
1	Main unit	White	1	R70M	Serial number, model, etc.
2	Product label	/	1	/	
3	LED indicators	/	3	/	
4	Mounting holes	/	4	/	For M5 screws

5	SIM card slot	/	1	Standard SIM	
6	USB socket	/	1	USB 2.0	
7	TNC connector	/	1	/	R62GT: 3 pcs
8	9-pin LEMO socket	/	1	/	Ethernet port

3.4 Connector

3.4.1 Connector Appearance

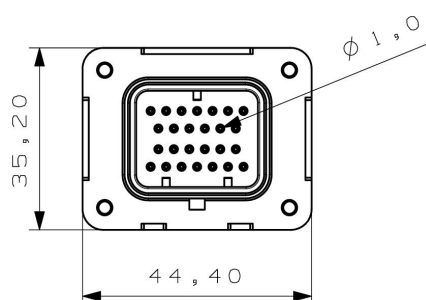


Figure 3 Connector Overview

3.4.2 Connector Pinout

Signal	Pin	Remarks
B+	1	Power positive
B-	2	Power negative
ACC	3	
CAN_1-H	4	
CAN_1-L	5	
12V-OUTPUT	6	
CAMERA-CVBS_3	7	
GND	8	Signal ground
RS485-A	9	
RS485-B	10	
CAN_2-H	11	
CAN_2-L	12	
CAMERA-CVBS_2	13	
RS232_1-TX	14	

/	15
CAMERA-CVBS_1	16
RS232_2-RX	17
/	18
/	19
RS232_1-RX	20
/	21
CAMERA-CVBS_0	22
CAMERA-12V	23
RS232_2-TX	24
RS232_2-RXCAMERA	25
RS232_2-TXCAMERA	26

3.5 LED Indicators

No.	Name	Color	Function	Status
1	POW	Red	Power status	Off: No power On: Power normal
2	SA	Red	RTK status	Off: Not fixed On: Fixed solution
3	LINK	Red	Software status	Software-configurable

4 Technical Specifications

System	OS	Linux (Ubuntu 22.04) or Android
	CPU	RK3588, 4*Cortex-A76 + 4*Cortex-A55
	GPU	ARM Mail-G610 MC4
	NPU	6TOPS
	Memory	8 GB + 64 GB or higher; up to 16 GB + 128 GB
	BT	BT 5.2; backward compatible with 4.2/2.1; supports BLE
	WIFI	2.4 GHz + 5 GHz dual band; 802.11a/b/g/n/ac/ax
satellite Tracking	Constellations	BDS+GPS+GLONASS+Galileo+QZSS+L-Band

	Signals	BDS: B1I, B3I, B2a, B2b GPS: L1 C/A, L2P(Y), L2C, L5 GLONASS: G1, G2 Galileo: E1, E5a, E5b, E6 QZSS: L1 C/A, L1C/B, L2C, L5, L6 L-Band
	Channels	1408
	Initialization reliability	>99.9%
	Satellite corrections	PPP: B2b, E6-HAS, L6MDCPPP PPP-AR: AllyLink Service PPP-RTK: L6CLAS (supported with dedicated firmware versions)
Accuracy	Position accuracy (RMS)	Single point (RMS): Horizontal: 1.5 m; Vertical: 2.5 m RTK (RMS): Horizontal: 0.8 cm + 1 ppm; Vertical: 1.5 cm + 1 ppm PPP (RMS): Horizontal: 5 cm; Vertical: 10 cm PPP-AR (RMS): Horizontal: 3 cm; Vertical: 6 cm CLAS (RMS): Horizontal: 5 cm @ 1 min; Vertical: 10 cm @ 1 min
	Time accuracy (RMS)	20 ns
	Velocity accuracy (RMS)	0.03m/s
	IMU/GNSS fusion accuracy	GNSS/IMU fusion accuracy: Roll and pitch: <0.2° RMS Heading: <0.3° RMS

		(Vehicle speed >1 km/h; vehicle-mounted application with Ackermann steering)
User Interface	LED indicators	1 POW LED, 1 SA LED and 1 LINK LED
	Screen mirroring	Via USB, network, etc.
Electrical	Power input	9-36V DC
	Output voltage	12V
	Power consumption	≤6.0W
Physical	Dimensions	161 mm × 149.1 mm × 52.5 mm
Physical	Weight	<1.1 kg
	Material	Metal
	Operating temperature	-20°C~+70°C
	Storage temperature	-40°C~+85°C
	Ingress protection	IP67
Data Output	Data formats	NMEA-0183, RTCM3.X
	Update rate	RTK+: 20 Hz; IMU: 100 Hz
	Baud rate	9600–460800
	CAN bit rate	250k, 500k
Communication	Cellular network	Global coverage LTE-FDD: B1/2/3/4/5/7/8/12/13/14/18/19/20/25/26/28/66/71 LTE-TDD: B38/39/40/41 WCDMA: B1/2/4/5/6/8/19
		LTE-FDD: B1/2/3/4/5/7/8/12/13/14/18/19/20 /25/26/28/66/71 LTE-TDD: B38/39/40/41 WCDMA: B1/2/4/5/6/8/19
	Radio	Radio: Frequency: 410–470 MHz

		Range: <5 km (open area / vehicle-mounted radio antenna) Supports TRIMATLK, TRIMMARK3, TRANSEOT, SP, CSS and other protocols
	AMP26Pin	4 × AHD, 3 × RS232, 2 × CAN, 1 × RS485
	Interfaces	TNC, USB 3.0, SIM card slot
	Ethernet	Up to 1000 Mbps

5 Operating Instructions

5.1 Introduction

R62GT and R70GT operating system: Ubuntu 22.04.

The development board provides a debug serial port and a system flashing port. The debug port uses USB Type-C; the flashing port uses USB-A.

5.2 Functional Modules

5.2.1 Ethernet

Ethernet 0 (eth0) supports 100 Mbps and 1000 Mbps.

R62GT and R70GT use a static IP address by default. The configuration file is /etc/network/interfaces.d/eth0.

- For dynamic IP addressing, use:

```
auto eth0
```

```
iface eth0 inet dhcp
```

- For static IP addressing, the following example sets eth0 to 192.168.0.232:

```
auto eth0
```

```
iface eth0 inet static
```

```
address 192.168.0.232
```

```
netmask 255.255.255.0
```

```
gateway 192.168.0.1
```

Parameters:

Parameter	Description
iface	Interface for static IP
address	Static IP address
netmask	Subnet mask
gateway	Gateway

- After editing, run sync to synchronize the file system. Restart the board or networking service to apply the configuration.

```
ip addr flush dev eth0 // Clear the interface IP address
```

```
systemctl restart networking.service // Restart networking
```

5.2.2 CAN

R62GT and R70GT have two CAN bus interfaces. Connect the CAN H terminal to the H terminal of the other CAN device, and CAN L to the other device's L terminal.

- List CAN network devices:

```
ifconfig -a
```

- Set the CAN bit rate:

```
ip link set can0 type can bitrate 500000 triple-sampling on
```

```
ip link set can1 type can bitrate 500000 triple-sampling on
```

- Enable the CAN devices:

```
ifconfig can0 up
```

```
ifconfig can1 up
```

- Send data from the client and receive it at the server:

Use can0 as the server (run this command on the server first):

```
candump can0&
```

Use can1 as the client (send data):

```
cansend can1 1F334455#112233445566778
```

Note: The IMU614-AG in R62GT and R70GT shares the CAN bus with RK3588 can0. After power-on, the IMU614-AG continuously sends heading-related CAN data at 500 kbps.

5.2.3 Wi-Fi

Check that the Wi-Fi chip has loaded successfully:

```
root@ok3588:~# lspci
0004:40:00.0 PCI bridge: Fuzhou Rockchip Electronics Co., Ltd Device 3588 (rev 01)
0004:41:00.0 Network controller: Realtek Semiconductor Co., Ltd. Device b852
```

```
root@ok3588:~# ifconfig
docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
    ether 02:42:af:15:37:38 txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 192.168.0.232 netmask 255.255.255.0 broadcast 192.168.0.255
    ether 9e:24:2d:df:cc:70 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 77

eth1: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 96:24:83:f2:ca:4e txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 79

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 226 bytes 17830 (17.8 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 226 bytes 17830 (17.8 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

p2p0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 9a:03:cf:d9:d6:4b txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlan0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 98:03:cf:d9:d6:4b txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Note: p2p0 and wlan0 are interfaces mapped from the same hardware. Both can operate as an access point or station. Normally, p2p0 is used for Wi-Fi Direct, while wlan0 is used for standard STA or AP mode.

5.2.3.1 STA Mode

In STA mode, the device acts as a station connected to a wireless network.

In this test, the router uses WPA encryption. The Wi-Fi hotspot is named Future and its password is 12345678. Adjust these settings to match your network environment.

- Enter this command in the board terminal to connect to the hotspot:

```
fltest_wifi.sh -s Future -p 12345678
```

```

root@ok3588:~# fltest_wifi.sh -h
/usr/bin/fltest_wifi.sh: illegal option -- h
eg: /usr/bin/fltest_wifi.sh -s SSID -p password
root@ok3588:~# fltest_wifi.sh -s Future -p 12345678
Device 'p2p0' successfully activated with 'e3c3ad8b-d104-4dc8-9e18-162624785aea'.
root@ok3588:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.137.231 netmask 255.255.255.0 broadcast 192.168.137.255
    inet6 fe80::69a7:4fd9:fe95:2ab6 prefixlen 64 scopeid 0x20<link>
    ether 16:d7:f9:2a:88:44 txqueuelen 1000 (Ethernet)
    RX packets 161 bytes 26092 (26.0 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 131 bytes 20643 (20.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 77

eth1: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether d2:09:43:bd:d5:78 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 79

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 375 bytes 34314 (34.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 375 bytes 34314 (34.3 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

p2p0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.43.101 netmask 255.255.255.0 broadcast 192.168.43.255
    inet6 2408:8471:5f0a:564:649d:2ca5:5c0d:ff62 prefixlen 64 scopeid 0x0<global>
    inet6 2408:8471:5f0a:564:4c06:163b:bc3:6237 prefixlen 64 scopeid 0x0<global>
    inet6 fe80::9030:73c7:5781:9066 prefixlen 64 scopeid 0x20<link>
    ether 16:5d:34:ca:65:66 txqueuelen 1000 (Ethernet)
    RX packets 38 bytes 7346 (7.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 90 bytes 13628 (13.6 KB)
    TX errors 0 dropped 64 overruns 0 carrier 0 collisions 0
    device interrupt 143

wlan0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 14:5d:34:ca:65:66 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 143

wwan0: flags=193<UP,RUNNING,NOARP> mtu 1500
    inet 10.75.24.159 netmask 255.255.255.192
    inet6 fe80::28ca:aff:fea2:4b42 prefixlen 64 scopeid 0x20<link>
    ether 2a:ca:0a:a2:4b:42 txqueuelen 1000 (Ethernet)

```

Command parameters:

Parameter	Description
-s	Name of the Wi-Fi hotspot to connect to
-p	Password for the Wi-Fi hotspot

- To check external network connectivity, run a ping command in the terminal:

For example, ping Baidu: ping -I p2p0 www.baidu.com -c 10

```
root@ok3588:~# ping -I p2p0 www.baidu.com -c 10
PING www.baidu.com (2409:8c54:870:310:0:ff:b0ed:40ac (2409:8c54:870:310:0:ff:b0ed:40ac)) from 2408:8471:5f0a:564:649d:2ca5:5c0d:ff62 p2p0: 56 data bytes
64 bytes from 2409:8c54:870:310:0:ff:b0ed:40ac (2409:8c54:870:310:0:ff:b0ed:40ac): icmp_seq=1 ttl=49 time=79.2 ms
64 bytes from 2409:8c54:870:310:0:ff:b0ed:40ac (2409:8c54:870:310:0:ff:b0ed:40ac): icmp_seq=2 ttl=49 time=81.3 ms
64 bytes from 2409:8c54:870:310:0:ff:b0ed:40ac (2409:8c54:870:310:0:ff:b0ed:40ac): icmp_seq=3 ttl=49 time=78.3 ms
64 bytes from 2409:8c54:870:310:0:ff:b0ed:40ac (2409:8c54:870:310:0:ff:b0ed:40ac): icmp_seq=4 ttl=49 time=72.4 ms
64 bytes from 2409:8c54:870:310:0:ff:b0ed:40ac (2409:8c54:870:310:0:ff:b0ed:40ac): icmp_seq=5 ttl=49 time=82.2 ms
64 bytes from 2409:8c54:870:310:0:ff:b0ed:40ac (2409:8c54:870:310:0:ff:b0ed:40ac): icmp_seq=6 ttl=49 time=87.9 ms
64 bytes from 2409:8c54:870:310:0:ff:b0ed:40ac (2409:8c54:870:310:0:ff:b0ed:40ac): icmp_seq=7 ttl=49 time=71.7 ms
64 bytes from 2409:8c54:870:310:0:ff:b0ed:40ac (2409:8c54:870:310:0:ff:b0ed:40ac): icmp_seq=8 ttl=49 time=73.4 ms
64 bytes from 2409:8c54:870:310:0:ff:b0ed:40ac (2409:8c54:870:310:0:ff:b0ed:40ac): icmp_seq=9 ttl=49 time=91.0 ms
64 bytes from 2409:8c54:870:310:0:ff:b0ed:40ac (2409:8c54:870:310:0:ff:b0ed:40ac): icmp_seq=10 ttl=49 time=69.1 ms

--- www.baidu.com ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9013ms
rtt min/avg/max/mdev = 67.918/76.641/90.976/6.723 ms
root@ok3588:~#
```

Note: fltest_wifi.sh selects an available interface from p2p0 or wlan. This example uses p2p0. To specify an interface, use:

nmcli dev wifi connect SSID password PASSWORD ifname CARD

Parameter	Description
SSID	Name of the Wi-Fi hotspot to connect to
PASSWORD	Password for the Wi-Fi hotspot
CARD	Interface to use: p2p0 or wlan0

5.2.3.2 AP Mode

In AP mode, the device acts as a Wi-Fi hotspot.

- Configure the hotspot:

fltest_hostapd.sh

```
root@ok3588:~# fltest_hostapd.sh
net.ipv4.ip_forward = 1
Device 'p2p0' successfully activated with 'dea42717-6cf7-4ad3-b0c9-0c24a7ff29a9'.
Hint: "nmcli dev wifi show-password" shows the Wi-Fi name and password.
root@ok3588:~#
```

Important:

- 1、 fltest_hostapd.sh uses nmcli to configure the hotspot. Edit the script at /usr/bin/fltest_hostapd.sh to change the SSID, password or network interface.
- 2、 The script also includes NAT forwarding rules so devices connected to the AP can access the Internet. Edit the script to modify these rules.
- 3、 For AP configuration, create your own script or integrate the configuration into your code to simplify future upgrades.

5.2.4 RS232

R62GT and R70GT provide two external RS232 interfaces.

RS232	Device node
RS232-1	/dev/ttyS5
RS232-2	/dev/ttyS3

Quick test:

Open a serial terminal on the PC. Set 115200 baud, 8 data bits, 1 stop bit, no parity and no flow control, then open the serial port.

Enter this command in the board's serial console (the test program uses a fixed baud rate of 115200): fltest_uarttest -d /dev/ttyS5

Example output:

```
root@ok3588:~# fltest_uarttest -d /dev/ttyS5
Welcome to uart test
Send test data:
forlinux_uart_test.1234567890...
```

5.2.5 Radio

R62GT and R70GT connect to the radio via RS232. The radio's device node is /dev/ttyS8.

5.2.6 GNSS Board

R62GT and R70GT connect to the GNSS board via RS232. Its device node is /dev/ttyS7.

5.2.7 Gyroscope

R62GT and R70GT connect to the gyroscope via RS232. Its device node is /dev/ttyS0.

5.2.8 SIM Card

- Connect the module and power on the board and module. Run lsusb to check USB status.

```
root@ok3588:~# lsusb
Bus 006 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 005 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 004 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 002 Device 002: ID 2c7c:030e Quectel Wireless Solutions Co., Ltd. Quectel EM05-G
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 003 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

- Check the device nodes under /dev:

```
root@ok3588:~# ls /dev/ttyUSB*
/dev/ttyUSB0 /dev/ttyUSB1 /dev/ttyUSB2 /dev/ttyUSB3
```

● Switching SIM Cards

R62GT and R70GT support dual SIM, single standby: one onboard SIM and one external user-supplied SIM.

Query the active SIM through /dev/ttyUSB2:

Query command: at+qdsim?

Response meanings:

+QDSIM: 0	External user-supplied SIM is active
+QDSIM: 1	Onboard eSIM is active (for use in China only)

```
root@ok3588:~#
root@ok3588:~#
root@ok3588:~#
root@ok3588:~#
root@ok3588:~# busybox microcom /dev/ttyUSB2
at+qdsim?
+QDSIM: 0
OK
```

Note: Subsequent production releases use the external user-supplied SIM by default.

● Cellular Data Connection

Run fltest_quectel.sh to establish a cellular data connection.

The script is located at /usr/bin/fltest_quectel.sh and calls quectelCM.

```

root@ok3588:~# fltest_quectel.sh &
[1] 1952
root@ok3588:~# eth0: flags=4098<BROADCAST,MULTICAST> mtu 1500
eth1: flags=4098<BROADCAST,MULTICAST> mtu 1500
[04-02_01:41:39:705] Quectel_QConnectManager_Linux_V1.6.0.24
[04-02_01:41:39:707] Find /sys/bus/usb/devices/2-1 idVendor=0x2c7c idProduct=0x30e, bus=0x002, dev=0x002
[04-02_01:41:39:716] Auto find qmichannel = /dev/cdc-wdm0
[04-02_01:41:39:720] Auto find usbnet_adapter = wwan0
[04-02_01:41:39:725] netcard driver = qmi_wwan_q, driver version = V1.2.9
[04-02_01:41:39:732] Modem works in QMI mode
[04-02_01:41:39:761] cdc_wdm_fd = 7
[04-02_01:41:39:838] Get clientWDS = 6
[04-02_01:41:39:870] Get clientDMS = 1
[04-02_01:41:39:902] Get clientNAS = 3
[04-02_01:41:39:934] Get clientUIM = 3
[04-02_01:41:39:965] Get clientWDA = 1
[04-02_01:41:39:998] requestBaseBandVersion EM05GFAR07A07M1G
[04-02_01:41:40:125] requestGetSIMStatus SIMStatus: SIM_READY
[04-02_01:41:40:157] requestGetProfile[1] cmnet///0
[04-02_01:41:40:190] requestRegistrationState2 MCC: 460, MNC: 0, PS: Attached, DataCap: LTE
[04-02_01:41:40:221] requestQueryDataCall IPv4ConnectionStatus: DISCONNECTED
[04-02_01:41:40:223] ifconfig wwan0 0.0.0.0
[04-02_01:41:40:243] ifconfig wwan0 down
[04-02_01:41:40:318] requestSetupDataCall WdsConnectionIPv4Handle: 0x86ca4fc0
[04-02_01:41:40:447] ifconfig wwan0 up
[04-02_01:41:40:464] No default.script found, it should be in '/usr/share/udhcpc/' or '/etc//udhcpc' depend on your udhcpc version!
[04-02_01:41:40:470] busybox udhcpc -f -n -q -t 5 -i wwan0
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 10.30.14.176
udhcpc: lease of 10.30.14.176 obtained, lease time 7200
[04-02_01:41:40:617] ip -4 address flush dev wwan0
[04-02_01:41:40:632] ip -4 address add 10.30.14.176/27 dev wwan0
[04-02_01:41:40:644] ip -4 route add default via 10.30.14.177 dev wwan0

```

- Check the configuration before testing:

Check the gateway configuration:

```

root@ok3588:~# route -n
Kernel IP routing table
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface
0.0.0.0          10.30.14.177    0.0.0.0          UG    0      0      0 wwan0
10.30.14.160     0.0.0.0         255.255.255.224  U    0      0      0 wwan0
172.17.0.0       0.0.0.0         255.255.0.0      U    0      0      0 docker0
root@ok3588:~#

```

Check the DNS configuration:

```

root@ok3588:~# cat /etc/resolv.conf
nameserver 120.196.165.7 # IPV4 wwan0
nameserver 221.179.38.7 # IPV4 wwan0
nameserver 8.8.8.8
nameserver 114.114.114.114
nameserver 127.0.0.53

```

- After configuring DNS and routing, ping a domain name:

ping www.google.com

Note:

1、By default, fltest_quectel.sh does not configure the APN, username, password or authentication method. For SIM cards whose carriers require these settings, use the command below. Terminate fltest_quectel.sh first.

quectelCM -s APN USER PASSWD AUTH

Parameter	Description
APN	Access point name required by the SIM carrier
USER	Access username; enter "" if none
PASSWORD	Access password; enter "" if none
AUTH	Authentication: 1-PAP; 2-CHAP

2、Alternative APN Configuration

(1) Terminate fltest_quectel.sh.

(2) Send these commands through /dev/ttyUSB2 in order:

Order	Command	Description
1	AT+QICSGP=1,3,APN,username,password,auth	APN: Access point name. username: User name; "" if none. password: Password; "" if none.

		auth: 0=None;1-PAP;2-CHAP;3-PAP_CHAP
2	at+cfun=0	Set minimum functionality (approximately powers off the modem)
3	at+cfun=1	Set full functionality (approximately powers on the modem)

(3) Release /dev/ttyUSB2, then run fltest_quectel.sh again.

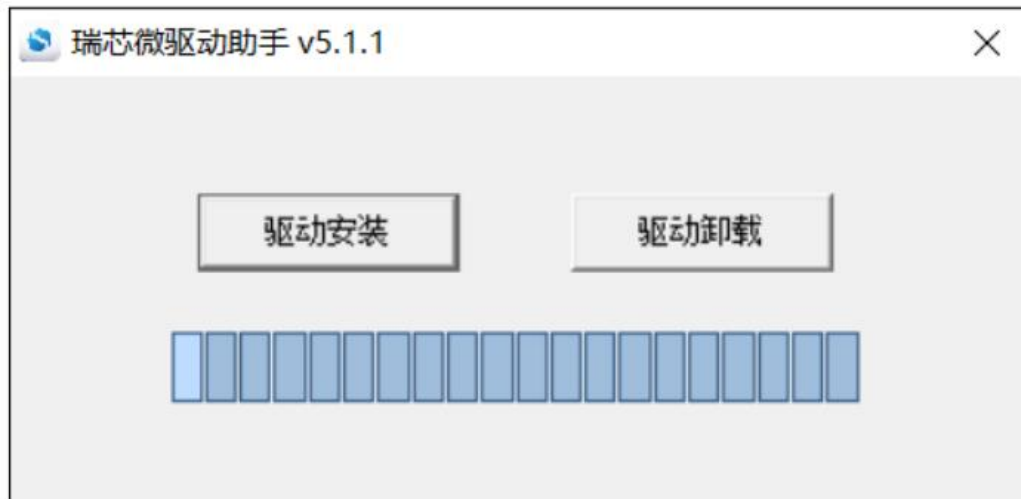
5.3 System Flashing and Development Setup

5.3.1 System Flashing via OTG

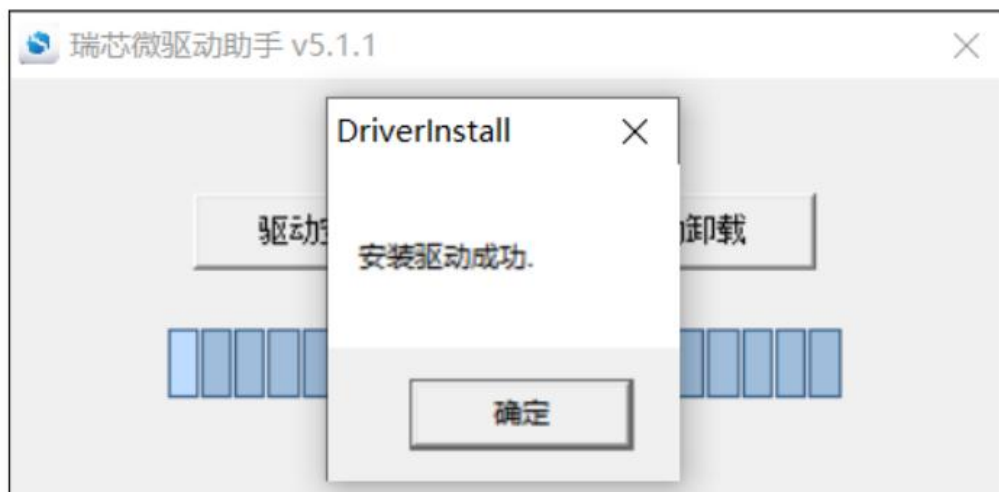
5.3.1.1 OTG Driver Installation

Driver package: DriverAssitant_v5.11.zip

Extract the package to any directory. Run DriverInstall.exe as administrator.



Click Driver Install.



5.3.1.2 System Flashing via OTG

Flashing tool: RKDevTool_Release_v2.84.zip

Extract the tool to a path containing only English characters. Connect a USB-A cable to the board's flashing port.

Enter flashing mode:

reboot loader

```
root@ok3588:~# reboot loader
Stopping Session c1 of user forlinux.
Stopping Session c2 of user root.
[ OK ] Removed slice system-getty.slice.
[ OK ] Removed slice system-modprobe.slice.
[ OK ] Stopped target Graphical Interface.
[ OK ] Stopped target Multi-User System.
[ OK ] Stopped target Login Prompts.
[ OK ] Stopped target Host and Network Name Lookups.
[ OK ] Stopped target Sound Card.
[ OK ] Stopped target Timers.
[ OK ] Stopped Daily apt upgrade and clean activities.
[ OK ] Stopped Daily apt download activities.
[ OK ] Stopped Periodic ext4 Online Metadata Check for All Filesystems.
```

5.3.1.2.1 Flashing a Complete Image

After entering flashing mode, the Rockchip development tool reports that a loader device has been found.

Open the Rockchip development tool:



Select the Upgrade Firmware tab. Click Firmware and select the complete update.img image.

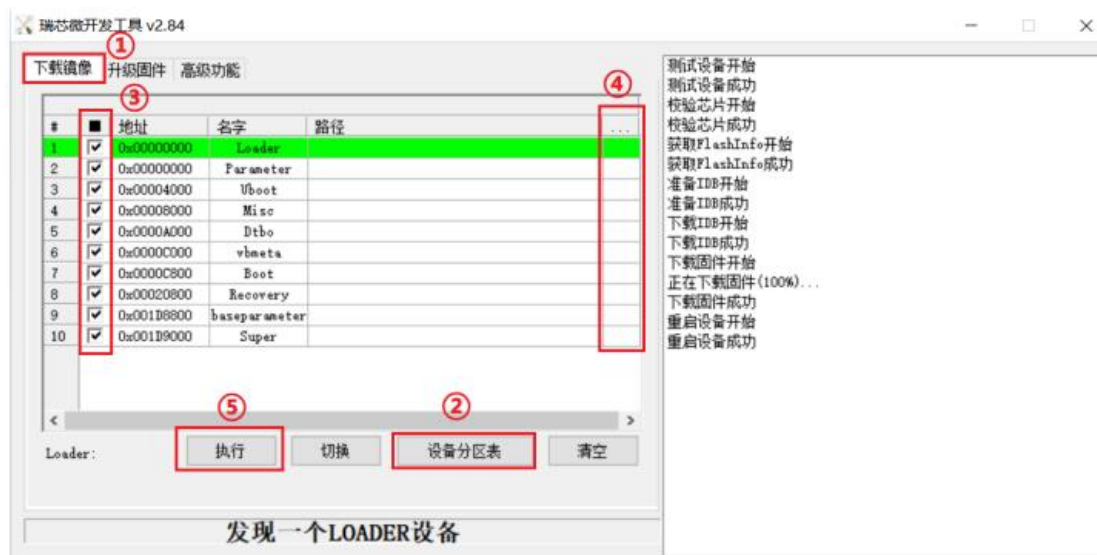
Wait while the tool parses the firmware.



Click Upgrade to start the update.

5.3.1.2.2 Flashing Individual Images

Use this function when individual images need to be downloaded.



1. Click ① Download Image.
2. Click ② Device Partition Table to read partition locations.

3. Select ③ the checkboxes for the images to flash.
4. Click ④ to select an image.
5. Click ⑤ Run to start flashing.
6. Restart the board after flashing is complete.

5.3.2 Development Environment Setup

5.3.2.1 Development Environment

The development environment runs in a VMware Player virtual machine:

VMware® Workstation 17 Player

Version:

17.5.2 build-23775571

5.3.2.2 Development VM Image

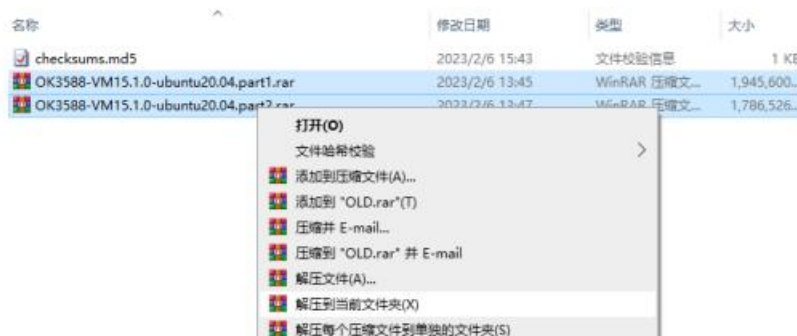
The supplied virtual machine already includes the cross-compiler.

Default user: forlinx; password: forlinx

Download the development environment supplied by Forlinx. An MD5 checksum file is included. Verify the downloaded archive using an online MD5 tool or a locally installed utility. If the checksum matches the supplied file, the download is intact. If it does not match, the archive may be corrupted and should be downloaded again.

名称	修改日期	类型	大小
checksums.md5	2023/2/6 15:43	文件校验信息	1 KB
OK3588-VM15.1.0-ubuntu20.04.part1.rar	2023/2/6 13:45	WinRAR 压缩文...	1,945,600...
OK3588-VM15.1.0-ubuntu20.04.part2.rar	2023/2/6 13:47	WinRAR 压缩文...	1,786,526...

Select all archive parts. Right-click and extract them to the current folder or a folder of your choice:



After extraction, open the OK3588-VM15.1.0-ubuntu20.04 folder. Load its virtual-machine file, 3588 开发环境.vmx, in the installed VMware application. Click Power on this virtual machine to start the system.

5.3.2.3 Source Code and Compilation

Development OS: Ubuntu 20.04, 64-bit

Cross-toolchain: aarch64-linux-gnu

Board bootloader: u-boot-2017.09

Board kernel: linux-5.10.66

5.3.2.3.1 Full Build

1. Extract the source: `tar -xvf OK3588_Linux_fs.tar.bz2`
2. Run a full build: `sudo ./build.sh`

After a successful build, system images are generated in the rockdev folder.

The packaged update.img is used for complete OTG flashing; the other images are used for individual flashing.

5.3.2.3.2 Individual Build

Command: `sudo ./build.sh kernel`

This build does not update the kernel inside update.img. Flash kernel/boot.img separately.

5.3.2.3.3 Cleaning Build Files

Command: `sudo ./build.sh cleanal`

5.3.3 Compiling and Running Applications

This example uses the watchdog test program. The source is copied to /home/forlinux/3588 by default.

- 1、 Use cd to enter the test source directory:

```
cd app/forlinx/forlinx_cmd/fltest_watchdog
```

2、 Add the cross-compiler path and run make to cross-compile:

```
forlinx@ubuntu: ~/3588/OK3588_Linux_fs/app/forlinx/forlinx_cmd/fltest_watchdog$ export  
PATH=/home/forlinx/aarch64-buildroot-linux-gnu_sdk-buildroot/bin/:$PATH  
forlinx@ubuntu: ~/3588/OK3588_Linux_fs/app/forlinx/forlinx_cmd/fltest_watchdog$ make  
aarch64-linux-gcc watchdog.c -o fltest_watchdog  
generate fltest_watchdog success!!!
```

3、 Copy the compiled fltest_watchdog to the board, for example to /forlinx, using a TF card or ADB. The following example uses a TF card to copy the program and run the test.

```
root@ok3588:/# cp /run/media/sda1/fltest_watchdog /home/forlinx/  
root@ok3588:/# cd /home/forlinx/  
root@ok3588:/home/forlinx# ./fltest_watchdog  
Watchdog Ticking Away!
```

FCC STATEMENT:

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions:

(1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

Any Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

Note: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

This equipment complies with FCC radiation exposure limits set forth for an uncontrolled environment. This equipment should be installed and operated with minimum distance 30cm between the radiator & your body