

low_level_control_without_hands_node - низкоуровневое управление моторами робота Unitree H1

На GitHub представлена [инструкция](#) по установке репозитория, управляющего моторами робота Unitree H1. Выполните всё до 6 шага не включая. В этой же инструкции ниже можно найти описание всех модулей включая указанный. Мы будем рассматривать только работу с нодой `low_level_control_without_hands_node`, которая реализует безопасное управление моторами робота.



Что делает этот узел (`low_level_control_without_hands_node`)?

Этот Python-скрипт — **низкоуровневый ROS 2-узел**, который:

1. **Принимает команды на движение суставов** (например, "поднять правую руку", "согнуть левое колено") через специальный топик.
 2. Обеспечивает **плавное и безопасное** управление положением моторов.
 3. **Ограничивает скорость движений**, чтобы робот не дёргался резко.
 4. Использует **обратную связь** от робота (его текущее положение суставов), чтобы корректировать команды.
-



Основные компоненты

1. Топик для управления: `positions_to_unitree`

- Это **входной топик**, через который вы отправляете желаемые позиции суставов.
- Формат сообщения:

```
{"0": 0.5, "1": -0.3, ...}$0.8
```

- Левая часть — **JSON-объект**: ключи — номера суставов (см. таблицу ниже), значения — углы в радианах. Есть библиотека для python для работы с JSON-объектами (`import python json`). Которая позволяет "упаковать" словарь в JSON.
- Правая часть после `$` — **"impact"** (влияние): число от `0.0` (робот не двигается) до `1.0` (полный контроль). (Присоединяется к уже "запакованному" словарю с помощью обычной конкатинации (сложения) строк)

💡 Пример:

```
{"12": 1.0, "15": -0.5}$1.0
```

 — поднять правое плечо и согнуть правый локоть.

2. Нумерация суставов (важно!)

В коде есть словарь `FROM_NAMES_TO_INDEXES`. Вот ключевые суставы:

Номер	Название сустава	Где находится
12	<code>right_shoulder_pitch_joint</code>	Правое плечо (вверх/вниз)
13	<code>right_shoulder_roll_joint</code>	Правое плечо (вперёд/назад)
15	<code>right_elbow_joint</code>	Правый локоть
16–19	Левые плечо и локоть	Аналогично правой руке
0–2	Правая нога (бедро, колено)	Нижняя часть тела
3–5	Левая нога	
6	<code>torso_joint</code>	Поворот корпуса
9	ИМПАКТ	Не сустав! Управляет "влиянием" пользовательских команд

⚠ В этом узле **пальцы и запястья отключены** (как указано в аннотации), хотя в словаре они есть. Управление идёт только по `active_joints_H1`.

Как управлять роботом? Пошагово

Шаг 1: Запустите узел

В терминале:

Управление руками во время ходьбы (реальный робот):

```
ros2 run low_level_control low_level_control_without_hands_node
# (по умолчанию target_topic_param = "arm_sdk")
```

Полный контроль (реальный робот в development mode или MuJoCo):

```
ros2 run low_level_control low_level_control_without_hands_node \
--ros-args \
-p target_topic_param:="lowcmd" \
```


Шаг 2: Отправьте команду через терминал

Откройте **новый терминал** и отправьте JSON-сообщение:

```
ros2 topic pub /positions_to_unitree std_msgs/msg/String "data: '{\"12\": 0.8, \"15\": -1.0}$1.0'"
```

Что это значит:

- `"12": 0.8` → правое плечо поднимается на 0.8 радиан (~45°)
- `"15": -1.0` → правый локоть сгибается
- `$1.0` → максимальное влияние (робот точно выполнит команду)

 Совет: начинайте с малых углов (0.1–0.5 рад), чтобы не повредить робота!

Шаг 3: Наблюдайте за движением

Робот **плавно** начнёт двигать рукой. Движение замедлено в начале (за счёт `TIME_TO_CHANGE_VELOCITY = 20 сек`), но через 20 секунд станет быстрее.


Если отправить `impact = 0.0`, робот перестанет реагировать на команды и "отдаст контроль" обратно (например, системе балансировки).



Режимы управления Unitree H1: что важно знать



Управление роботом Unitree H1 зависит не только от того, работаете ли вы в симуляции или с реальным роботом, но и от **режима работы самого робота**. Есть два ключевых режима:


1. Обычный режим (`preparation / sport`)

- Робот стоит или ходит под управлением встроенного контроллера (например, с пульта).
- **Вы можете управлять положением ТОЛЬКО верхней частью тела:** руками и торсом.
- Для этого ваши команды **должны публиковаться в топик `arm_sdk`**.
- Ноги остаются под контролем встроенной системы балансировки — вы их не трогаете.
- **Режим разработчика (`development mode`) НЕ требуется.**
-  Подходит для безопасного взаимодействия: робот ходит, а вы управляете жестами рук.

 **Используйте `target_topic_param:=arm_sdk`** — это значение по умолчанию в коде.

2. Режим разработчика (`development mode`)

- Робот **полностью отдаёт контроль внешней системе** (вашему ROS-узлу).
- Вы можете управлять **всеми суставами**, включая ноги.
- В этом режиме робот **обычно подвешен на страховке** — ходить самостоятельно он не будет!
- Команды должны публиковаться в топик `lowcmd`.
-  **Обязательно включите `development mode`** на роботе (см. [Нашу документацию](#)).
-  Без этого режима публикация в `lowcmd` будет проигнорирована.

 **Используйте `target_topic_param:=lowcmd`**, если работаете в этом режиме.

3. Симуляция в MuJoCo

- В симуляторе **нет "обычного режима"** — Вы всегда управляете всеми поторами робота напрямую.
- Поэтому **работает ТОЛЬКО топик `lowcmd`**.
- Топик `arm_sdk` в MuJoCo **не поддерживается** и не будет иметь эффекта.
- ✓ Запускайте узел с `target_topic_param:=lowcmd`.

Итоговая сводка

Сценарий	Топик для публикации	Требуется development mode?	Управление
Реальный робот, ходьба + управление руками	<code>arm_sdk</code>	✗ Нет	Только руки + торс
Реальный робот, полный контроль (подвешен)	<code>lowcmd</code>	✓ Да	Все суставы
Симуляция (MuJoCo)	<code>lowcmd</code>	✗ (не применимо)	Все суставы

Безопасность

- Код **автоматически ограничивает углы** суставов (через `limits` из `hl_info_library`), так что вы не сможете выйти за физические пределы.
- При завершении (`Ctrl+C`) узел **плавно снижает `impact` до 0**, чтобы робот не "уронил" руки.

? Частые вопросы новичков

Q: Где взять текущие значения углов?

A: Используйте `ros2 topic echo /lowstate`, чтобы посмотреть текущие углы суставов.

Q: Почему робот не двигается?

- Проверьте, что `impact > 0`.
- Убедитесь, что робот включен и подключён.
- Проверьте, что номера суставов правильные (только из `active_joints_H1`).

Q: Можно ли управлять ногами?

Да! Например: `{"0": 0.2, "1": -0.3, "2": 0.5}$0.5` — немного пошевелить правой ногой.

Итог

Этот код — **мост между вашими командами и железом робота**.
Вы говорите: «рука в позицию X», а узел **безопасно и плавно** это выполняет.

Главное — помните:

- Используйте **номера суставов**, а не названия.
- Всегда указывайте **impact** (иначе ничего не произойдёт).
- Начинайте с **малых углов и низкого impact**.

Удачи в управлении Unitree H1! 🤖