



Кибердром 2024



Всероссийский проект «Кадры для цифровой промышленности. Создание законченных проектно-конструкторских решений в режиме соревнований «Кибердром.2023»

Название предприятия, учреждения

Модуль: ВВЕДЕНИЕ В ДЕЯТЕЛЬНОСТЬ «ЭКСПЛУАТАЦИЯ БЕСПИЛОТНЫХ АВИАЦИОННЫХ СИСТЕМ»

Создание полетных заданий, часть 1

Преподаватель:

Рычагов Егор

Программист-стажер

ГРУППА КОМПАНИЙ

GEOSCAN



ФОНД СОДЕЙСТВИЯ РАЗВИТИЮ
ВОЕННОГО ОБРАЗОВАНИЯ



omegabot



ГБПОУ г. Москвы
«Колледж полиции»





1. Существующие ограничения для полета коптера
2. Особенности написания полетного задания с помощью `piosdk`
3. Функции, используемые для полетного задания
4. Выполнение полета несколькими коптерами



Научиться писать полётные задания
с использованием библиотеки `piosdk` для одного и
более коптеров

Формируемые компетенции (результат освоения)



Кибердром 2024



МИНПРОМТОРГ
РОССИИ

- Методы обучения: ЛЕКЦИЯ, практическое задание
- Технические средства обучения: КОМПЬЮТЕР
- Программное обеспечение: Microsoft Windows 10, VS code (рекомендуем PyCharm), PioneerSim

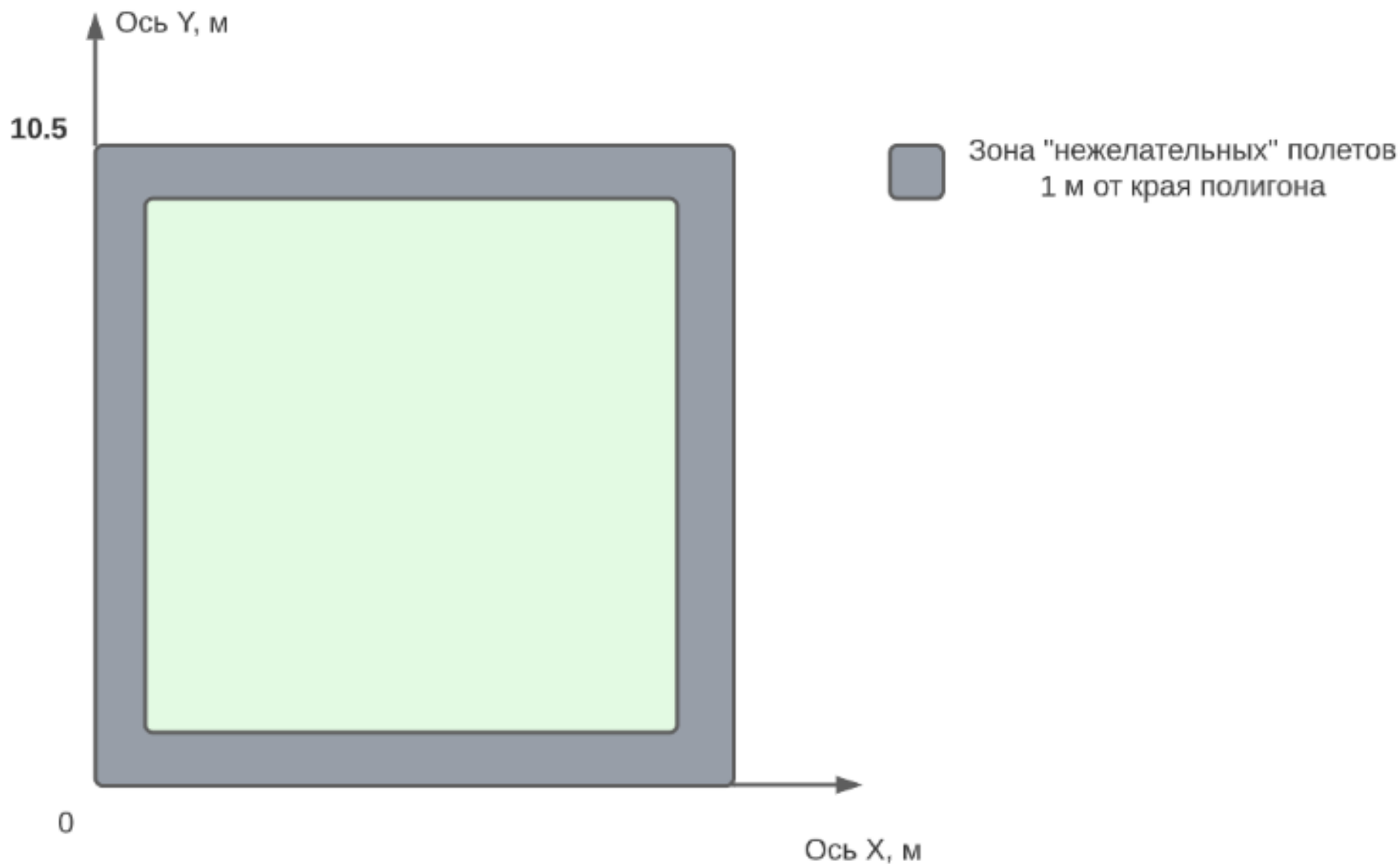
Ограничения полигона



Кибердром 2024



МИНПРОМТОРГ
РОССИИ



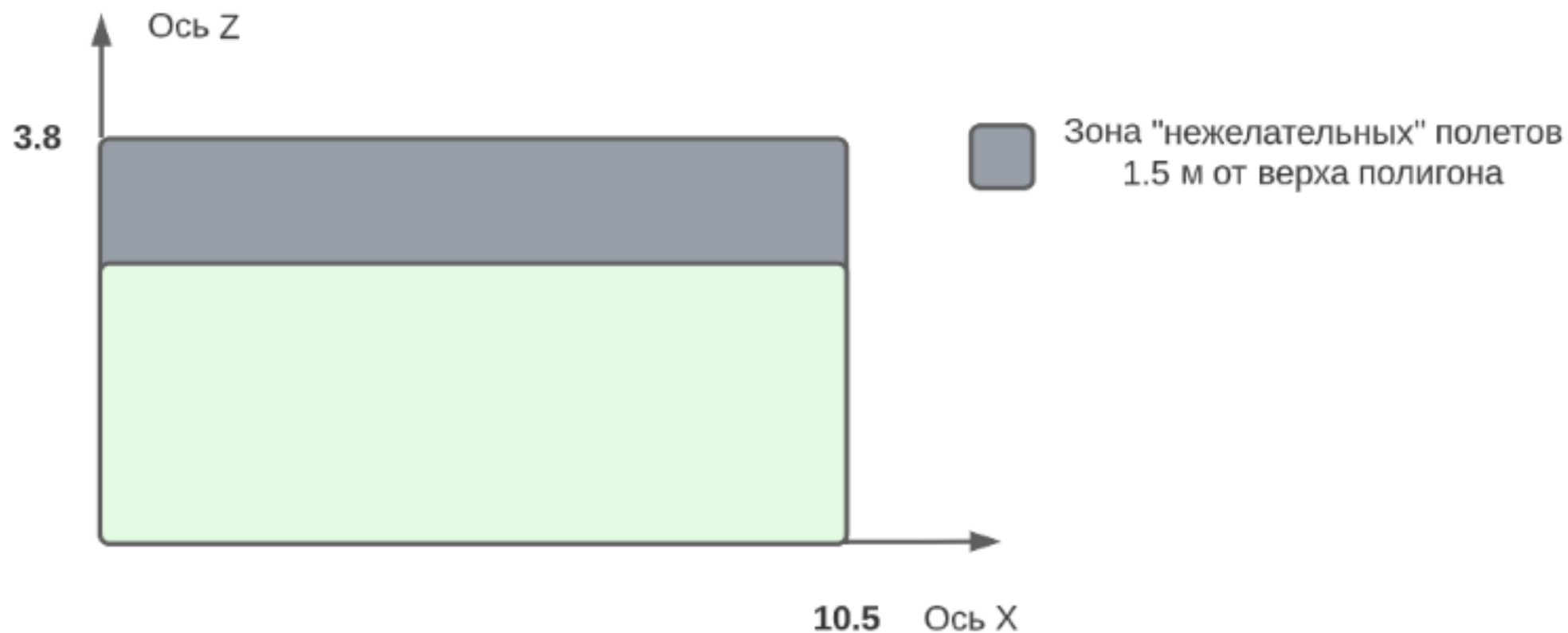
Ограничения полигона



Кибердром 2024



МИНПРОМТОРГ
РОССИИ



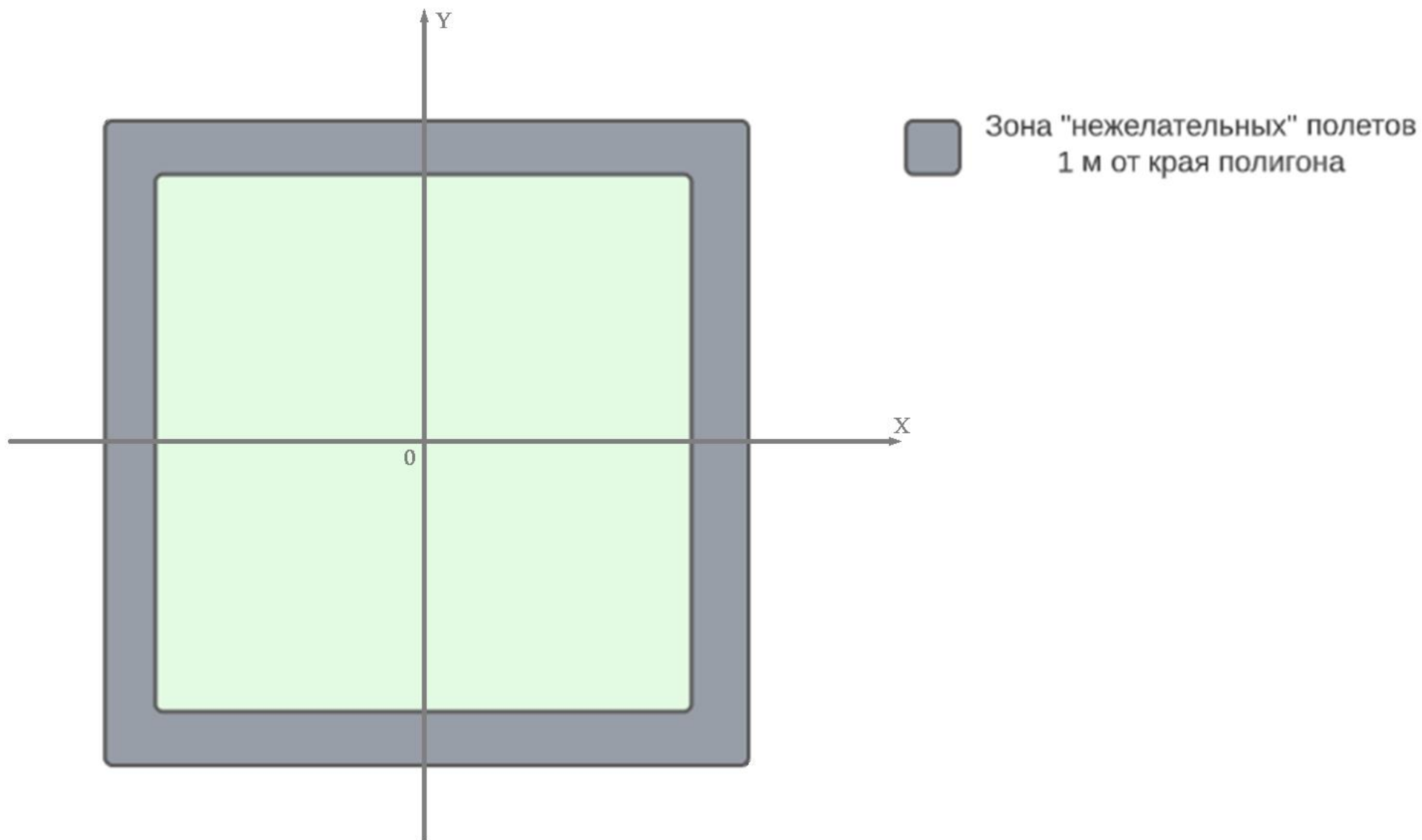
Координатная плоскость



Кибердром 2024



МИНПРОМТОРГ
РОССИИ



В некоторых случаях функции piosdk возвращают None:

```
def get_local_position_lps(self, blocking=False):
    """ Возвращает данные от системы навигации LPS """
    position = self.__mavlink_socket.recv_match(type='LOCAL_POSITION_NED', blocking=blocking,
                                                timeout=self.__ack_timeout)

    if not position:
        return None

    if position.get_type() == "BAD_DATA":
        if mavutil.all_printable(position.data):
            sys.stdout.write(position.data)
            sys.stdout.flush()
    else:
        if position._header.srcComponent == 26:
            return [position.x, position.y, position.z]
        else:
            return None
```


В некоторых случаях функции piosdk возвращают None:

```
while True:
    position = drone.get_local_position_lps()
    if position is not None:
        print(position[0], position[1], position[2])
```

Альтернативный способ - обработать исключение. Однако в большинстве случаев проверки хватает

```
while True:
    position = drone.get_local_position_lps()
    try:
        print(position[0], position[1], position[2])
    except TypeError:
        print("Получено значение None")
```

Особенности полета в точку



Кибердром 2024

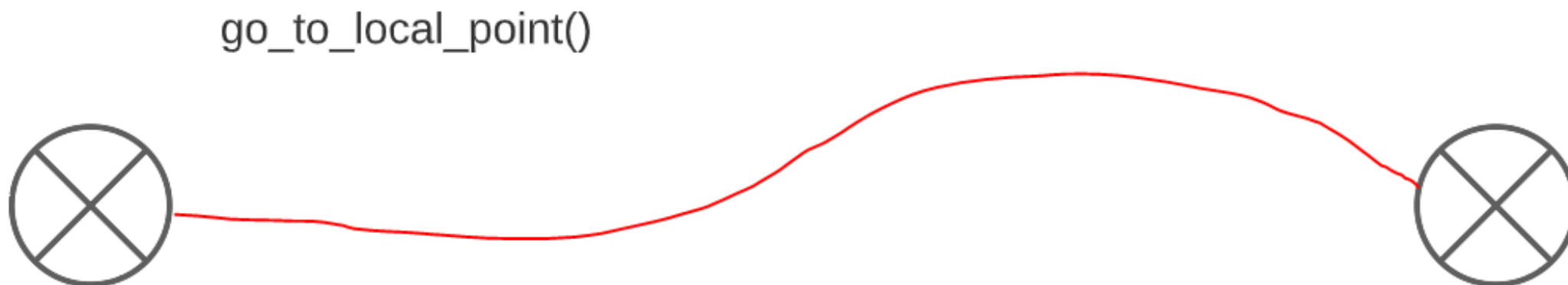


МИНПРОМТОРГ
РОССИИ

В симуляторе траектория выглядит так:

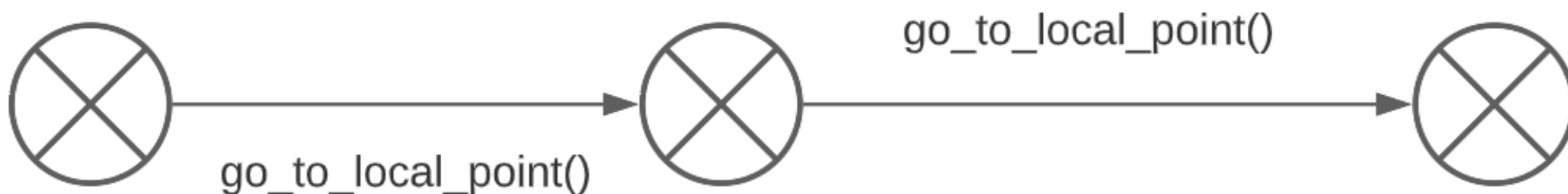


В реальности скорее так:



Что делать?

Траектория движения станет более прямой, если добавить промежуточную точку:



Особенности посадки

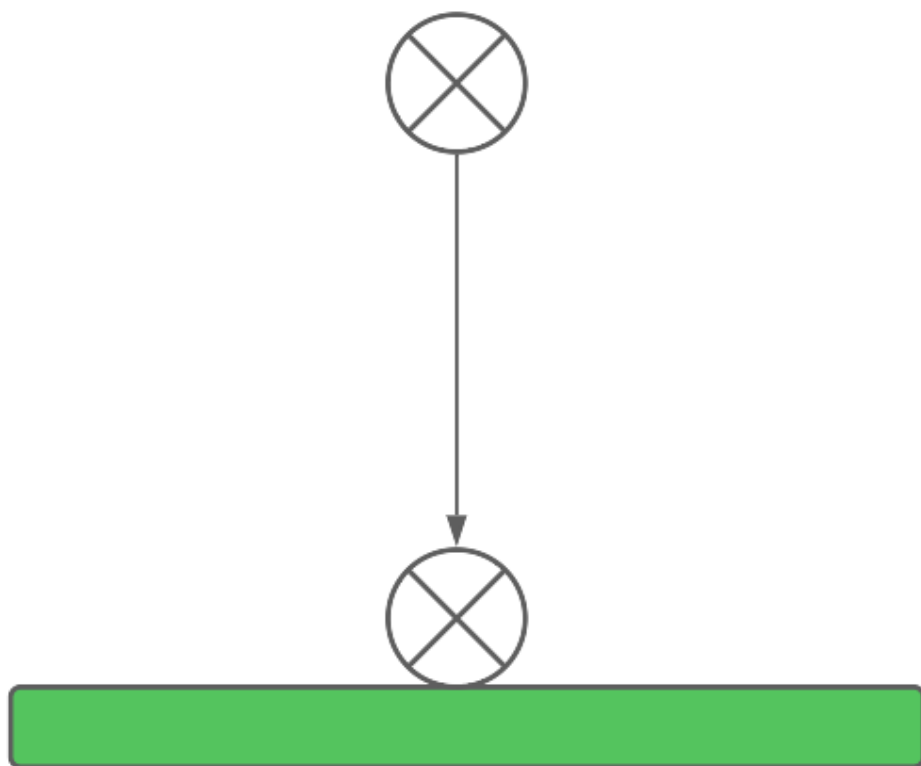


Кибердром 2024

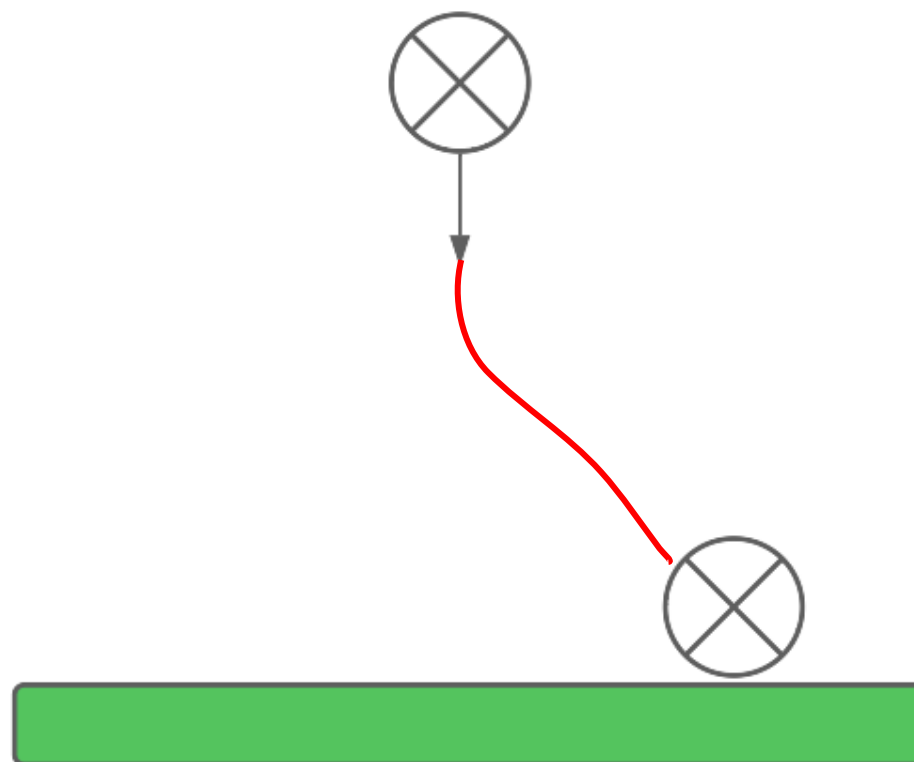


МИНПРОМТОРГ
РОССИИ

Ожидание



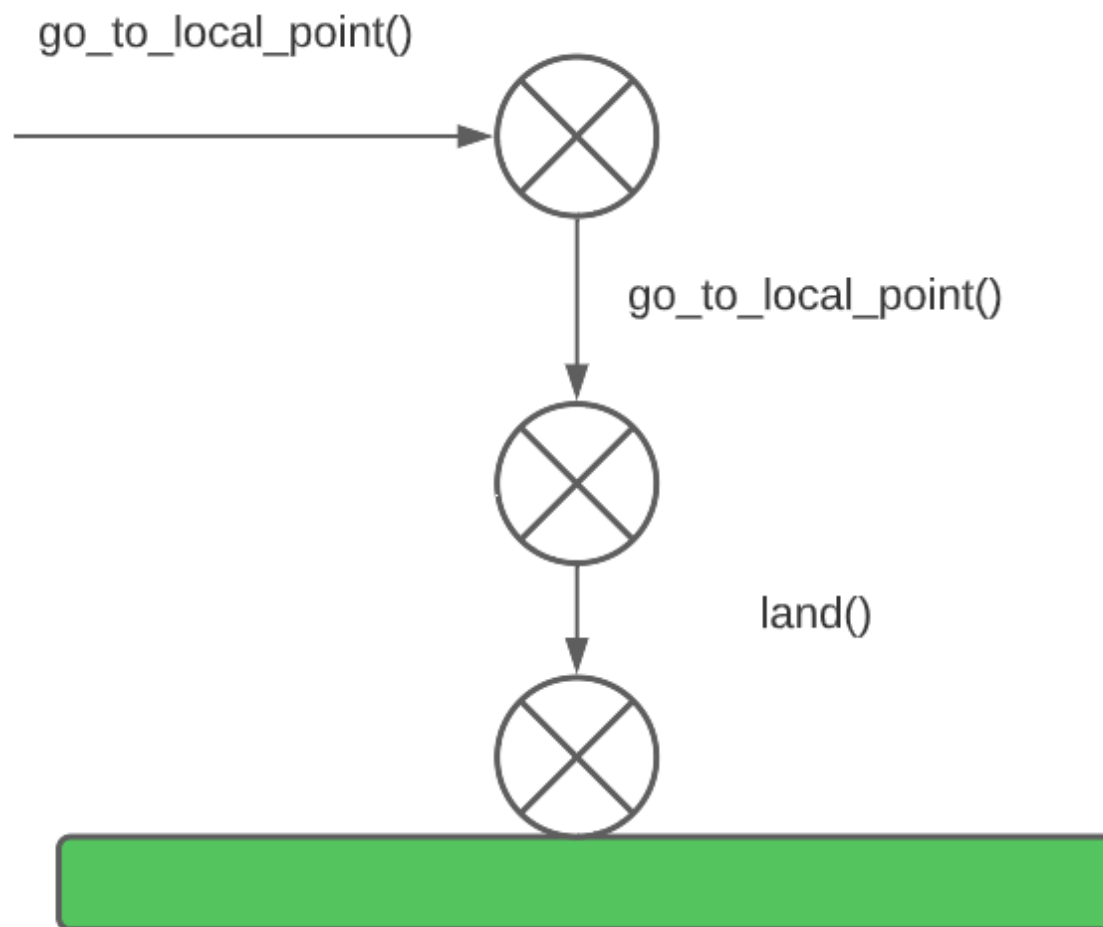
Реальность



Особенности посадки



Добавим промежуточную точку, чтобы снизиться. Бинго!



Основные функции



Кибердром 2024



МИНПРОМТОРГ
РОССИИ

<code>arm()</code>	Завести моторы
<code>disarm()</code>	Заглушить моторы
<code>takeoff()</code>	Взлет
<code>land()</code>	Посадка
<code>go_to_local_point(x, y, z)</code>	Полёт в точку с указанными координатами
<code>point_reached() -> bool</code>	Достигнута ли предыдущая заданная точка
<code>get_local_position_lps() -> [x, y, z]</code>	Получить текущие координаты коптера
<code>get_piro_sensor_data() -> temp</code>	Получить температуру с пиросенсора
<code>get_qr_reader_data() -> qr_data</code>	Получить данные о QR метке

- https://github.com/GeoScan-Pioneer/pioneer_sdk.git

Пример полетного задания



Кибердром 2024



МИНПРОМТОРГ
РОССИИ

```
from piosdk import Pioneer

drone = Pioneer()

drone.arm()
drone.takeoff()
drone.go_to_local_point(0, 1)
while not drone.point_reached():
    pass
drone.land()
drone.disarm()
```

Важно!

Проверять статус
с помощью point_reached()

Задание для нескольких коптеров



Если написать программу таким образом, она будет блокироваться во время движения одного коптера, и другой будет простаивать.

```
drone.go_to_local_point(0, 1)
drone2.go_to_local_point(1, 0)
while not drone.point_reached():
    pass
while not drone2.point_reached():
    pass
```

```
def drone_task():  
    drone.arm()  
    drone.takeoff()  
    drone.go_to_local_point(0, 1)  
    while not drone.point_reached():  
        pass  
    drone.land()  
    drone.disarm()
```

Полетное задание для каждого из коптеров можно вынести в функцию и запустить эту функцию отдельным потоком.

```
thread1 = threading.Thread(target=drone_task_1)  
thread2 = threading.Thread(target=drone_task_2)  
thread1.start()  
thread2.start()
```



1. В каких зонах полигона лучше не летать?
2. Как добиться более прямой траектории полета в точку?
3. Почему при вызове `land()` в точке с координатами (0, 0, 3) коптер может приземлиться в точку (1, 1, 0)?
4. Зачем использовать несколько потоков для нескольких коптеров?

Домашнее задание

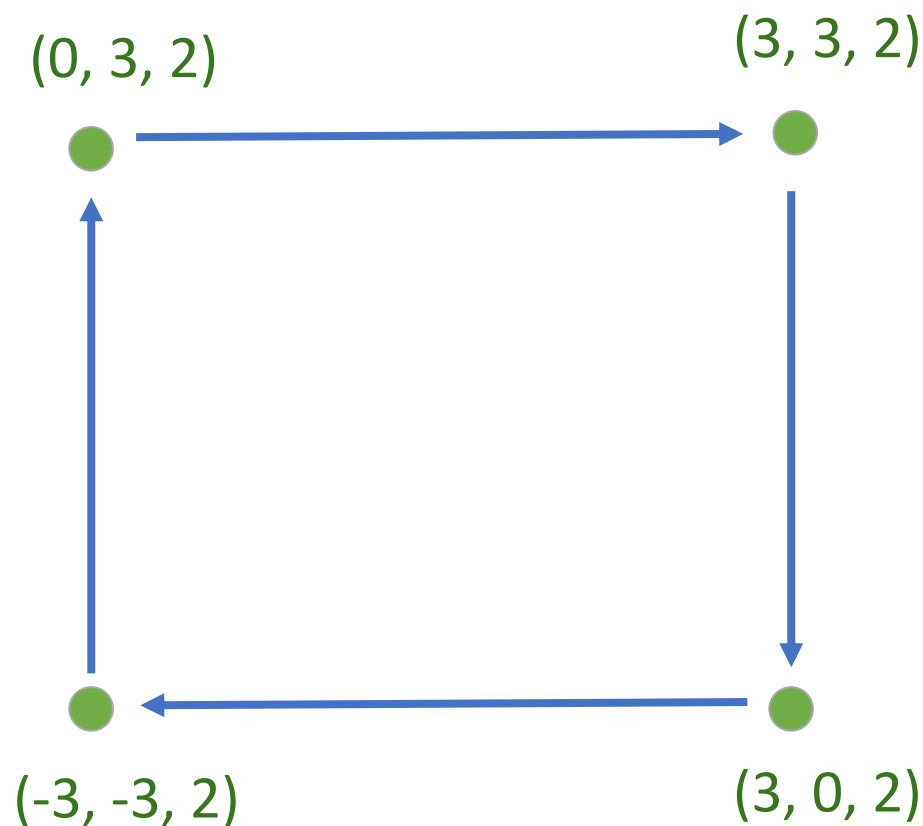


Кибердром 2024



МИНПРОМТОРГ
РОССИИ

1. Написать и протестировать полетное задание, включающее в себя достижение точки с координатами $(3, 3, 1)$ и посадку. Использовать промежуточные точки.
2. Проверить, что произойдет, если написать команды последовательно, не дожидаясь `point_reached()`
3. Реализовать полетное задание для 4 коптеров согласно изображенной схеме



Дополнительные материалы



Кибердром 2024



МИНПРОМТОРГ
РОССИИ

1. Исходный код библиотеки pioneer sdk:
https://github.com/GeoScan-Pioneer/pioneer_sdk.git