



ЦЕНТР МОЛОДЁЖНОЙ  
РОБОТОТЕХНИКИ  
МГТУ ИМ. Н.Э. БАУМАНА

# Базовое управление четвероногими роботами

# Научно-исследовательский кружок по ИИ в робототехнике

Организован при содействии  
Лаборатории Робототехники  
Сбербанка

## Шагающие роботы:

Unitree Aliengo  
Tinker

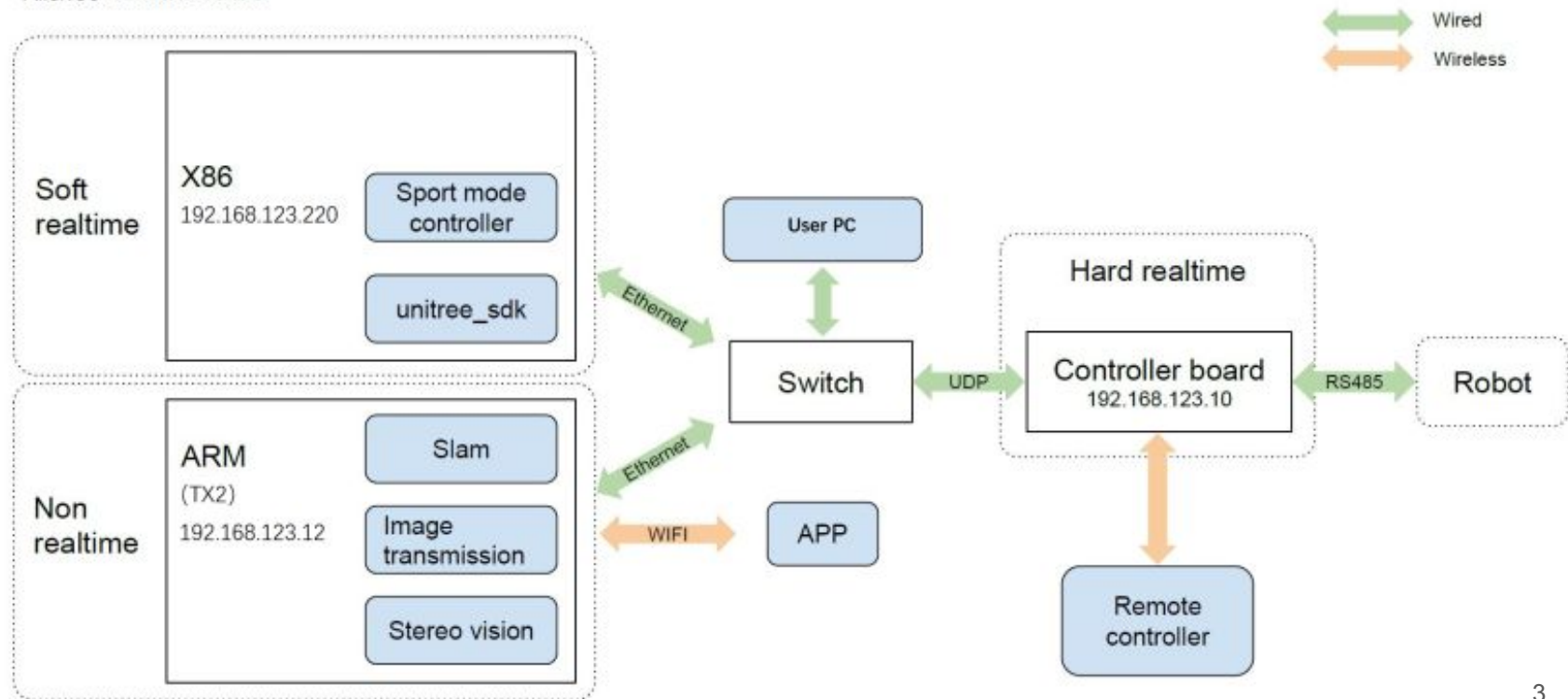
## Зачем кружок:

- Написание НИР, дипломов и статей по шагающим роботам
- Выступление на конференциях
- Создание своих проектов на основе шагающего робота



# Архитектура Unitree-робота

AlienGo architecture



## Режимы работы

- low-level (192.168.123.10) -> Управляющая плата
- high-level (192.168.123.220) -> Mini PC

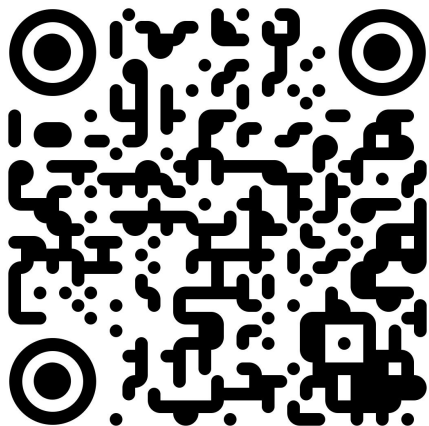
Переключить в low-level:

- L2+A, L2+A, L2+B - опуститься и расслабить моторы
- L1 + L2 + Start - отключение спортивного режима

# Код проекта

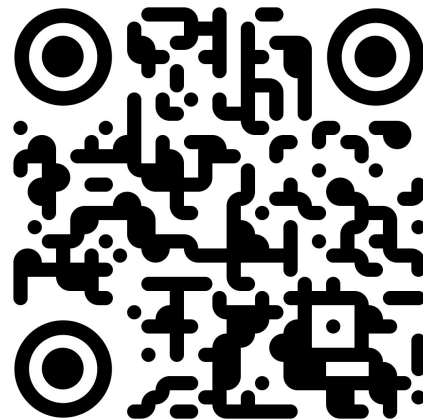
Оригинальное sdk для управления unitree-роботами. Нам нужна ветка **Aliengo**.

unitree\_legged\_sdk



**Проект-обертка**, в котором **unitree\_legged\_sdk** — один из подмодулей. С помощью **rl\_go1** скрипт без изменения может быть запущен как в симуляторе **Mujoco**, так и на реальном роботе

rl\_go1



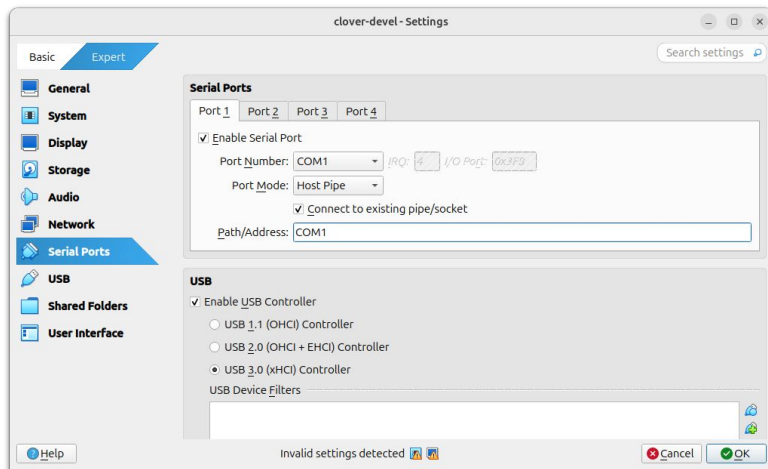
# Запуск через VirtualBox

В виртуальной машине:

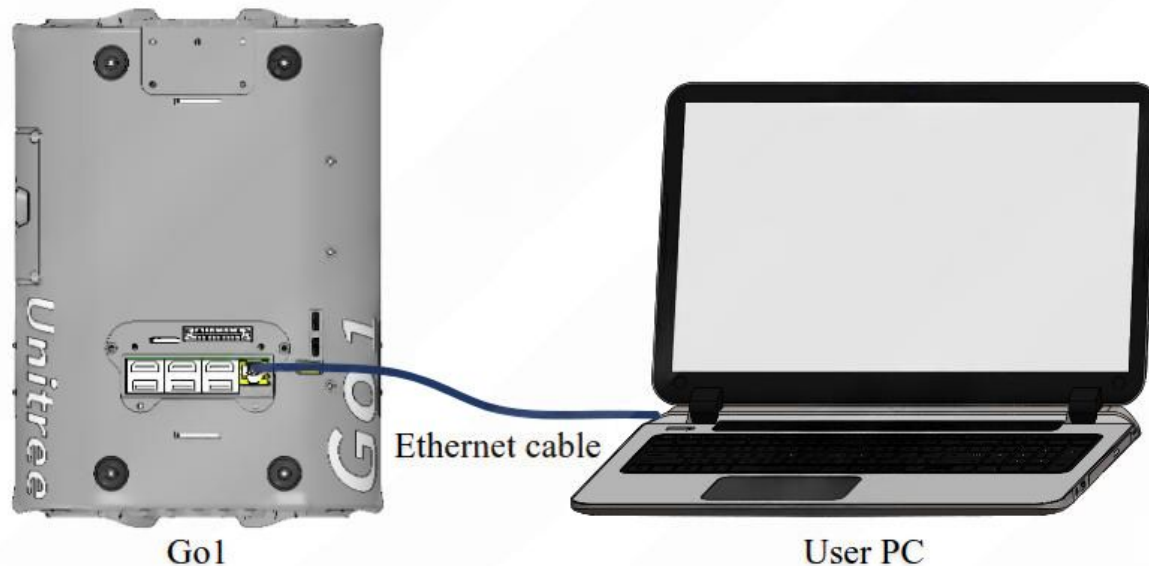
- VSCode
- Проект rl\_go1
  - unitree\_legged\_sdk
  - mujoco (симулятор)

Внутри контейнера выполнить:

```
conda activate rl_go1
cd rl_go1
python scripts/standup.py
```



# Управление через сдк



```
sudo ifconfig eth0 down # ethxxx is your PC port
sudo ifconfig eth0 192.168.123.200/24
sudo ifconfig eth0 up
ping 192.168.123.10
```

# Управление через сдк

CancelWiredApply

DetailsIdentityIPv4IPv6Security

**IPv4 Method**

☐ Automatic (DHCP)☐ Link-Local Only

☒ Manual☐ Disable

☐ Shared to other computers

**Addresses**

Address	Netmask	Gateway	
192.168.123.200	255.255.255.0		

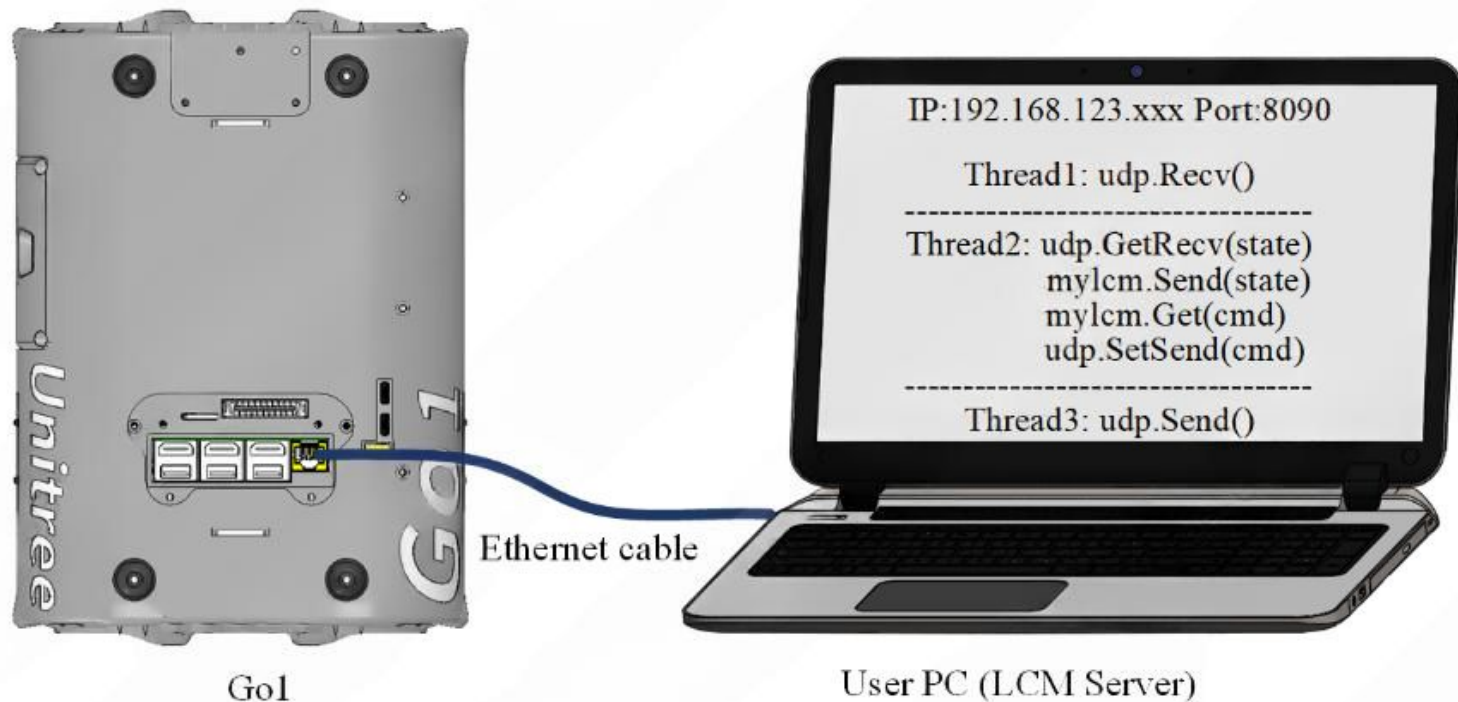
**DNS**

Automatic ☒

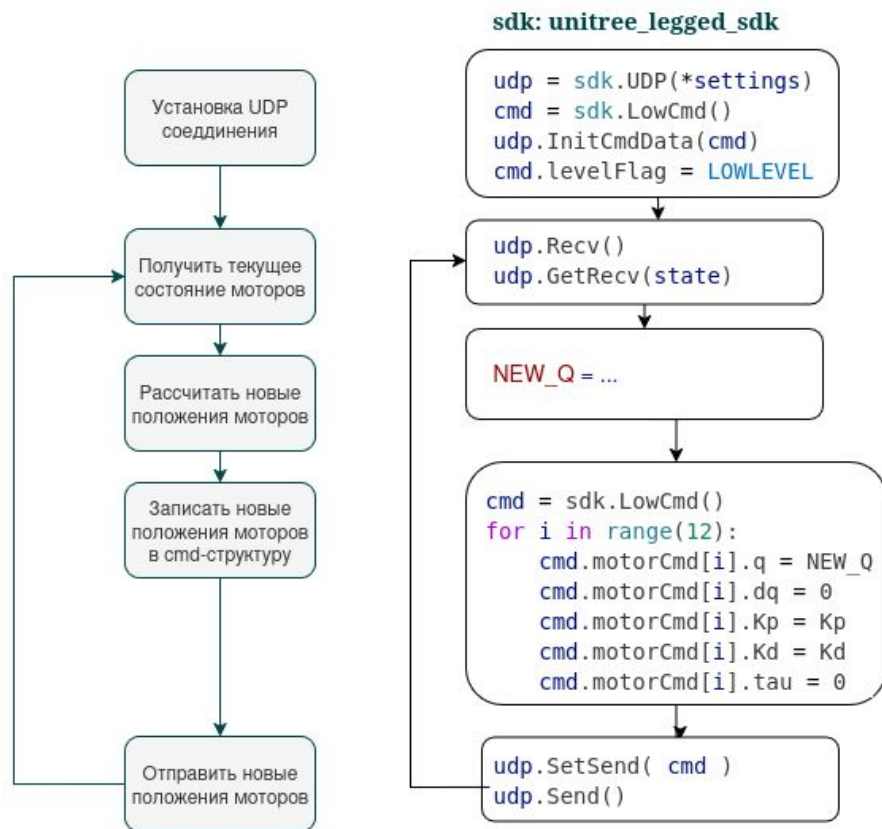
Separate IP addresses with commas



# Структура кода



# Структура кода



Нужно установить UDP-соединение с роботом, а затем в цикле получать текущие состояния в структуре **LowState**, рассчитывать на их основе новые — и отправлять их роботу в структуре **LowCmd**.

# Управление моторами

- Угловое положение ротора —  $q$
- Угловая скорость ротора —  $dq$
- Крутящий момент ротора —  $\tau$
- Пропорциональный коэффициент ПД-регулятора (коэффициент жесткости) —  $K_p$
- Дифференциальный коэффициент ПД-регулятора (коэффициент затухания) —  $K_d$



Position mode:

```
q = q(time)
dq = 0
Kp = Kp_const
Kd = Kd_const
tau = 0
```

Speed mode:

```
q = PosStopF
dq = dq(time)
Kp = 0
Kd = Kd_const
tau = 0
```

Torque mode:

```
q = PosStopF
dq = VelStopF
Kp = 0
Kd = 0
tau = tau(time)
```

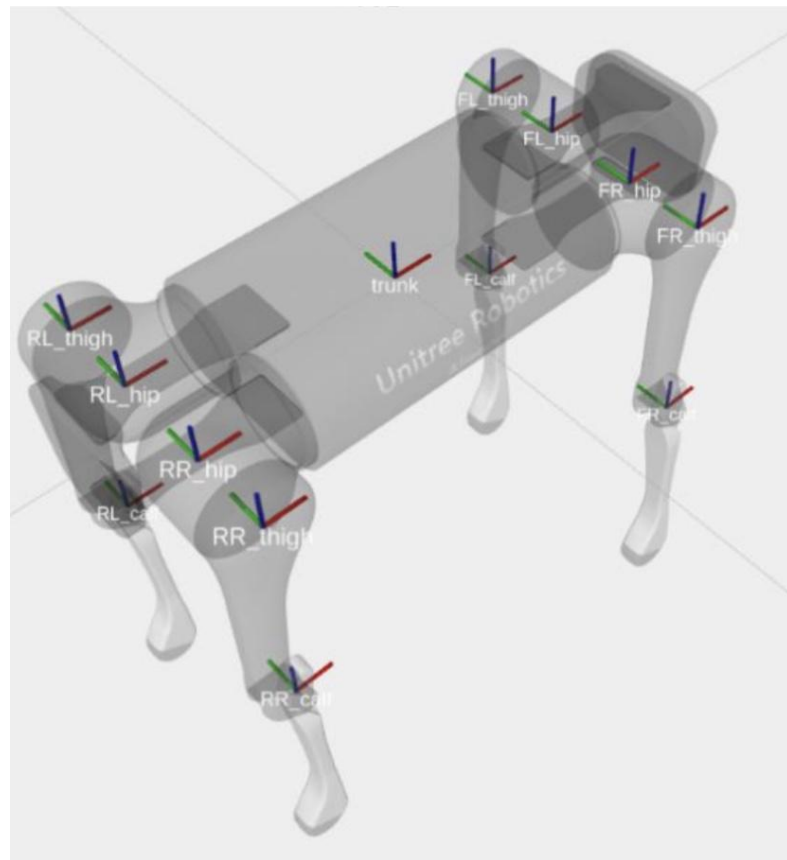
# Управление моторами

Суммарный крутящий момент:

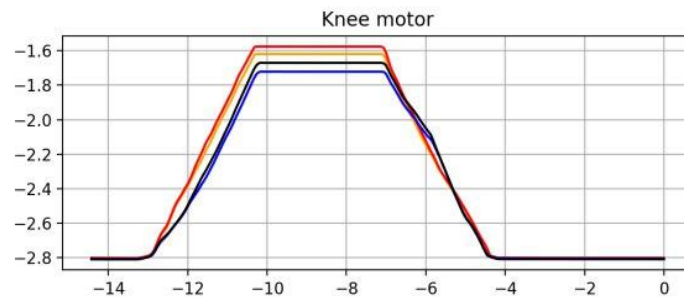
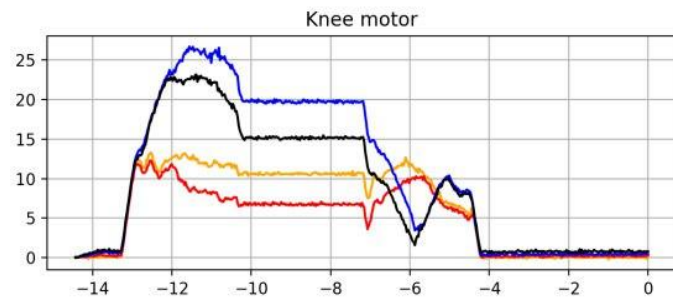
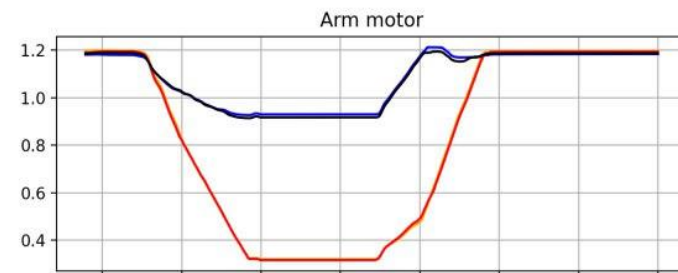
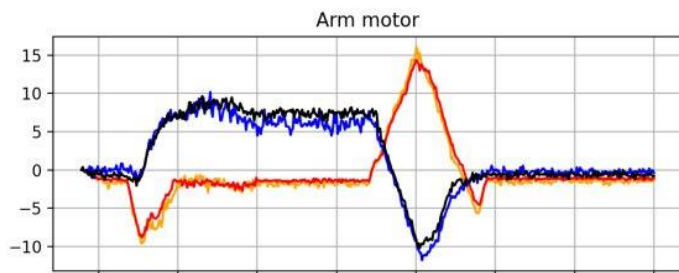
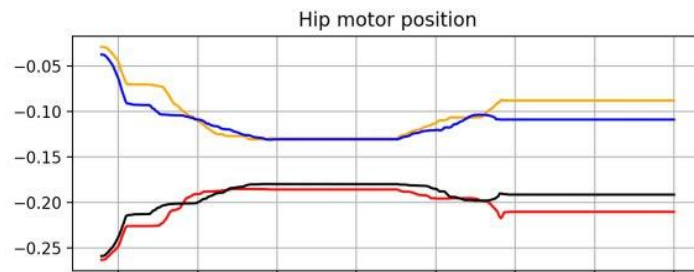
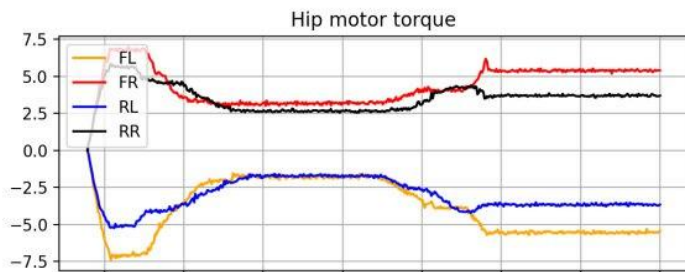
$$\text{out} = \tau + K_p * (q_{\text{des}} - q) + K_d * (dq_{\text{des}} - dq)$$

Состоит из:

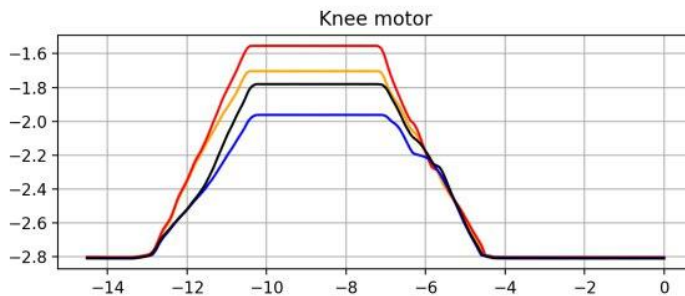
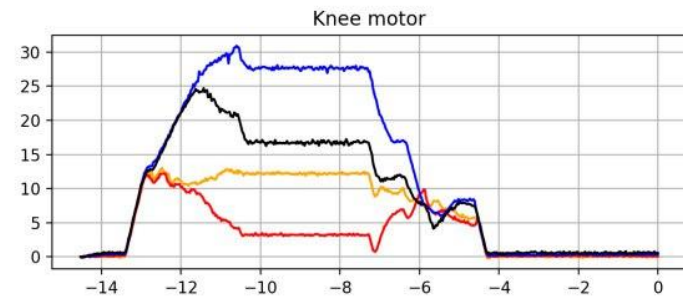
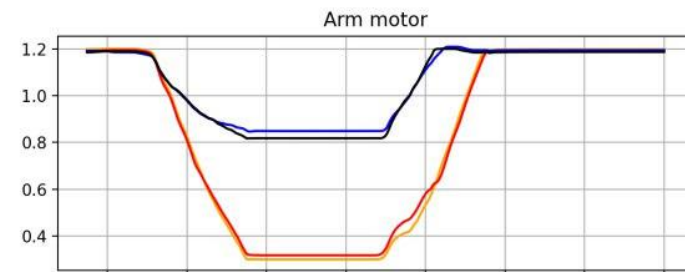
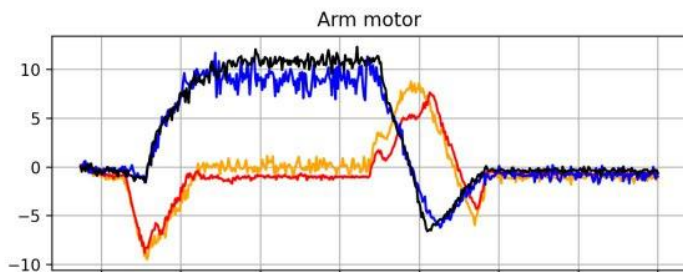
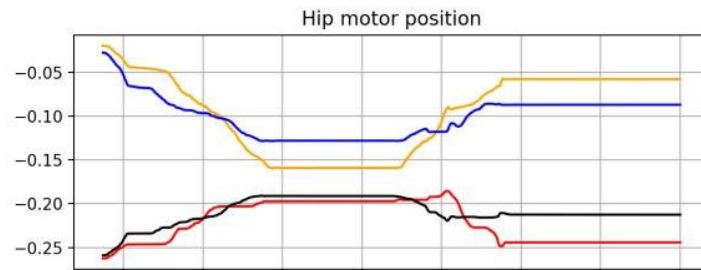
1.  $\tau$  - уменьшает ошибку регулятора, компенсируя известные силы
2.  $K_p * (q_{\text{des}} - q)$  - Пропорциональная составляющая
3.  $K_d * (dq_{\text{des}} - dq)$  - Дифференциальная составляющая



$K_p = 90, K_d = 2$

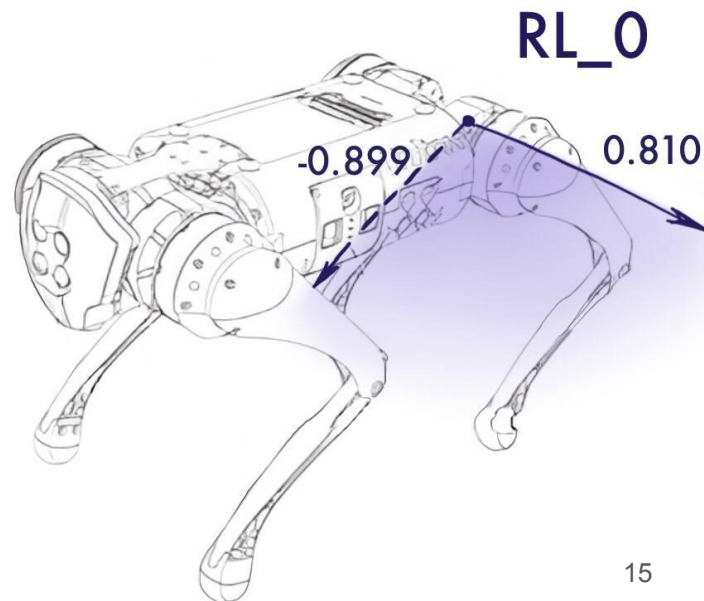
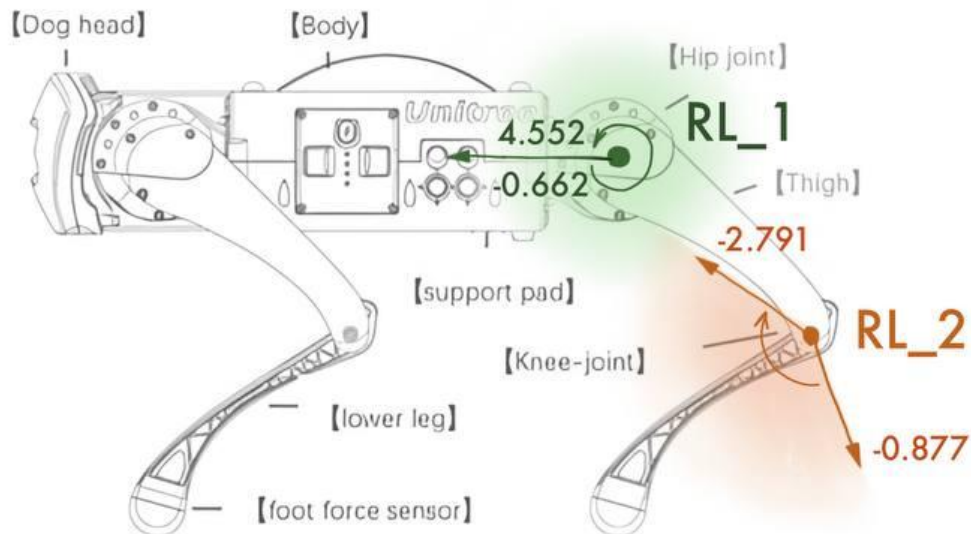


$K_p = 60, K_d = 2$

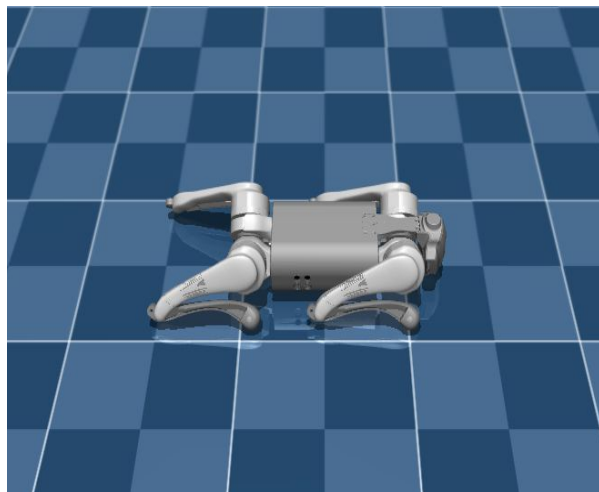


# Границы значений моторов

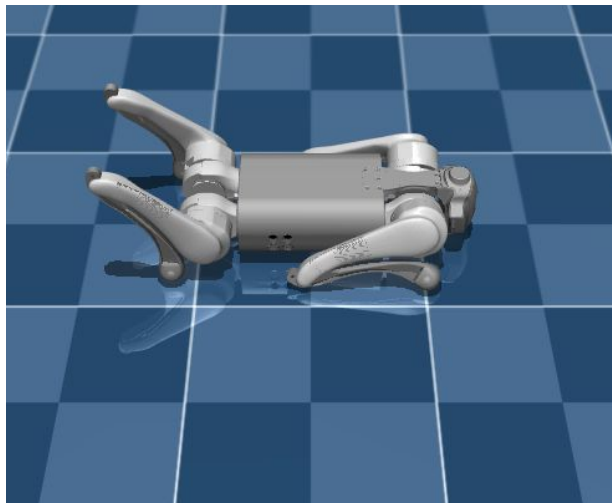
```
...  
( 'RL_0' , -0.899 , 0.810 ) , # hip -70°~70°  
( 'RL_1' , -0.662 , 4.552 ) , # thigh -360°-360°  
( 'RL_2' , -2.791 , -0.877 ) # calf -159°-37°  
...
```



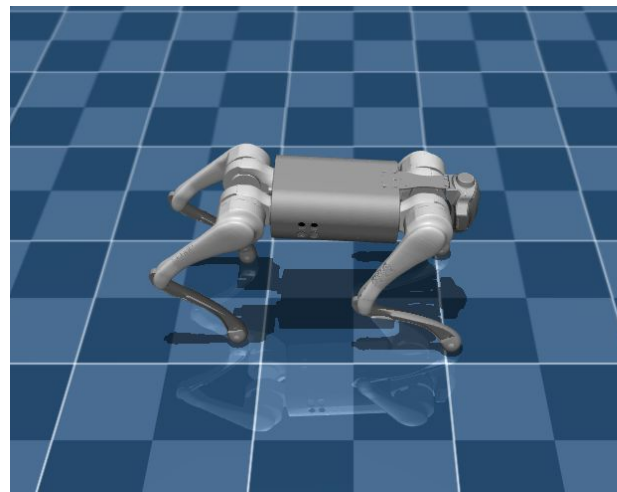
Фаза 1



Фаза 2



Фаза 3



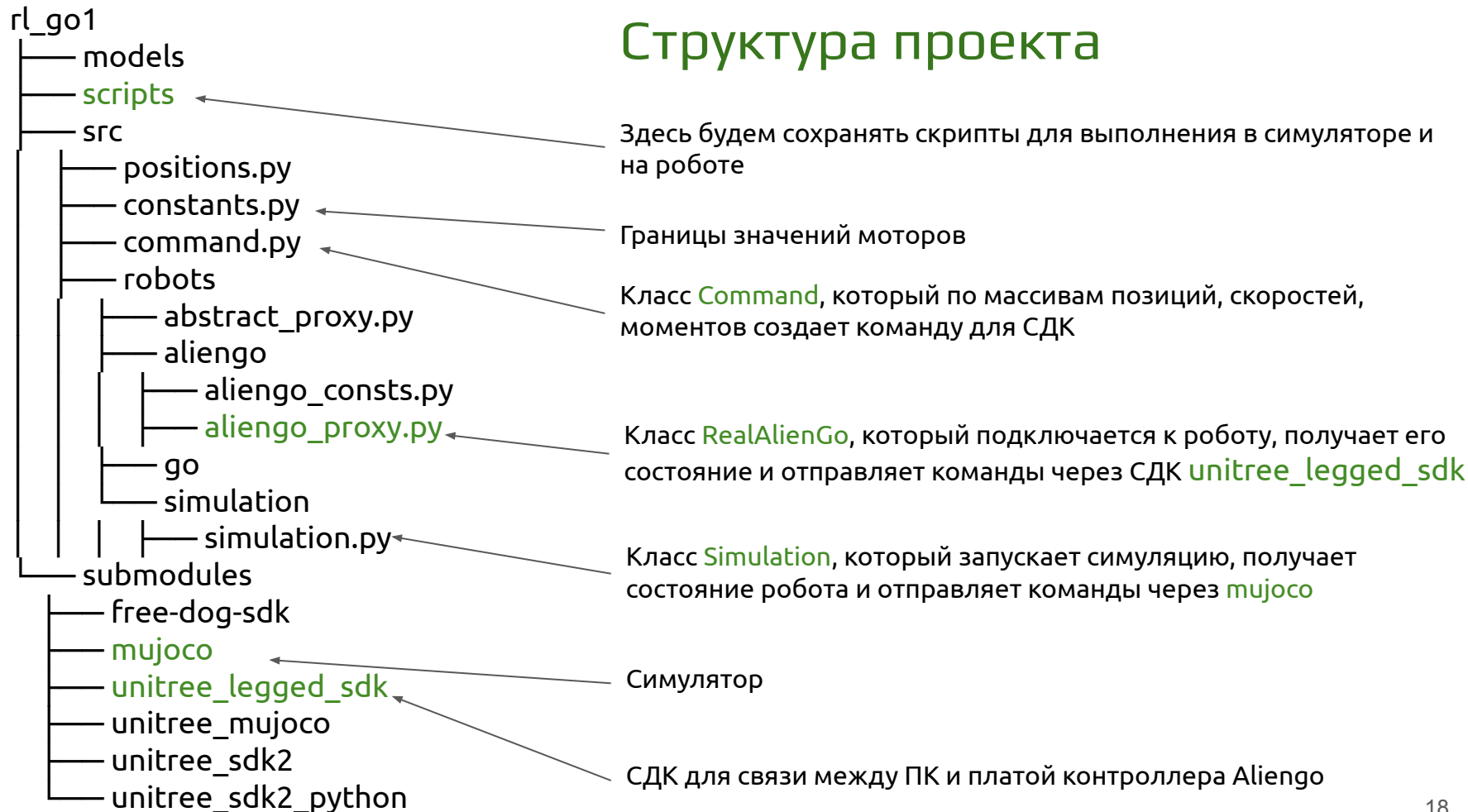


# Интерполяция

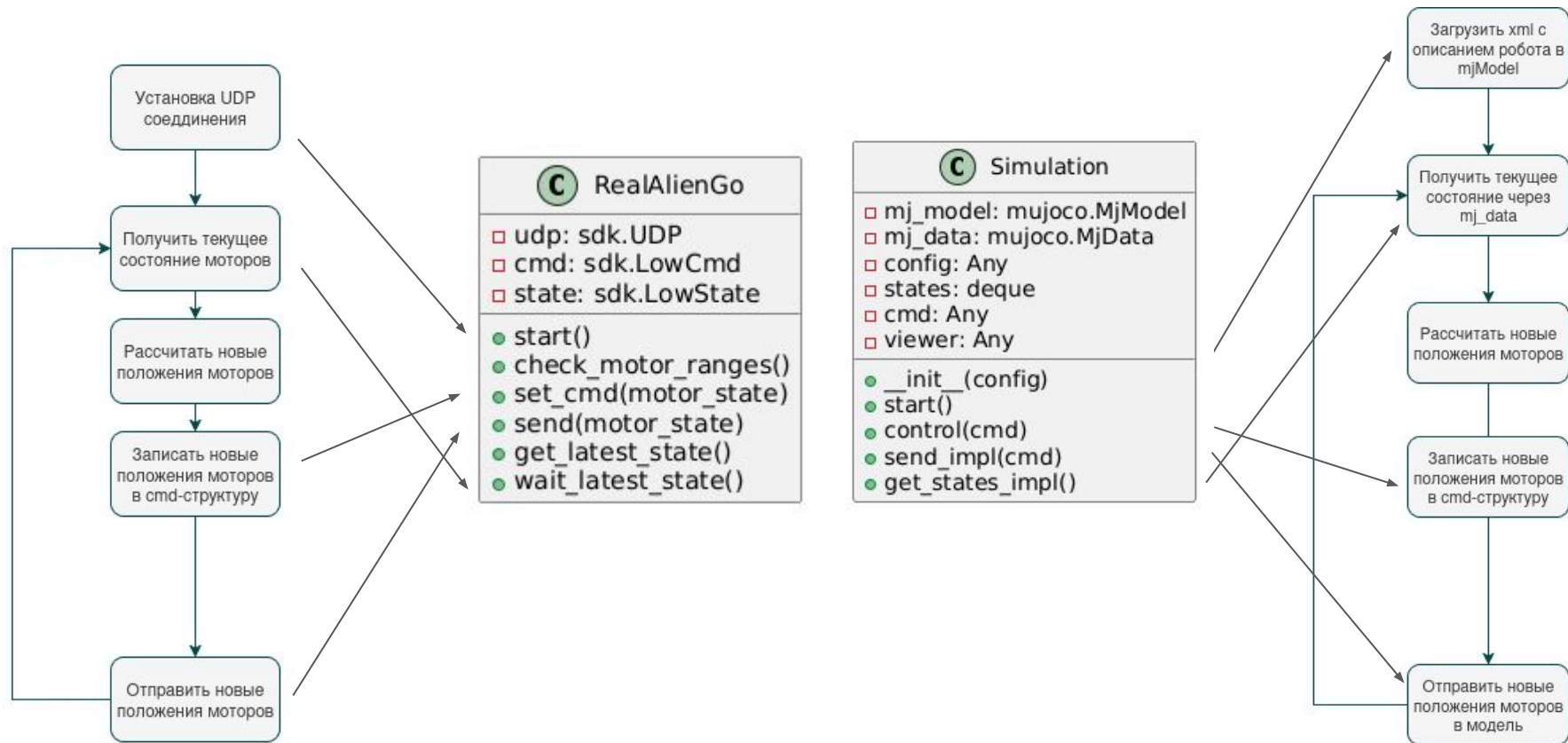
Нельзя за одну секунду резко изменить положение моторов - моторы сгорят.

Нужно разбить изменение на маленькие шаги, например, с помощью линейной интерполяции.

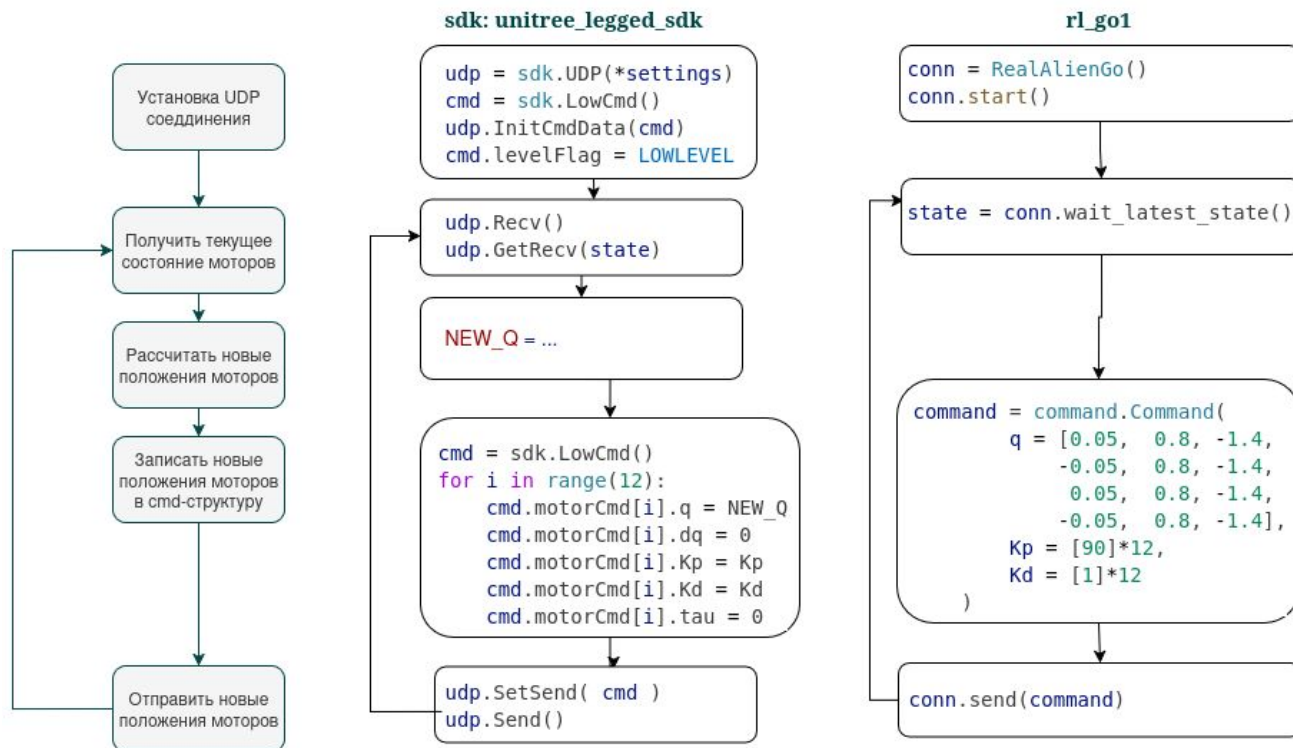
```
def jointLinearInterpolation(initPos, targetPos, rate):  
  
    rate = min(max(rate, 0.0), 1.0)  
  
    p = initPos*(1-rate) + targetPos*rate  
    return p
```



# Структура кода



# Структура кода

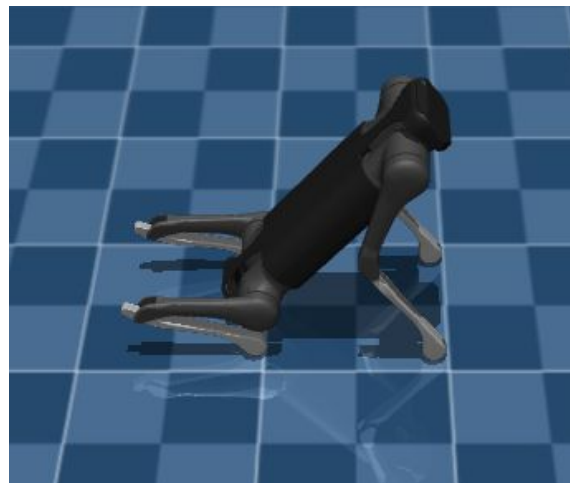


С помощью rl\_go1 скрипт без изменения может быть запущен как в симуляции, так и на реальном роботе

# Напишите свой скрипт

Измените скрипт `standup` — измените целевые значения  $q$  моторов, количество фаз. Например, можно научить робособаку сидеть.

Протестируйте скрипт в `mujoco`.



## Управление локомоцией

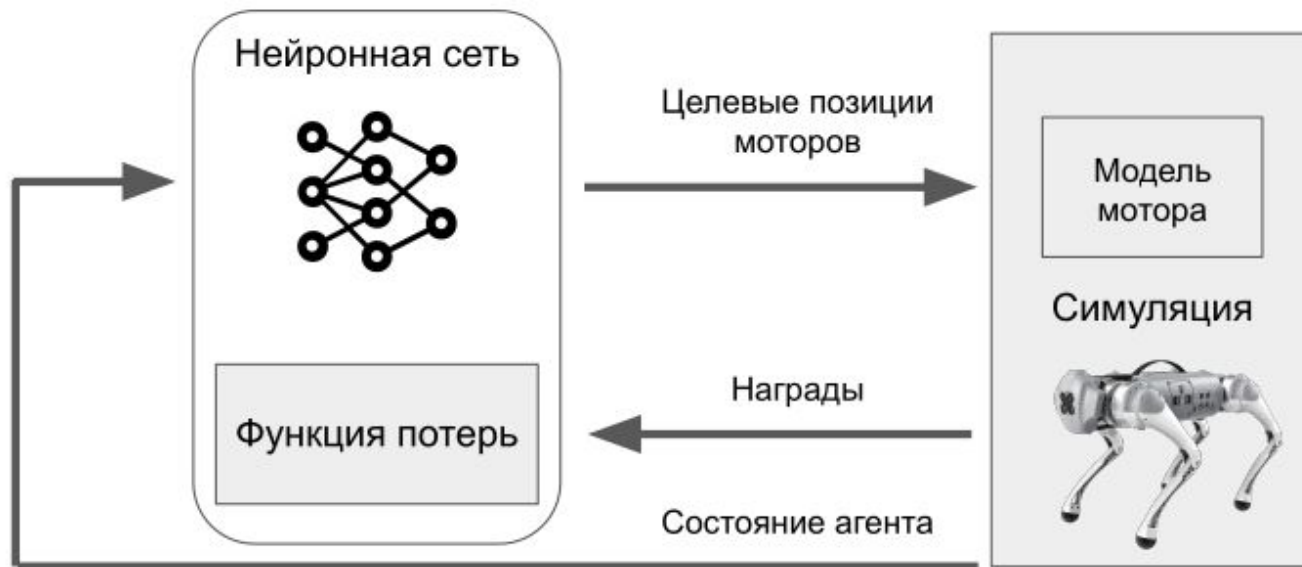
Методы оптимального  
управления

Гибридные методы

Методы глубокого обучения  
с подкреплением (RL)



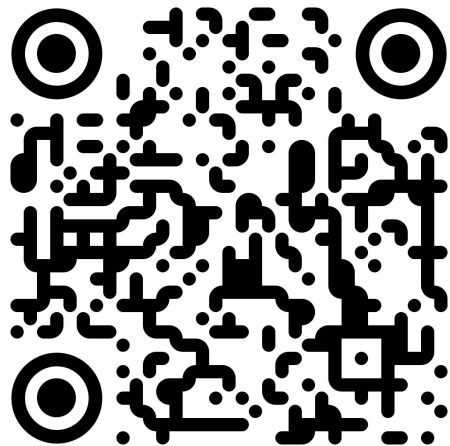
# Обучение с подкреплением



# Дополнительные материалы

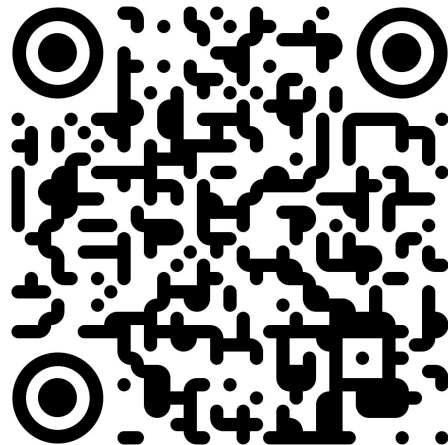
Сайт с гайдами и полезными ссылками  
про Unitree Aliengo и RL.

Документация

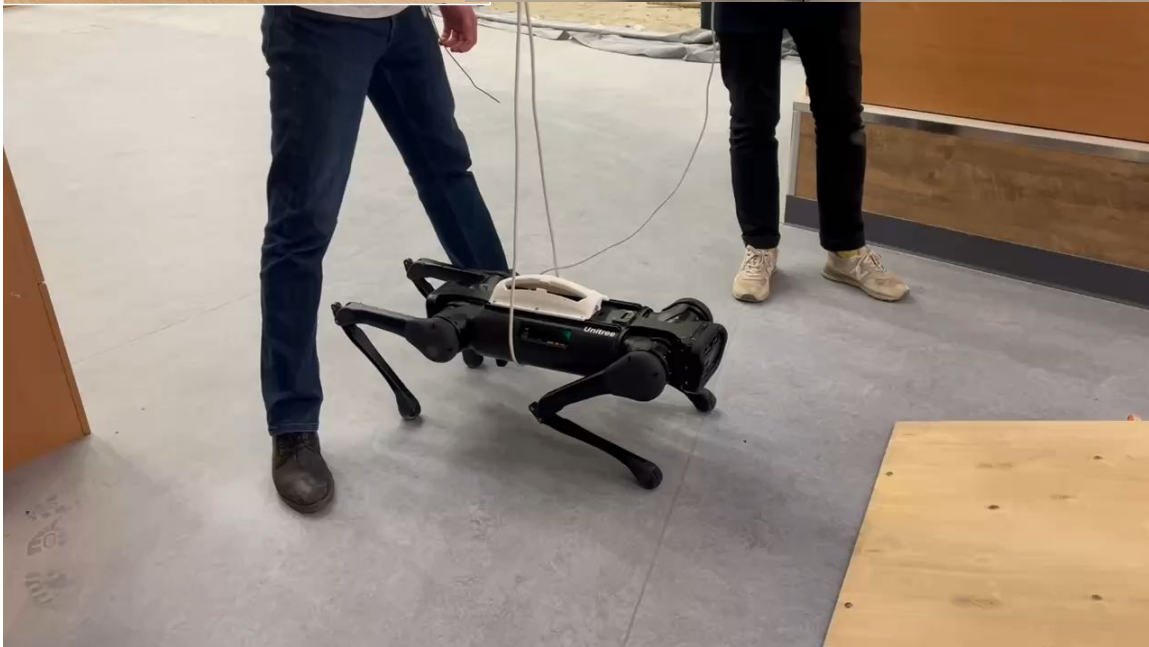
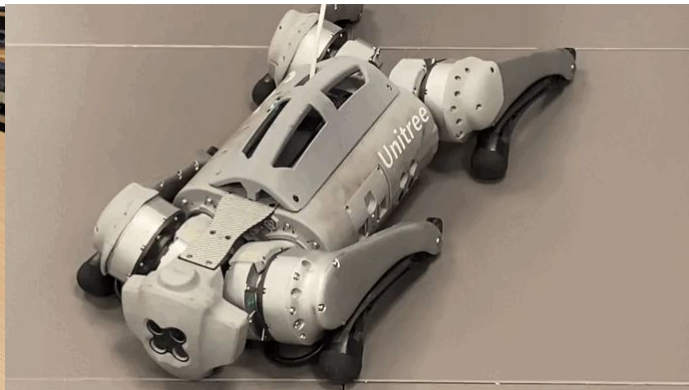
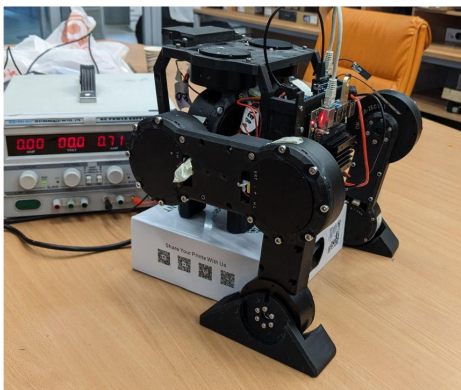


СДК для управления Aliengo через **ROS**  
**Noetic**. В проекте есть docker для быстрой  
сборки.

unitree\_ros\_to\_real





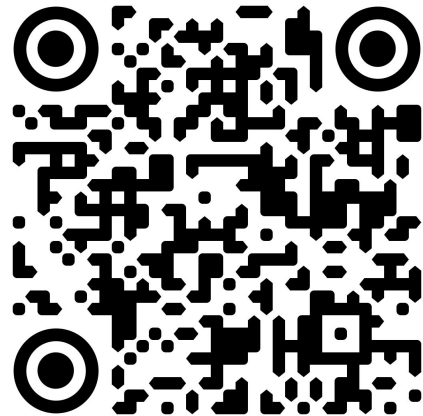


# Кружок шагающих роботов

Сборка и программирование  
шагающих роботов

Проект от **Центра Робототехники**  
**Сбербанка**

Статья на хабре



Спасибо за внимание

