

Quadruped-Roboter Sekundäre Entwicklungs-Version

Entwicklerhandbuch V1.0

Unverbindliche Übersetzung von NORWA Robotics. Bei Abweichungen ist die englische Dokumentation des Herstellers maßgeblich.

1. Beschreibung der Sekundären Entwicklungs-Version

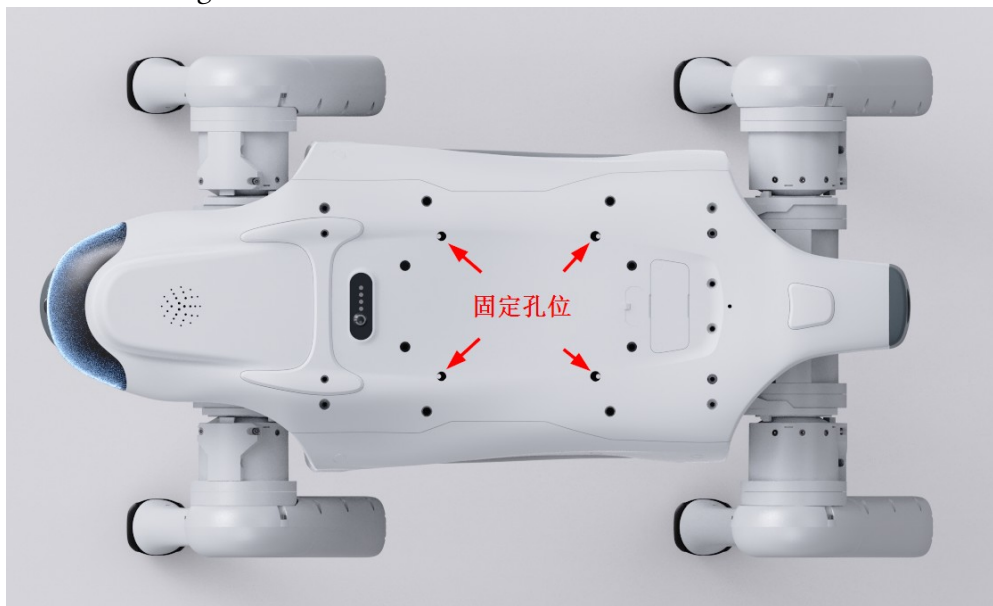
(1) Gerätebefestigung

Die Rückseite des Quadruped-Roboters ist mit Befestigungslöchern, Erweiterungs- und Stromanschlüssen ausgestattet.

Die Befestigungslöcher werden verwendet, um die Entwicklergeräte zu fixieren.

Die geschraubten Löcher können nach Entfernen der Silicone-Stopper an den in der nachfolgenden Abbildung gezeigten Positionen gesehen werden.

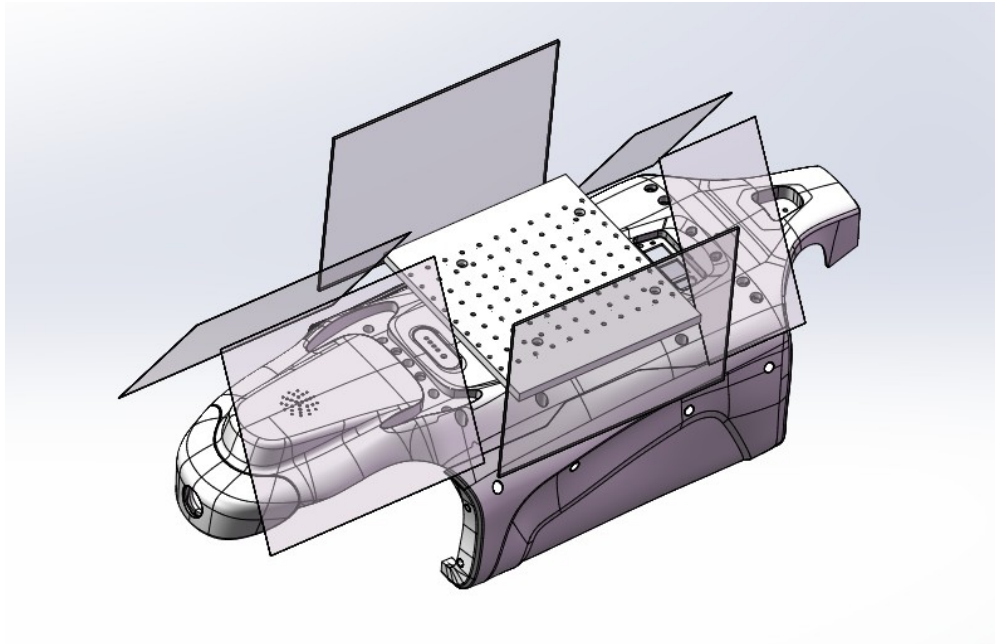
Die Schraubengröße hier ist M4



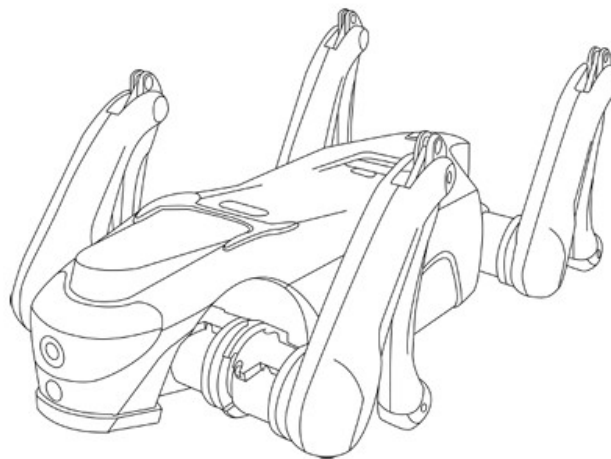
Bitte wenden Sie sich an den Vertriebsmitarbeiter, um die 3D-Zeichnung zu erhalten, wenn Sie das oberflächige Ausrüstung gestalten.

Die Zeichnung beinhaltet das in der nachfolgenden Abbildung gezeigte Referenzdesign, und das Entwicklergerät kann auf der in der Abbildung gezeigten Flachplatte befestigt werden. Die geschraubten Löcher auf der Flachplatte sind M4.

Da die Beine des Roboterhundes während des Homing, des Aufstehens und des Liegen in einem festen Trajektorium schwingen, empfehlen wir, das Gerät innerhalb des in der Abbildung gezeigten Bereichs zu installieren.



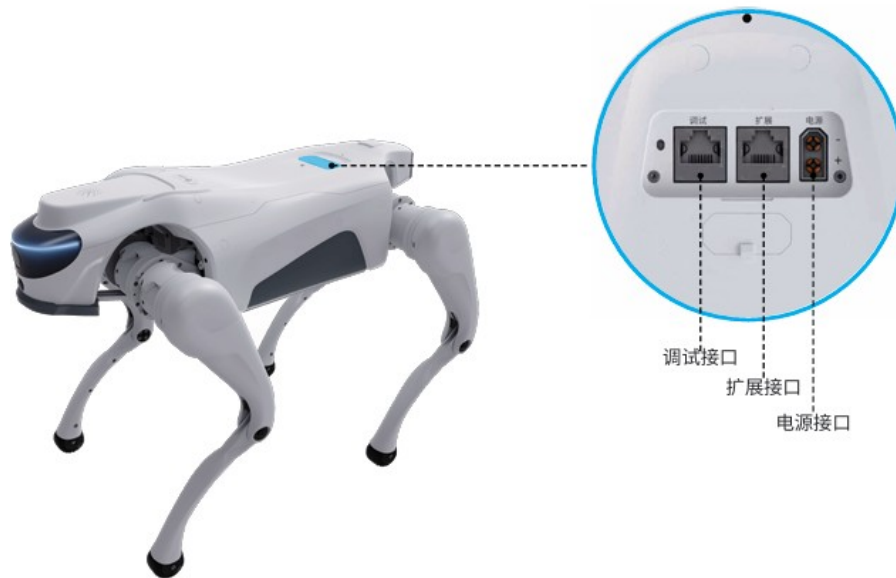
Hinweis: Stellen Sie die vier Beine in die im folgenden Bild gezeigte Haltung vor dem Einschalten.



Hinweis: Blockieren Sie weder die Einstellknopf noch den Ladeschluss beim Gestalten der oberflächigen Ausrüstung. Der Roboterhund kann ein Gewicht von maximal 8 kg tragen, und das Lastgewicht sollte so zentriert auf dem Körper wie möglich platziert werden.

(2) Stromversorgungs- und Kommunikations-Schnittstellen

Entfernen Sie die Rückseite mit einem Sechsspindel-Schraubenzieher, um die Erweiterungsschnittstelle und die Stromversorgungsschnittstelle zu sehen, wie im folgenden Bild gezeigt.

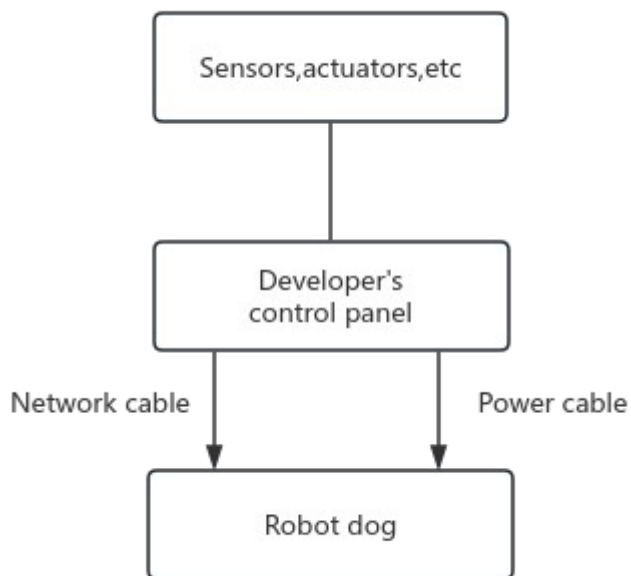


Erweiterungsschnittstelle: Es wird ein Gigabit-Ethernet-Port verwendet.
Stromversorgungsschnittstelle: Es wird ein XT60-Stecker verwendet. Die Energiequelle stammt aus dem Batteriemodul mit einem Spannungsbereich von 22,4V ~ 33,6V.

Hinweis: Reversieren Sie nicht die Polster (+ für positiv, - für negativ).
 Stellen Sie sicher, dass der Stromstecker beim Ausgeschalteten Gerät eingesteckt und ausgeschoben wird. Die Batterie kann einen kontinuierlichen Strom von 40A und einen Spitzenstrom von 60A liefern, wovon ein Teil für die Operation des Roboterhundes verwendet wird. Wenden Sie sich bei der Gestaltung an den Vertriebsmitarbeiter für technische Unterstützung.

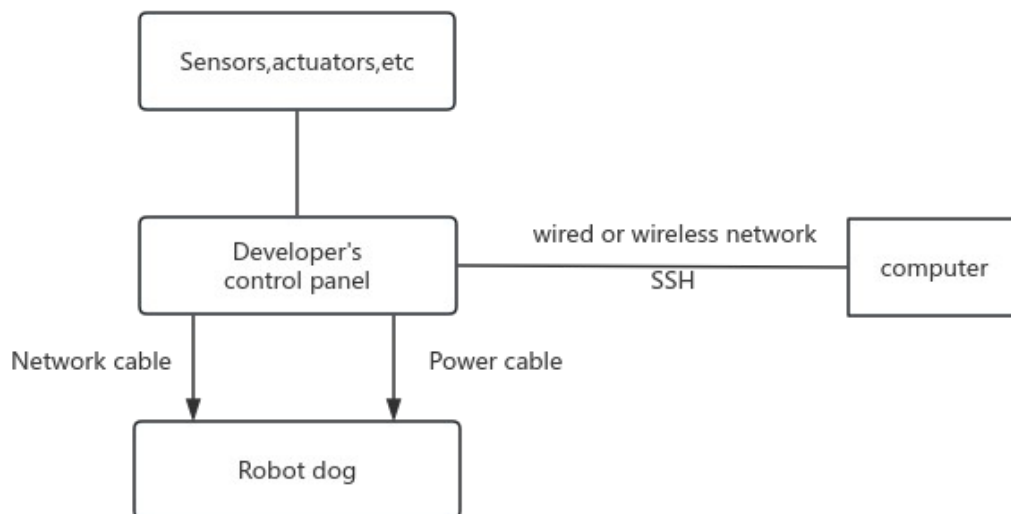
2. Hardwareverbindung

Die Hardwareverbindungsbeziehung ist im folgenden Bild dargestellt.



3. Entwicklungs-Umgebungsrahmen

Die empfohlene Struktur der Entwicklungsumgebung ist in der folgenden Abbildung dargestellt.

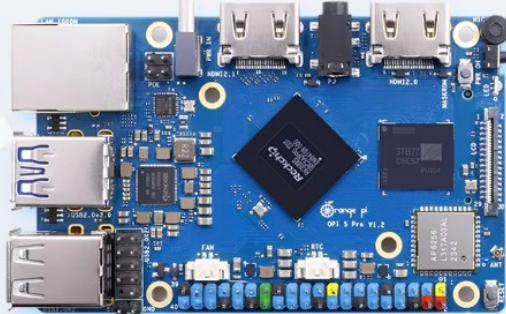


(1) Laboratoriumstestumgebung

Steuerungstafel: Orange Pi 5 Pro Entwicklungsboard

Betriebssystem: Ubuntu 20.04

 Orange Pi
Orange Pi 5 Pro 开发板
Wi-Fi 5+BT5.0 | 千兆以太网



主控芯片 八核 64位CPU 主频高达2.4GHz
RK3588S **4GB/8GB/16GB可选**

(2) SDK

Wir bieten ein SDK auf C++ basierend. Bitte wenden Sie sich an den Vertriebspersonal, um es zu erhalten.

Das SDK läuft auf der Steuerungstafel.

Entwickler können sich über SSH in die Steuerungstafel einloggen und die Funktionen des SDK über die Tastatur testen.

4. Testen des SDK

(1) Konfiguration der Steuerungstafel

Installieren Sie das Ubuntu-System

Das SDK hat keine starke Abhängigkeit von der Ubuntu-Version, und Entwickler können es selbst anpassen.

Das System, das wir getestet haben, ist Ubuntu 20.04.

```
orangepi@orangepi5pro:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 20.04 LTS
Release:        20.04
Codename:       focal
```

Verbinden Sie das Steuerboard mit dem WIFI.

Verbinden Sie den Anzeigebildschirm mit dem HDMI-Schnittstellen des Steuerboards, verbinden Sie Tastatur und Maus mit dem USB-Schnittstellen des Steuerboards und schalten Sie das Steuerboard ein.

Nachdem Sie sich in das Ubuntu-System eingeloggt haben, verwenden Sie die grafische Benutzeroberfläche, um zuerst den Benutzernamen und das Passwort einzugeben, um sich mit dem WIFI zu verbinden.

Installieren Sie das jsoncpp-Abhängigkeitspaket:

Öffnen Sie das Terminal und geben Sie den Befehl ein:

```
sudo apt install libjsoncpp-dev
```

```
orangeipi@orangeipi5pro:~$ sudo apt install libjsoncpp-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  libjsoncpp-dev
0 upgraded, 1 newly installed, 0 to remove and 294 not upgraded.
Need to get 85.5 kB of archives.
After this operation, 496 kB of additional disk space will be used.
Get:1 http://mirrors.tuna.tsinghua.edu.cn/ubuntu-ports focal/main arm64 libjsoncpp-dev arm64 1.7.4-3.1ubuntu2 [85.5 kB]
Fetched 85.5 kB in 0s (184 kB/s)
Selecting previously unselected package libjsoncpp-dev:arm64.
(Reading database ... 221692 files and directories currently installed.)
Preparing to unpack .../libjsoncpp-dev_1.7.4-3.1ubuntu2_arm64.deb ...
Unpacking libjsoncpp-dev:arm64 (1.7.4-3.1ubuntu2) ...
Setting up libjsoncpp-dev:arm64 (1.7.4-3.1ubuntu2) ...
```

Rufen Sie die IP-Adressen der kabelgebundenen und drahtlosen Netzwerkschnittstellen ab:

Öffnen Sie das Terminal und geben Sie den Befehl ein:

```
ifconfig
```

Die kabelgebundene Netzwerkadresse wird auf dem Steuerboard auf 192.168.96.3 eingestellt.

Notieren Sie die Adresse von wlan0 (z.B. 192.168.110.82, die Adresse kann je nach dem Router variieren) für die nachfolgende SSH-Verbindung.

```
orangeipi@orangeipi5pro:~/cyber1_dev_sdk$ 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 02:42:48:05:03:7d 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

enP4p65s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.96.3 netmask 255.255.255.0 broadcast 192.168.96.255
    inet6 fe80::c274:2b1f:efc:e25e prefixlen 64 scopeid 0x20<link>
    ether c0:74:2b:fc:e2:5e txqueuelen 1000 (Ethernet)
    RX packets 47 bytes 3027 (3.0 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 16 bytes 1290 (1.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 136 base 0x8000

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 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=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.110.82 netmask 255.255.255.0 broadcast 192.168.110.255
    inet6 fe80::9938:4280:63fc:ef67 prefixlen 64 scopeid 0x20<link>
    ether 9c:b8:b4:8c:34:bc txqueuelen 1000 (Ethernet)
```

Falls die IP-Adresse der kabelgebundenen Netzwerkschnittstelle nicht angezeigt werden kann, konfigurieren Sie zunächst eine statische IP als 192.168.96.3. Für spezifische Konfigurationsschritte verweisen Sie auf die statische IP-Konfigurationsmethode im Punkt 6.

```
TX packets 0 bytes 0 (0.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp4p6s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
ether c0:74:2b:fc:e2:5e 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 136 base 0x8000

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 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=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.110.82 netmask 255.255.255.0 broadcast 192.168.110.255
inet6 fe80::9938:4280:63fc:ef67 prefixlen 64 scopeid 0x20<link>
ether 9c:b8:b4:8c:34:bc txqueuelen 1000 (Ethernet)
RX packets 1242 bytes 350428 (350.4 KB)
RX errors 0 dropped 156 overruns 0 frame 0
TX packets 208 bytes 26702 (26.7 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

orangeipi@orangeipi5pro:~$
```

Kompilieren Sie das SDK

Kopieren Sie das komprimierte SDK-Paket auf die Entwicklungsplatte (empfohlen wird, es im Benutzerverzeichnis zu platzieren).

```
orangeipi@orangeipi5pro:~$ ls
cyber1_dev_sdk.zip Desktop Documents Downloads Music Pictures Public Templates Videos
```

Geben Sie im Terminal den Befehl ein, um die Datei cyber1_dev_sdk.zip zu entpacken:

```
unzip cyber1_dev_sdk.zip
```

```
orangeipi@orangeipi5pro:~$ unzip cyber1_dev_sdk.zip
Archive: cyber1_dev_sdk.zip
  creating: cyber1_dev_sdk/
  creating: cyber1_dev_sdk/lib/
  inflating: cyber1_dev_sdk/lib/udp_client.o
  inflating: cyber1_dev_sdk/lib/imu_data.o
  inflating: cyber1_dev_sdk/lib/json_parser.o
  inflating: cyber1_dev_sdk/lib/libquadruped_sdk.a
  inflating: cyber1_dev_sdk/Makefile
  creating: cyber1_dev_sdk/src/
  inflating: cyber1_dev_sdk/src/udp_client.cpp
  inflating: cyber1_dev_sdk/src/json_parser.cpp
  inflating: cyber1_dev_sdk/src/imu_data.cpp
  creating: cyber1_dev_sdk/include/
  creating: cyber1_dev_sdk/include/cyber1_dev_sdk/
  inflating: cyber1_dev_sdk/include/cyber1_dev_sdk/udp_client.h
  inflating: cyber1_dev_sdk/include/cyber1_dev_sdk/json_parser.h
  inflating: cyber1_dev_sdk/include/cyber1_dev_sdk/imu_data.h
  creating: cyber1_dev_sdk/examples/
  inflating: cyber1_dev_sdk/examples/imu_client_example.cpp
  inflating: cyber1_dev_sdk/examples/example.cpp
  inflating: cyber1_dev_sdk/examples/keyboard_test.cpp
  creating: cyber1_dev_sdk/bin/
  inflating: cyber1_dev_sdk/bin/example
  inflating: cyber1_dev_sdk/bin/keyboard_test
  inflating: cyber1_dev_sdk/bin/imu_client_example
```

Wechseln Sie in das Verzeichnis cyber1_dev_sdk und führen Sie zunächst den Befehl aus:

```
make clean
```

```
orangeipi@orangeipi5pro:~$ cd cyber1_dev_sdk/
orangeipi@orangeipi5pro:~/cyber1_dev_sdk$ make clean
清理构建文件...
✓ 清理完成
```

Kompilieren Sie das SDK: führen Sie den Befehl aus

make

```
orangeipi@orangeipi5pro:~/cyber1_dev_sdk$ make -j8
编译 src/udp_client.cpp...
编译 src/imu_data.cpp...
编译 src/json_parser.cpp...
g++ -std=c++11 -Wall -Wextra -pedantic -pthread -Iinclude -fPIC -c src/udp_client.cpp -o lib/udp_client.o
g++ -std=c++11 -Wall -Wextra -pedantic -pthread -Iinclude -fPIC -c src/imu_data.cpp -o lib/imu_data.o
g++ -std=c++11 -Wall -Wextra -pedantic -pthread -Iinclude -fPIC -c src/json_parser.cpp -o lib/json_parser.o
创建静态库 lib/libquadruped_sdk.a...
ar rcs lib/libquadruped_sdk.a lib/udp_client.o lib/json_parser.o lib/imu_data.o
✓ 静态库创建完成
编译测试 bin/example...
编译测试 bin/keyboard_test...
编译测试 bin/imu_client_example...
g++ -std=c++11 -Wall -Wextra -pedantic -pthread -Iinclude examples/example.cpp lib/libquadruped_sdk.a -o bin/example -ljsoncpp -lm
g++ -std=c++11 -Wall -Wextra -pedantic -pthread -Iinclude examples/keyboard_test.cpp lib/libquadruped_sdk.a -o bin/keyboard_test -ljsoncpp -lm
g++ -std=c++11 -Wall -Wextra -pedantic -pthread -Iinclude examples/imu_client_example.cpp lib/libquadruped_sdk.a -o bin/imu_client_example -ljsoncpp -lm
examples/keyboard_test.cpp: In function 'int main(int, char**)':
examples/keyboard_test.cpp:212:17: warning: comparison is always true due to limited range of data type [-Wtype-limits]
 212 |         if (key != EOF) { // 有按键输入
      |                 ^
examples/imu_client_example.cpp: In function 'void signalHandler(int)':
examples/imu_client_example.cpp:20:24: warning: unused parameter 'signal' [-Wunused-parameter]
 20 | void signalHandler(int signal) {
      |                      ^~~~~~
✓ 测试编译完成
✓ 测试编译完成
✓ 测试编译完成
```

Im Normalfall wird die Kompilierung die oben genannte Information anzeigen

(2) Aufbau der Hardwareumgebung

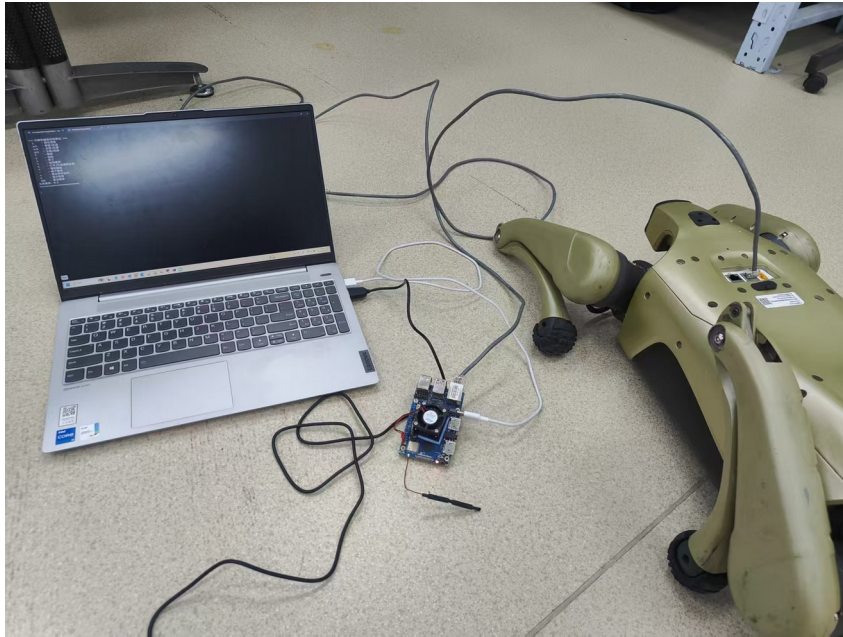
Stellen Sie sicher, dass keine Hindernisse innerhalb von 3 Metern vom Roboterhund liegen.

Verbinden Sie die Entwicklungsplatte mit dem Erweiterungs-Ethernet-Port des Roboterhundes über ein Netzkabel.

Verbinden Sie den Computer mit der Steuerplatte zur Energieversorgung über ein USB-Kabel.

Stellen Sie sicher, dass der Computer und die Steuerplatte mit dem gleichen WLAN verbunden sind.

Hinweis: Starten Sie den Fernbedienung nicht ein.



Öffnen Sie die Eingabeaufforderung (cmd) auf dem Computer.

Geben Sie den Befehl ein:

```
ssh orangepi@192.168.110.82
```

Geben Sie Ja ein, wenn Sie nach Ja/Nein gefragt werden.

Geben Sie das Passwort ein (eingebaut in die Steuerplatte).

```
orangepi@orangepi5pro: ~  
Microsoft Windows [版本 10.0.22631.6060]  
(c) Microsoft Corporation. 保留所有权利。  
  
C:\Users\admin>ssh orangepi@192.168.110.82  
The authenticity of host '192.168.110.82 (192.168.110.82)' can't be established.  
ED25519 key fingerprint is SHA256:kwzqTbFiwUkyH+lv3LIX7uvZzDixQEvgbT407DqaBVY.  
This key is not known by any other names.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '192.168.110.82' (ED25519) to the list of known hosts.  
orangepi@192.168.110.82's password:  
  
OPIS PRO  
  
Welcome to Orange Pi 1.0.6 Focal with Linux 5.10.160-rockchip-rk3588  
  
System load: 13%      Up time: 1 min      Local users: 3  
Memory usage: 7% of 7.75G      IP: 172.17.0.1 192.168.110.82  
CPU temp: 25°C      Usage of /: 70% of 29G  
  
[ 78 security updates available, 233 updates total: apt upgrade ]  
Last check: 2025-12-26 11:22  
  
Last login: Fri Dec 26 11:22:18 2025  
orangepi@orangepi5pro:~$
```

(3) Führen Sie die SDK-Beispielprogramme aus

Halten Sie die Energie-Taste am Rücken des Roboterhundes gedrückt für mehr als 3 Sekunden, um ihn einzuschalten. Der Roboterhund wurde erfolgreich gestartet, nachdem Sie die Sprachnachricht "Roboterhund-Start abgeschlossen" gehört haben.

Führen Sie das Programmbeispiel aus

Hinweis: Der Roboterhund wird eine Reihe kontinuierlicher Aktionen ausführen, einschließlich Homing-Reset, vorwärtsbewegen,

rückwärtsbewegen, nach links verschieben, nach rechts verschieben, nach links drehen, nach rechts drehen und liegen. Stellen Sie sicher, dass keine Hindernisse innerhalb von 3 Metern sind.

Geben Sie den Befehl in der Kommandozeile des Computers ein:

```
./bin/example
```

```
orangeipi@orangeipi5pro:~/cyber1_dev_sdk$ ./bin/example
=== UDP客户端已启动 ===
目标服务器: 192.168.96.2:8080
Trying to connect udp server...
=== 成功连接到UDP服务器 ===:
后台接收线程已启动
后台发送线程已启动
UDP send loop started at 1.0 Hz
=== change mode to Reset ===
=== change mode to forward ===
=== change mode to backward ===
=== change mode to move_left ===
=== change mode to move_right ===
=== change mode to turn_left ===
=== change mode to turn_right ===
=== change mode to Lie Down ===
=== change mode to 10 ===
UDP send loop stopped后台发送线程已停止
后台接收线程已停止
客户端连接已关闭
```

Führen Sie das Programm keyboard_test aus:

Geben Sie den Befehl in der Kommandozeile des Computers ein:

```
./bin/keyboard_test
```

Die vom Roboterhund ausgeführten Aktionen, die den Tasten auf der Tastatur entsprechen, werden angezeigt. Führen Sie die Befehle entsprechend den Tasten aus, und drücken Sie die h-Taste, um den Bildschirm zu löschen und die Tastaturaufforderung erneut anzuzeigen (die Zurücksetzaktion des Roboterhundes ist ungültig, wenn der obige ./bin/example-Befehl ausgeführt wurde):

```
=== 机器狗键盘控制测试 ===
r      - 复位回零
w/s    - 前进/后退
a/d    - 左移/右移
q/e    - 左转/右转
t      - 急停
y      - 站起
u      - 趴下
i      - 切换模型
o      - 打开/关闭障碍设置
+      - 增加速度
-      - 减少速度
x      - 停止所有运动
h      - 显示帮助
ESC    - 退出程序
当前速度: 0.5
=====
```

Starten Sie das Beispiel zur Anzeige der IMU-Daten:

Geben Sie den Befehl in der Kommandozeile des Computers ein:

```
./bin/imu_client_example
```

Die aktuellen IMU-Daten werden angezeigt.

```

Angular Velocity:
X: -0.0016 rad/s
Y: 0.0021 rad/s
Z: -0.0004 rad/s

Orientation (Quaternion):
W: 0.9756
X: -0.0044
Y: 0.0106
Z: -0.2193

Euler Angles:
Roll: -0.75 °
Pitch: 1.07 °
Yaw: -25.35 °

Magnitudes:
Acceleration: 9.759 m/s²
Angular Vel: 0.003 rad/s
=====
IMU Data - Time: 1766653703.860 | Accel: (-0.21, -0.15, 9.80) m/s² | Gyro: (-0.00, 0.00, 0.00) rad/s | RPY: (-0.8,
IMU Data - Time: 1766653703.912 | Accel: (-0.17, -0.15, 9.80) m/s² | Gyro: (-0.00, -0.00, -0.00) rad/s | RPY: (-0.8,
1.1, -25.3) degIMU Data - Accel: (-0.2, -0.1, 9.8) m/s², RPY: (-0.8, 1.1, -25.3) deg
IMU Data - Time: 1766653703.960 | Accel: (-0.18, -0.14, 9.81) m/s² | Gyro: (-0.00, 0.00, -0.00) rad/s | RPY: (-0.8,
IMU Data - Time: 1766653704.012 | Accel: (-0.16, -0.13, 9.81) m/s² | Gyro: (-0.00, 0.00, 0.00) rad/s | RPY: (-0.7,
IMU Data - Time: 1766653704.060 | Accel: (-0.18, -0.14, 9.78) m/s² | Gyro: (0.00, -0.00, -0.00) rad/s | RPY: (-0.7,
IMU Data - Time: 1766653704.112 | Accel: (-0.21, -0.10, 9.79) m/s² | Gyro: (0.00, 0.00, -0.00) rad/s | RPY: (-0.7,
IMU Data - Time: 1766653704.162 | Accel: (-0.18, -0.14, 9.78) m/s² | Gyro: (-0.00, -0.00, 0.00) rad/s | RPY: (-0.7,
IMU Data - Time: 1766653704.342 | Accel: (-0.18, -0.13, 9.79) m/s² | Gyro: (-0.00, 0.00, 0.00) rad/s | RPY: (-0.7,
IMU Data - Time: 1766653704.392 | Accel: (-0.17, -0.15, 9.78) m/s² | Gyro: (-0.00, -0.00, -0.00) rad/s | RPY: (-0.7,
IMU Data - Time: 1766653704.442 | Accel: (-0.19, -0.12, 9.77) m/s² | Gyro: (0.00, 0.00, 0.00) rad/s | RPY: (-0.7,
1.1, -25.4) deg

```

5. SDK API-Beschreibung

(1) Netzwerkverbindung

Erstellen eines UDP-Clients

```

std::string server_ip = "192.168.96.2";
int server_port = 8080;
UDPClient client(server_ip, server_port);

```

Erstellen eines JsonParser-Parsers

```

JsonParser parser(std::move(client));
parser.run();

```

Verbindung herstellen

Aufgrund der Zeitunterschiede beim Systemstart zwischen dem Roboterhund und der Steuerplatte muss die Steuerplatte überprüfen, ob der Roboterhund vollständig

geladen wurde. Verwenden Sie die folgende Operation und die testConnection-Methode, um zu bestimmen, ob der Roboterhund-Server verbunden ist:

```
bool isConnect = client.testConnection();
```

isConnect ist wahr bedeutet, dass der Roboterhund-Server verbunden ist.

(2) Roboterhund-Status abrufen

Funktionsname: getDogStatus

Parameter: Keine

Rückgabotyp: DogStatus

Verwendung:

```
DogStatus status = parser.getDogStatus();
```

Details der Statusinformationen:

- battery: Batteriestand
- motion_mode: Aktueller Bewegungsmodus des Roboterhundes (0 - Liegend ohne Homing, 1 - Liegend, 2 - Stehend, 3 - Niedrige Haltung, 4 - Umgekippt)
- velocity_level: Geschwindigkeitsstufe (aktuell nicht verwendet)
- model: 1 - Hochgeschwindigkeitsmodus, 0 - Hinderniserkennungsmodus
- hw_failure: Liste defekter Hardware
- software_version: Softwareversion
- latest_version: Die neueste Softwareversion auf dem Server
- robot_number: Seriennummer des Roboterhundes
- robot_type: Roboterhund-Modell
- hardware_error: Ob ein Fehler vorliegt (Gesamtfehler) (0 - Kein Fehler, 1 - Fehler existiert)

(3) Homing-Reset

Funktionsname: handleResetCommand

Parameter: None

Rückgabewert: None

Verwendung:

```
parser.handleResetCommand();
```

(4) Aufstehen

Funktionsname: handleStandUpCommand

Parameter: None

Rückgabewert: None

Verwendung:

```
parser.handleStandUpCommand()
```

(5) Liegen

Funktionsname: handleLieDownCommand

Parameter: None

Rückgabewert: None

Verwendung:

```
parser.handleLieDownCommand();
```

(6) Notaus

Funktionsname: handleEmergencyCommand

Parameter: None

Rückgabewert: None

Verwendung:

```
parser.handleEmergencyCommand();
```

(7) Modell wechseln

Funktionsname: handleSwitchMnnCommand

Parametertyp: int

Parameterbedeutung: 0 - Zum Hinderniserkennungsmodus wechseln, 1 - Zum Hochgeschwindigkeitsmodus wechseln

Rückgabewert: None

Verwendung:

```
parser.handleSwitchMnnCommand(1);
```

(8) Ultraschallradar-Hinderniserkennung einstellen

Funktionsname: handleSwitchObstacleCommand

Parameter-Typ: int

Parameter-Bedeutung: 0 - Hindernisvermeidung deaktivieren, 1 - Hindernisvermeidung aktivieren

Rückgabewert: None

Verwendung:

```
parser.handleSwitchMnnCommand(1);
```

(9) Bewegungssteuerung

Methode 1

Funktionsname: sendDirectionVel

Parameter-Typ: int direction, double velocity

Parameter-Bedeutung:

- Der erste Parameter entspricht der Richtung: 1 - Vorwärts/Rückwärts, 2 - Links/Rechts, 3 - Links/Rechts drehen
- Der zweite Parameter entspricht der Geschwindigkeit: Positive Werte entsprechen vorwärts, links nach links und links nach links; Negative Werte entsprechen rückwärts, rechts nach rechts und rechts nach rechts.
- Der Bereich des zweiten Parameters: -1 bis 1

Rückgabewert: None

Verwendung:

```
parser.sendDirectionVel(0.5 0 0); // 0.5
```

(10) Registrieren des IMU-Clients

Funktionsname: enableIMUStream

Parameter: None

Rückgabotyp: bool

Verwendung:

```
// Setzen des IMU-Daten-Callbacks
parser.setIMUDataCallback(imuDataCallback)
// Starte den Parser
parser.run();
bool result = parser.enableIMUStream();
```

(11) IMU-Dateninformationen abrufen

Funktionsname: getLatestIMUData

Parameter: None

Rückgabotyp: IMUData

Verwendung:

```
// Setze IMU-Daten-Callback
parser.setIMUDataCallback(imuDataCallback)
// Starte den Parser
parser.run();
// Registriere IMU-Client
bool result = parser.enableIMUStream();
// Abrufen von IMU-Dateninformationen
IMUData imu_data = parser.getLatestIMUData(); //IMUData Details see protocol
documentation
```

(12) Unregister IMU Client (No IMU information will be read after unregistration)

Function Name: disableIMUStream

Parameters: None

Return Value Type: bool

Usage:

```
bool result = parser.disableIMUStream
```

6. Static IP Configuration Method

(1) Erstellen Sie /etc/netplan/01-network-manager-all.yaml mit folgendem Inhalt. Speichern Sie die Datei und führen Sie anschließend (2) sudo reboot aus oder starten Sie die Stromversorgung neu:

```
# Let NetworkManager manage all devices on this system
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    enP4p65s0:
      dhcp4: no
      dhcp6: no
      addresses: [192.168.96.3/24]
      routes:
        - to: default
          via: 192.168.96.1
      nameservers:
        addresses: [114.114.114.114, 8.8.8.8]
```