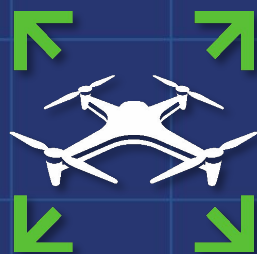




Всероссийский конкурс
«КАДРЫ ДЛЯ ЦИФРОВОЙ ПРОМЫШЛЕННОСТИ. СОЗДАНИЕ ЗАКОНЧЕННЫХ
ПРОЕКТНО-КОНСТРУКТОРСКИХ РЕШЕНИЙ В РЕЖИМЕ СОРЕВНОВАНИЙ
«КИБЕРДРОМ»



Кибердром 2022

Название предприятия, учреждения

Модуль: ВВЕДЕНИЕ В ДЕЯТЕЛЬНОСТЬ

«ЭКСПЛУАТАЦИЯ БЕСПИЛОТНЫХ АВИАЦИОННЫХ СИСТЕМ»

Название темы: Основы программирования БЛА.

Дополнительные модули. Взаимодействие БЛА и модулей.

Обзорная лекция

Преподаватель: Самарин Алексей Сергеевич, инженер-
программист



Москва, 2022



Изучаемые вопросы



Кибердром²⁰²²

МИНПРОМТОРГ
РОССИИ

1. Дополнительные модули БЛА
2. Взаимодействие БЛА и модулей
3. Программирование БЛА

Цель занятия



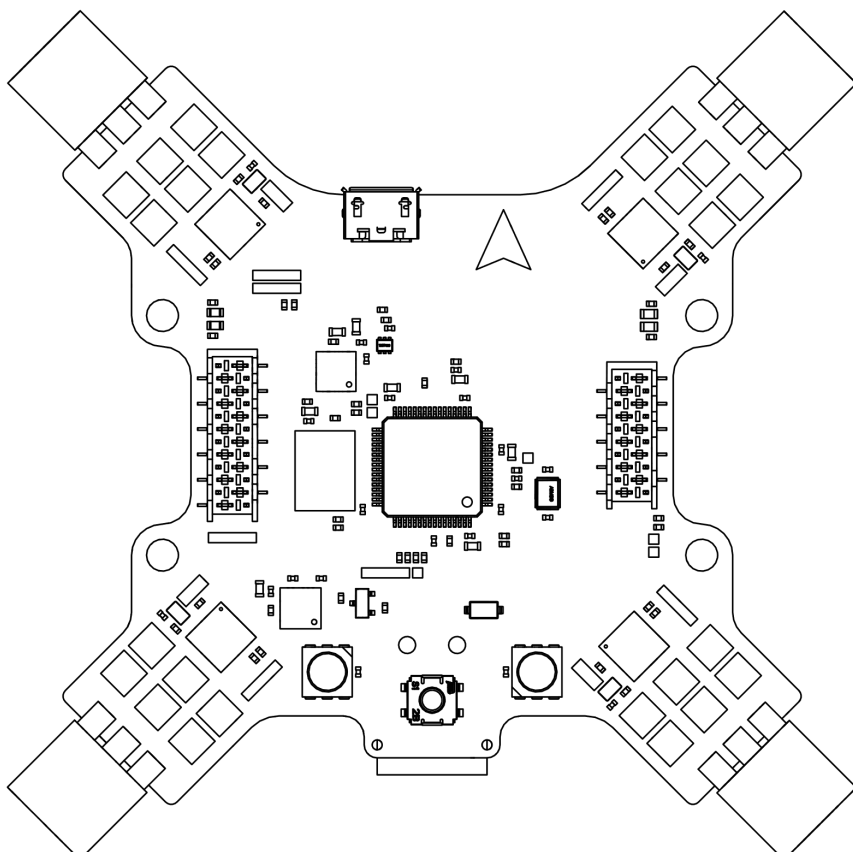
1. Рассмотреть модули
2. Познакомиться с основами программирования на LUA;
3. Познакомиться с основами блочного программирования;
4. Познакомиться с основами программирования на Python.

Основные модули Платформа



Кибердром 2022

МИНПРОМТОРГ
РОССИИ



Процессор: ARM® Cortex® M4, 168 МГц, 32-Bit

*Системы позиционирования:
акселерометр, гироскоп, барометр*

Диапазон входных напряжений: 2S, 3S

*Возможность загрузки полетных
скриптов прямо на плату*

4 RGB светодиода

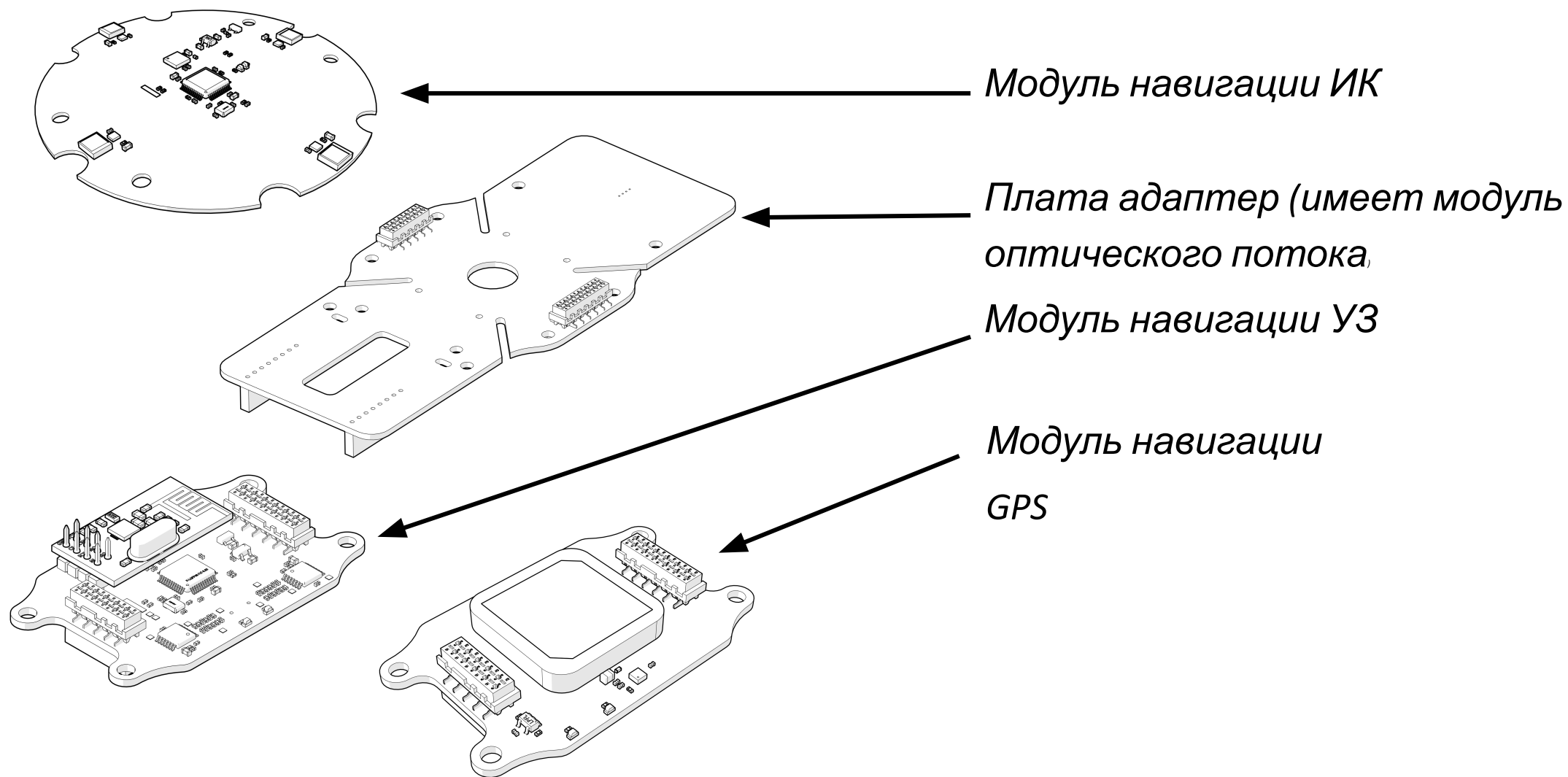
Основные модули Пилотажа

Модуль позиционирования



Кибердром 2022

МИНПРОМТОРГ
РОССИИ



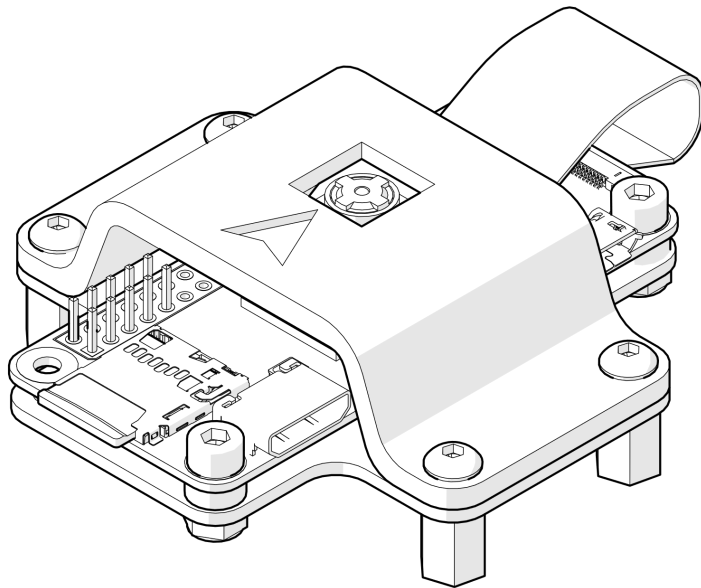
Основные модули

Видеомодуль Raspberry pi zero w



Кибердром 2022

МИНПРОМТОРГ
РОССИИ



- *Центральный процессор: CPU ARM1176JZ-F с тактовой частотой 1 ГГц*
- *Графический процессор: GPU VideoCore IV с тактовой частотой 400 МГц*
- *Оперативная память: 512 МБ*
- *Способ связи: Wi-Fi, Bluetooth 4.1*
- *Разъем видеочамеры: Camera Serial Interface*
- *Порты ввода-вывода: 40*

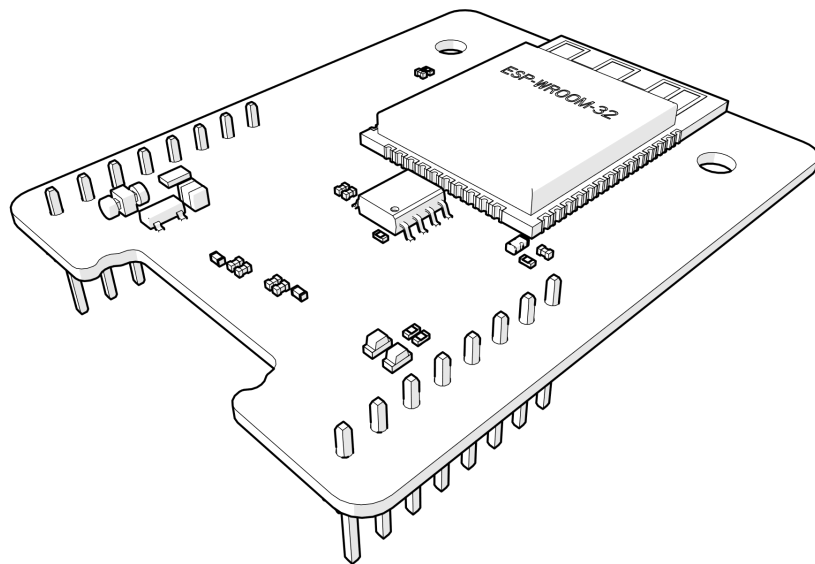
Основные модули Пионера

Модуль ESP32



Кибердром 2022

МИНПРОМТОРГ
РОССИИ



Модуль позволяет соединяться с Пионером по Wi-Fi и писать управляющие программы на языке Python

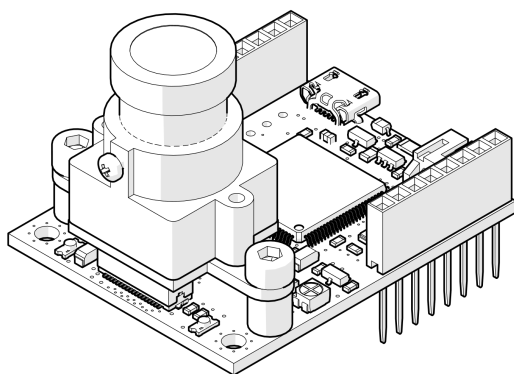
Основные модули Пионера

Устанавливаемые камеры



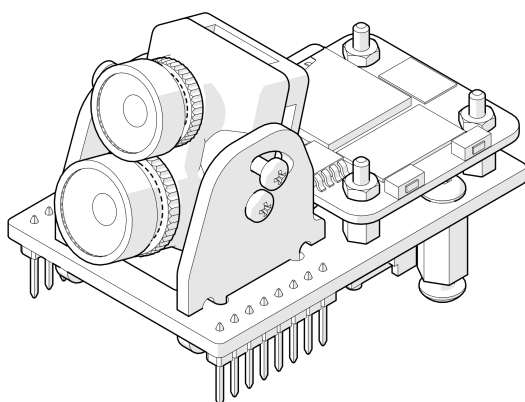
Кибердром 2022

МИНПРОМТОРГ
РОССИИ



Программируемая камера OpenMV

- *Протокол соединения - UART*
- *Макс. разрешение - 640×480: 75 fps — 8 бит.*
- *Возможность программирования*



Камера RunCam Hybrid для FPV полетов

- *Разрешение основной камеры - 8 МП*
- *Разрешение аналоговой камеры - 1,3 МП*
- *Разрешение видео основной камеры - 4K: 30 fps; 2,7K: 60 fps; 1080p: 120 fps*

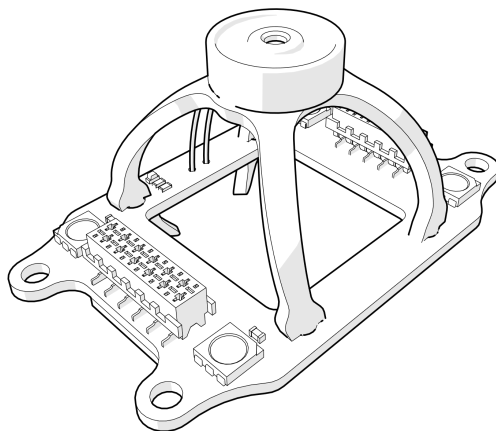
Основные модули Пионера

Устанавливаемые камеры



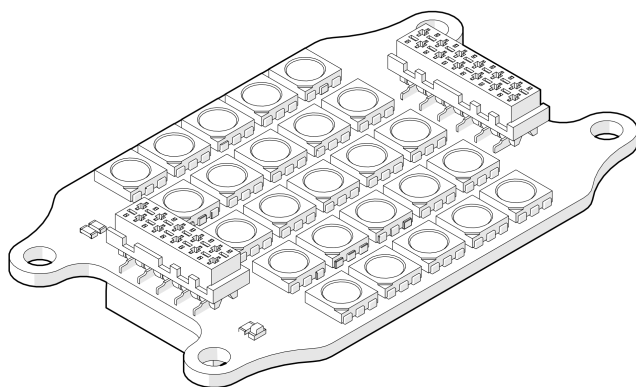
Кибердром 2022

МИНПРОМТОРГ
РОССИИ



Модуль захвата грузов

- *Протокол соединения - UART*
- *Макс. разрешение - 640×480: 75 fps — 8 бит.*
- *Возможность программирования*



LED модуль

- *Разрешение основной камеры - 8 МП*
- *Разрешение аналоговой камеры - 1,3 МП*
- *Разрешение видео основной камеры - 4K: 30 fps; 2,7K: 60 fps; 1080p: 120 fps*

Программирование

Введение



Кибердром 2022

МИНПРОМТОРГ
РОССИИ

Для выполнения автономных полетов управляющую программу можно написать на языке LUA, Python, а также с помощью блочного программирования.

Программа может включать в себя:

- взаимодействие с модулями, представленными ранее;*
- управление полетом: взлет, посадка, полет в точку (LPS, OPT, GPS).*



**PIONEER
STATION**



PyCharm



**GEOSCAN
JUMP**



**TRIK
STUDIO**

Программирование на Структура программы



Кибердром 2022

МИНПРОМТОРГ
РОССИИ

Объявление периферии (пульт,
магнит, светодиоды)

Объявление исполняемых функций

Объявление обработчика событий АП

Объявления таймеров



**PIONEER
STATION**

Программирование на Python

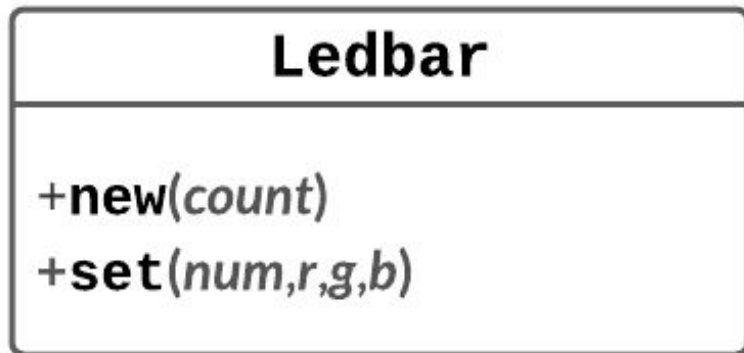
Работа со светодиодами



Кибердром 2022

МИНПРОМТОРГ
РОССИИ

Класс для работы со светодиодами:



Функции класса:

new(count) – создать новый Ledbar с заданным количеством светодиодов

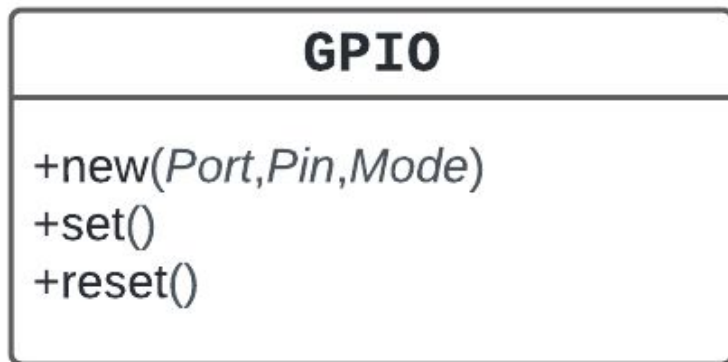
set(num, r, g, b) – установить цвет на заданный светодиод

Программирование на Python

Работа GPIO



Класс для работы с GPIO:



Функции класса:

new(port, pin, mode) - Создать GPIO на порте

set - установить значение в 1;

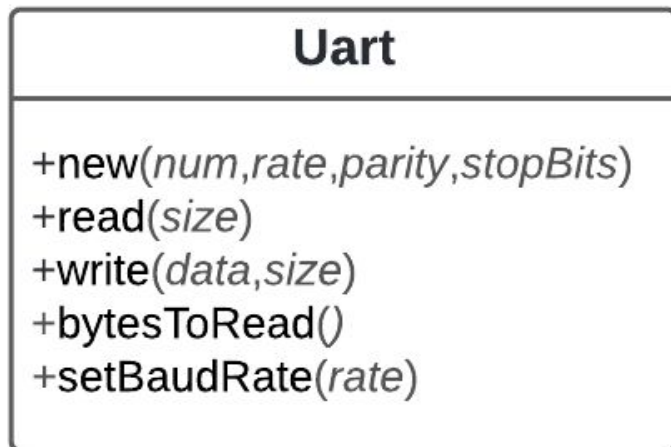
reset - установить значение в 0.

Программирование на C++

Работа с UART



Класс для работы с UART:



Функции класса:

new(num, rate, parity, stopBits) - создать Uart на порте с настройками.

read(size) - прочитать size байт.

write(data, size) - записать данные (data) длиной (size).

bytesToRead() - количество данных доступных для чтения.

setBaudRate(rate) - установить скорость rate.

Программирование на Python

Работа SPI



Класс для работы с SPI:

Spi
+new(<i>num,rate,seq,mode</i>)
+read(<i>size</i>)
+write(<i>data,size</i>)
+exchange(<i>data,size</i>)

Функции класса:

new(num, rate, seq, mode) - создать Spi на порте с настройками.

read(size) - прочитать size байт.

write(data, size) - записать данные (data) длиной (size).

exchange(data, size) - Записать данные (data) длиной (size) и прочитать size.

Программирование на Python

Работа с таймерами



Класс для работы с таймерами:

Timer
+new(<i>sec, func</i>)
+start()
+stop()
+callAt(<i>local_time, func</i>)
+callLater(<i>delay, func</i>)
+callAtGlobal(<i>global_time, func</i>)

Функции класса:

new(sec, func) - создать новый Timer.

start() - запускает таймер.

stop() - останавливает таймер.

callAt(local_time, func) - создает и запускает новый Timer с функцией, которая будет вызвана один раз.

callLater(delay, func) - создает и запускает новый Timer с функцией, которая будет вызвана один раз.

callAtGlobal(global_time, func) - создает и запускает новый Timer с функцией, которая будет вызвана один раз.

Программирование на Python

Работа с данными АП



Класс для работы с АП:

Sensors
+lpsPosition() : x, y, z
+lpsVelocity(): vx, vy, vz
+lpsYaw(): yaw
+orientation: roll, pitch, azimuth
+altitude(): z
+range(): z
+accel(): ax, ay, az
+ gyro(): gx, gy, gz
+ rc(): channel1 .. channel8

Функции класса:

lpsPosition() - возвращает значение позиции при использовании LPS. Результат: x, y, z

lpsVelocity() - возвращает значение скоростей при использовании LPS. Результат: vx, vy, vz

lpsYaw() - получить значение угла. Результат: yaw

orientation() - данные положения. Результат: roll, pitch, azimuth

altitude() - данные высоты по барометру. Результат: высота в метрах

range() - данные с датчиков расстояния. Результат: возвращает значения с датчика расстояния.

accel() - данные с акселерометра. Результат: ax, ay, az

gyro() - данные с гироскопа. Результат: gx, gy, gz

rc() - данные с пульта управления. Результат: channel1, channel2, channel3, channel4, channel5, channel6, channel7, channel8.

Программирование на Получение событий АП



Кибердром 2022

МИНПРОМТОРГ
РОССИИ

Событие	Описание
ENGINES_STARTED	Двигатели запущены
COPTER_LANDED	Коптер совершил посадку
TAKEOFF_COMPLETE	Коптер достиг высоты взлета
POINT_REACHED	Коптер достиг точки
POINT_DECELERATION	Коптер начал тормозить при подлёте к точке
LOW_VOLTAGE1	Низкое напряжение аккумулятора, для возвращения домой
LOW_VOLTAGE2	Низкое напряжение аккумулятора, переходит в режим посадки
SYNC_START	Получен сигнал синхронного старта от системы навигации
SHOCK	Столкновение или слишком сильные вибрации
CONTROL_FAIL	Угол наклона коптера превысил допустимый
ENGINE_FAIL	Отказ двигателя

Программирование на Получение событий АП



Данную функцию необходимо объявлять каждый раз самому и прописывать ее функционал. Функция каждый раз будет вызываться сама и **не нужно ее вызвать самим!**

```
function callback(event)
    if(event == Ev.TAKEOFF_COMPLETE) then
        nextPoint()
    end
    if(event == Ev.POINT_REACHED) then
        nextPoint()
    end
end
```

Программирование на C++

Управление событиями АП



Кибердром 2022

МИНПРОМТОРГ
РОССИИ

Класс для работы с АП:

ap
<pre>+push(Event) +goToPoint(latitude,longitude,altitude) +goToLocalPoint(x,y,z,time) +updateYaw(angle)</pre>

Функции класса:

ap.push(Event) - добавить событие автопилоту.

ap.goToPoint(latitude, longitude, altitude) - для полета с использованием GPS.

ap.goToLocalPoint(x, y, z, time) - для полёта с использованием локальной системы координат.

ap.updateYaw(angle) - установить рыскание.

Программирование на Управление событий АП



Кибердром 2022

МИНПРОМТОРГ
РОССИИ

Таблица отправляемых событий:

Событие	Описание
MCE_PREFLIGHT	Запустить двигатели и провести подготовку
ENGINES_DISARM	Отключить двигатели
MCE_LANDING	Отправить на посадку
MCE_TAKEOFF	Отправить на взлет

Программирование на Управление событий АП



Пример отправки событий АП:

```
-- функция обработки событий, автоматически вызывается автопилотом
function callback(event)
  -- Если коптер взлетел, вызываем функцию полета в точку
  if(event == Ev.TAKEOFF_COMPLETE) then
    nextPoint()
  end
end

-- Предстартовая подготовка
ap.push(Ev.MCE_PREFLIGHT)

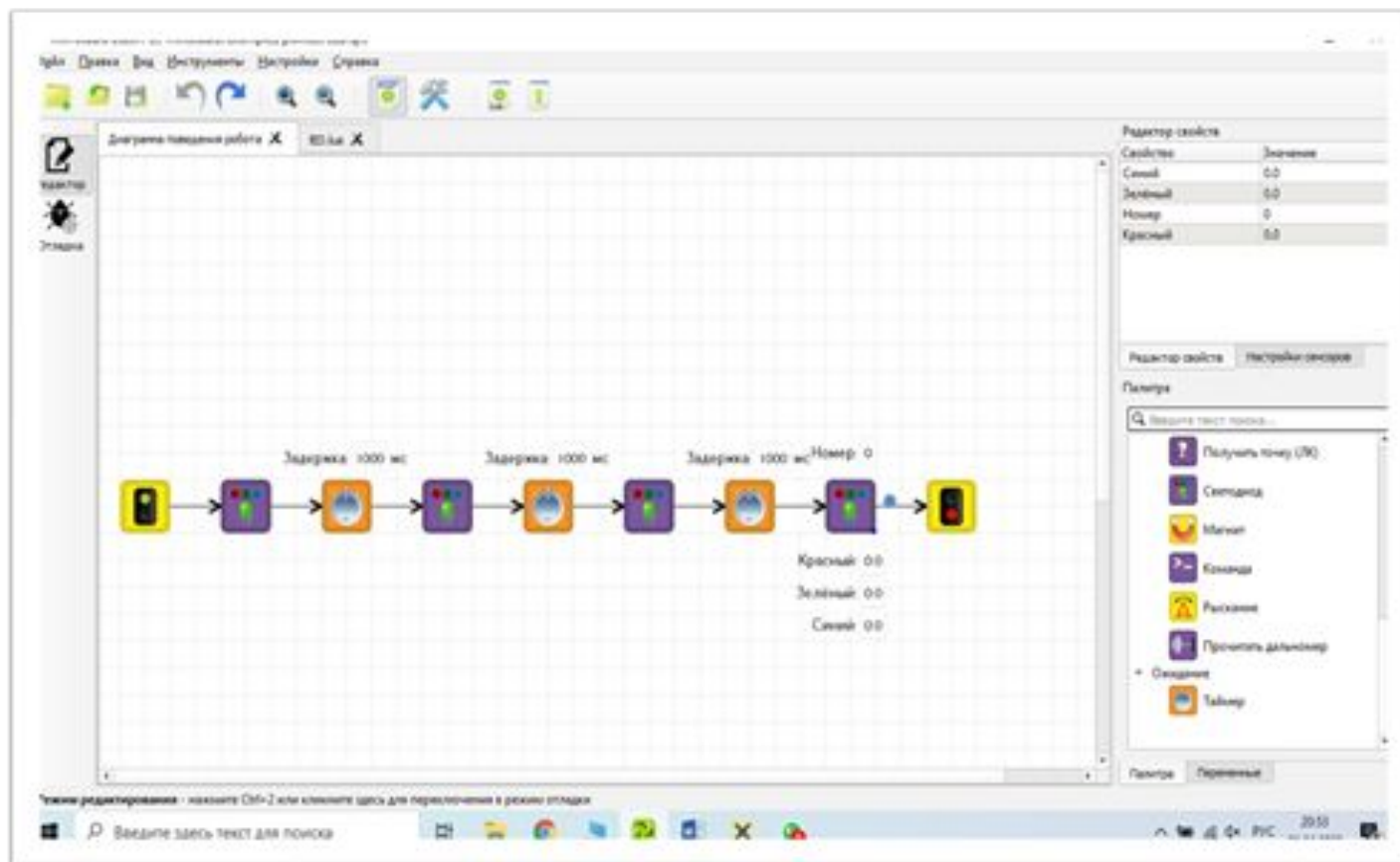
-- Таймер, через 2 секунды вызывающий функцию взлета
Timer.callLater(2, function() ap.push(Ev.MCE_TAKEOFF) end)
```

Блочное введение в программирование



Кибердром 2022

МИНПРОМТОРГ
РОССИИ







*Комбинация нужных
блоков позволяет
создать управляющую
программу на языке
LUA.*

Блочное программирование



Кибердром 2022

МИНПРОМТОРГ
РОССИИ






	<p>Условие - позволяет создать два сценария действия «Пионера» в зависимости от заданного логического условия. У блока должны быть две исходящие связи, в одной из которых в редакторе свойств должно быть назначено значение параметра «условие» (истина или ложь).</p>
	<p>Конец условия - обозначает слияние двух веток условного оператора. Никаких действий не выполняет, но полезен для обеспечения структурности программы.</p>
	<p>Инициализация переменной - позволяет объявить новую переменную. В редакторе свойств или прямо на диаграмме задается имя переменной и ее значение.</p>
	<p>Случайное число - присваивает выбранной переменной случайное значение из выбранного диапазона.</p>

Блочное программирование



Кибердром 2022

МИНПРОМТОРГ
РОССИИ






	Комментарий - позволяет включить в программу текстовые пояснения, упрощающие понимание структуры участка или конкретного блока.
	Взлет, Посадка - команды начала и завершения полета.
	Таймер - задает время ожидания перед выполнением следующего блока программы в миллисекундах.
	Лететь на точку - указывает точку назначения в локальных координатах.
	Светодиод - управляет работой светодиодов на плате «Пионера».

Блочное программирование



Кибердром 2022

МИНПРОМТОРГ
РОССИИ

	Получить точку (ЛК) - возвращает текущую позицию в локальной системе координат.
	Магнит - управляет работой модуля захвата груза. Чтобы включить магнит, поставьте галочку в чек боксе значения свойств блока.
	Команда - позволяет выполнить написанную в свойствах команду. Для написания команды используется язык Lua. Для выполнения команды поставьте галочку в чек боксе.
	Прочитать дальномер - считывает информацию с дальномера.
	Рыскание - управляет изменением направления полета «Пионера» вокруг вертикальной оси.

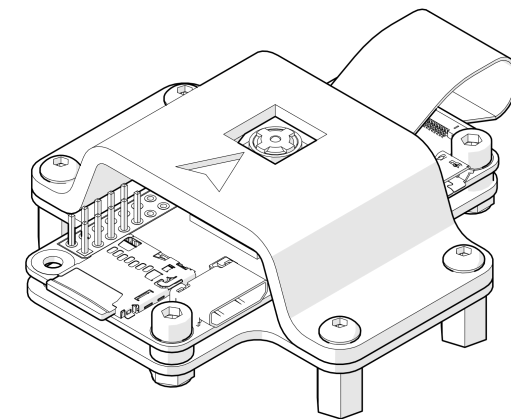
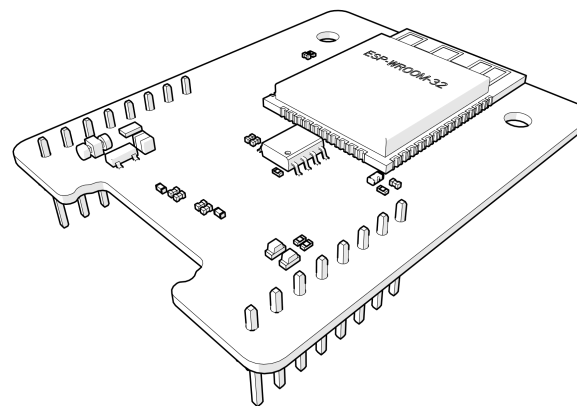
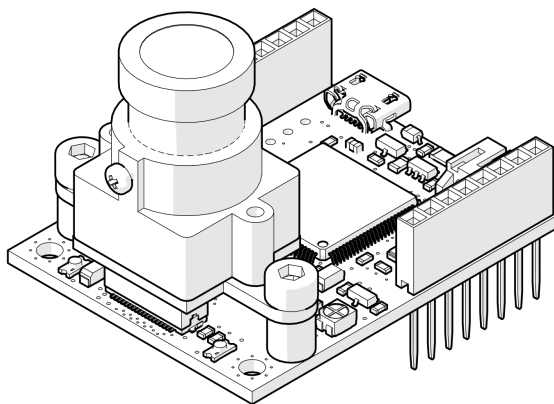
Программирование на Python

Введение



Для написания управляющих программ на языке Python есть 3 способа:

- Подключить модуль камеры OpenMV.
- Подключить модуль ESP32. Для написания программы используется библиотека PioneerSDK.
- Подключить модуль Raspberry pizero. Управляющая программа в таком случае может быть написана как на ПК, с помощью PioneerSDK, так и на модуле .



Программирование на Python

PioneerSDK



Кибердром 2022

МИНПРОМТОРГ
РОССИИ

PioneerSDK это api, использующее протокол MavLink.

Скачать можно по ссылке: https://github.com/geoscan/pioneer_sdk

Плюсы:

PioneerSDK и протокол MavLink имеют открытый исходный код, что позволяет при необходимости дорабатывать данное программное обеспечение самим пользователям;

Одна и та же программа может работать на Pioneer и Pioneer Mini;

Программирование на языке Python.

Программирование на Python

Установка PioneerSDK



Кибердром 2022

МИНПРОМТОРГ
РОССИИ

1. Установить Python на ПК;
2. Установить PyCharm;
3. Установить PioneerSDK



PyCharm

Программирование на Python

Установка Python



Кибердром 2022

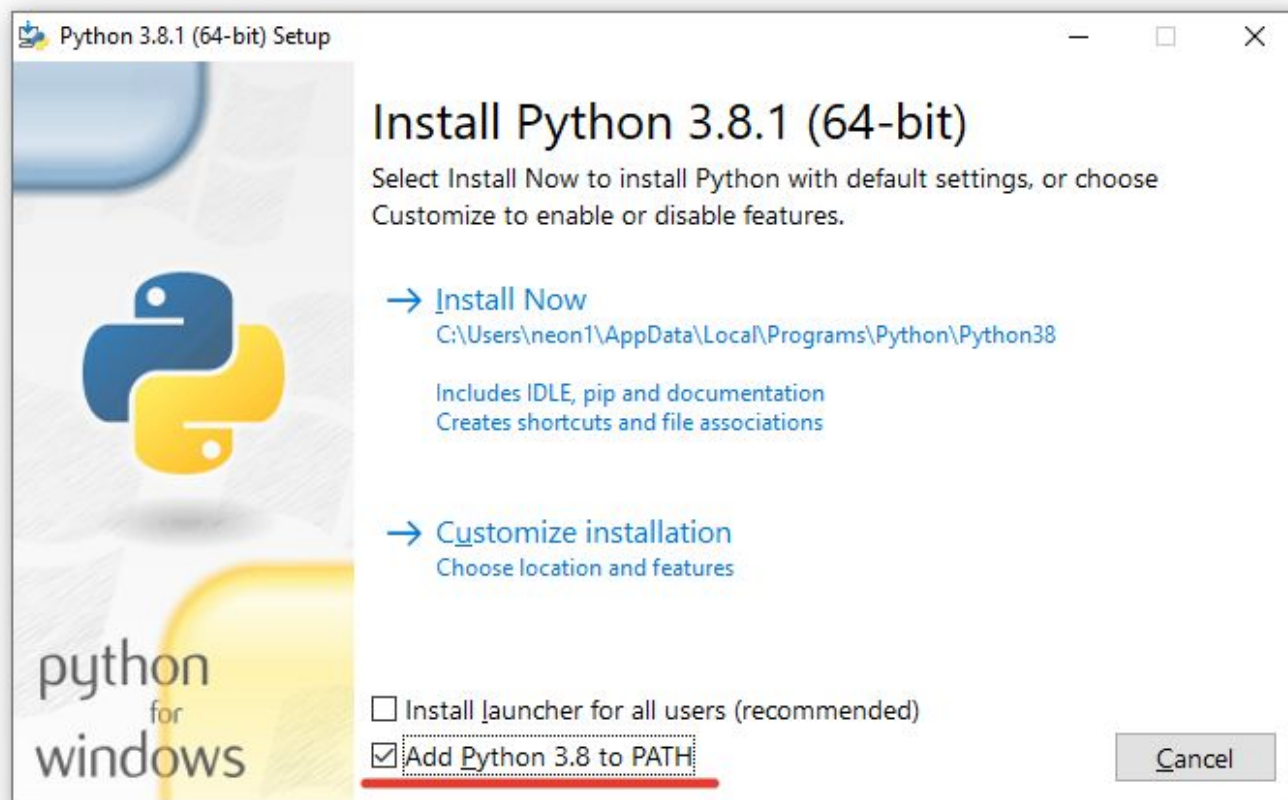
МИНПРОМТОРГ
РОССИИ



Ссылка на скачивание:

<https://www.python.org/>

Обратите внимание,
что при установке
необходимо выбрать
следующий пункт:



Программирование на Python

Установка PyCharm



Кибердром 2022

МИНПРОМТОРГ
РОССИИ



PyCharm

Ссылка на скачивание:

<https://www.jetbrains.com/ru-ru/pycharm/downloadWS>

Обратите
внимание, что
бесплатной
является PyCharm
Community



Версия: 2021.3.3
Сборка: 213.7172.26
17 марта 2022 г.

[Системные требования](#)

[Инструкция по установке](#)

[Другие версии](#)

[Стороннее ПО](#)

Скачать PyCharm

[Windows](#) [macOS](#) [Linux](#)

Professional

Для научной и веб-разработки на Python. Поддерживает HTML, JS и SQL.

Скачать

Бесплатная пробная версия

Community

Для разработки только на Python

Скачать

Бесплатная, на базе открытого исходного
кода



Установите Toolbox App для удобного
скачивания и обновления PyCharm

Программирование на Python

Создание проекта в PyCharm



Создавая проект для Python нужно выбрать его окружение. Проще говоря ограничить видимость мест, в которых он будет искать установленные библиотеки для проекта. К чему это может привести?

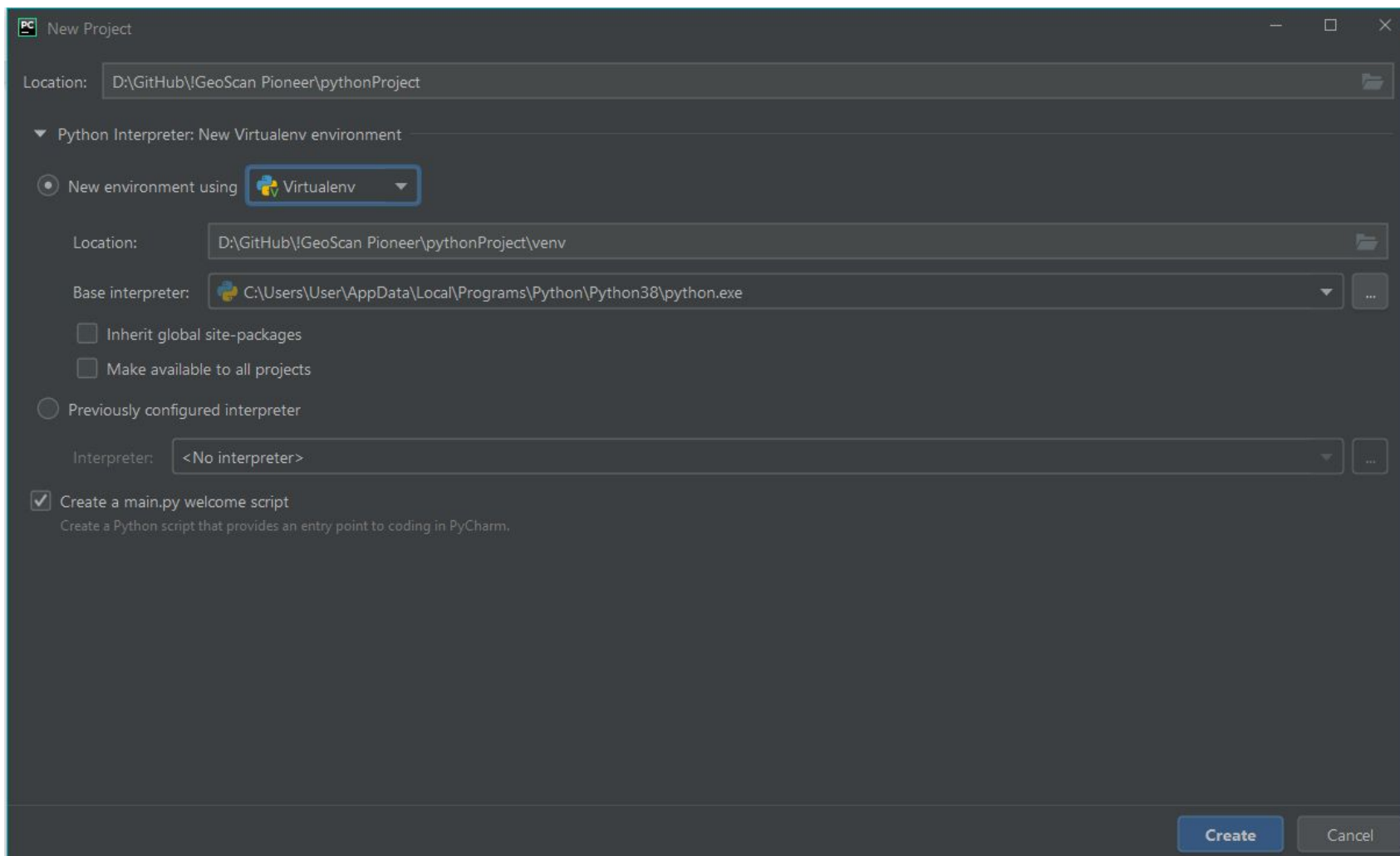
- Невозможность использовать библиотеки разных версий
- Если вы пользуетесь не своим компьютером, то вам могут просто запретить доступ к папке, в которой хранятся библиотеки.

Что дает виртуальное окружение?

- В каждый новый проект библиотеки ставятся индивидуально. У проекта А всегда будут нужные только ему библиотеки и проект Б никогда не увидит библиотеки проекта А.
- Для передачи проекта другому пользователю достаточно будет выполнить лишь одну команду, которая создаст список используемых библиотек. А пользователю, соответственно, нужно будет выполнить одну команду с этим списком, для установки всех библиотек.

Программирование на Python

Создание проекта в PyCharm



Программирование на Python

Решение проблем



Обратите внимание, что после создания проекта в терминале должно быть показано, что окружение является виртуальным.

```
Terminal: Local x +  
Microsoft Windows [Version 10.0.19044.1586]  
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.  
(venv) D:\GitHub\!GeoScan Pioneer\Education>
```

The screenshot shows a terminal window with a red circle around the prompt `(venv)` and a red arrow pointing to it. The terminal output includes the Windows version and copyright information. The IDE interface shows 'Structure' and 'Favorites' on the left, and 'Git', 'TODO', 'Problems', 'Terminal', 'Python Packages', and 'Python Console' at the bottom.

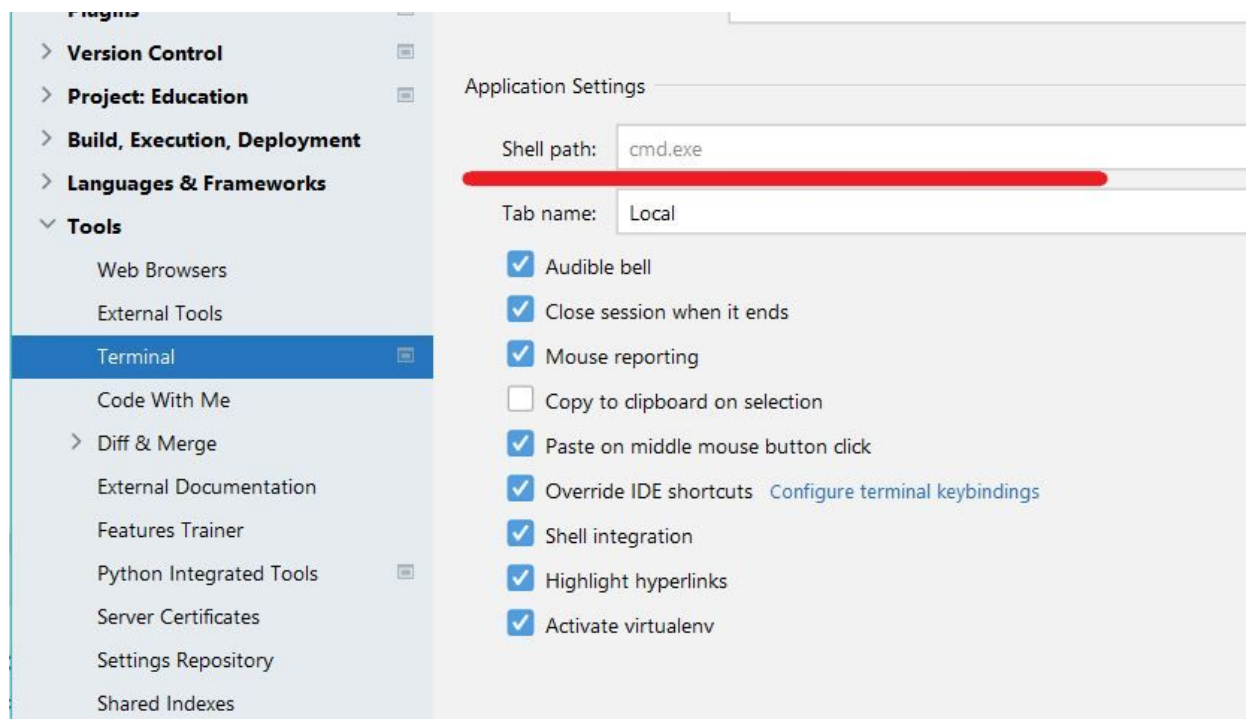
Программирование на Python

Решение проблем



Если в терминале не указано, что проект виртуальный, а вы создавали его именно виртуальным, то необходимо выполнить следующие действия:

- В PyCharm открыть настройки (например сочетанием `ctrl + alt + s`);
- Перейти во вкладку Tools;
- Перейти во вкладку Terminal;
- Убедиться, что в графе “Shell path” выбрано `cmd`.



Программирование на Python

Установка библиотек



Кибердром 2022

МИНПРОМТОРГ
РОССИИ

Установка библиотек осуществляется через терминал командой:

«pip install “название библиотеки”»

```
Terminal: Local × +
Microsoft Windows [Version 10.0.19042.1237]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

(venv) D:\GitHub\!GeoScan Pioneer\pythonProject>pip install opencv-contrib-python
```

Программирование на Python

Обзор PioneerSDK



Pioneer

```
+get_raw_video_frame()
+arm()
+disarm()
+takeoff
+land()
+lua_script_control(input_state)
+led_control(led_id, r, g, b)
+go_to_local_point(x, y, z)
+point_reached()
+get_local_position()
+get_dist_sensor_data()
```

Функции класса:

get_raw_video_frame()- получить кадр с камеры.

arm() – запуск моторов.

disarm() – остановка моторов.

takeoff()- взлет.

land() – посадка.

lua_script_control() – запустить LUA скрипт.

led_control- установить цвет на светодиоды.

go_to_loacal_point() – отправить Пионер в заданную позицию.

point_reached() – вернет True, если коптер долетел до точки.

get_local_position()- получить локальные координаты.

get_dist_sensor_data() – получить данные с высотомера.

Вопросы для самоподготовки



- Какие дополнительные модули используются для навигации?
- Как объявить светодиоды в программе на LUA?
- Как отправить по UART данные в программе на LUA?
- Как прочитать данные из UART в программе на LUA?
- Как получить текущие координаты квадрокоптера в системе навигации LPS?
- Как получать события от АП?
- Как отправлять события АП?

Домашнее задание

обязательно



- Ознакомиться с примерами скриптов в PioneerStation;
- Написать программу на языке LUA для взлета и посадки квадрокоптера;
- Написать программу на языке LUA для полета по точкам квадрокоптера;
- Написать программу для активации магнита с тумблера пульта управления.

Дополнительные материалы



Кибердром²⁰²²

МИНПРОМТОРГ
РОССИИ

1. <https://docs.geoscan.aero/ru/master/index.html>
2. <https://trikset.com/products/trik-studio>
3. https://github.com/geoscan/pioneer_sdk