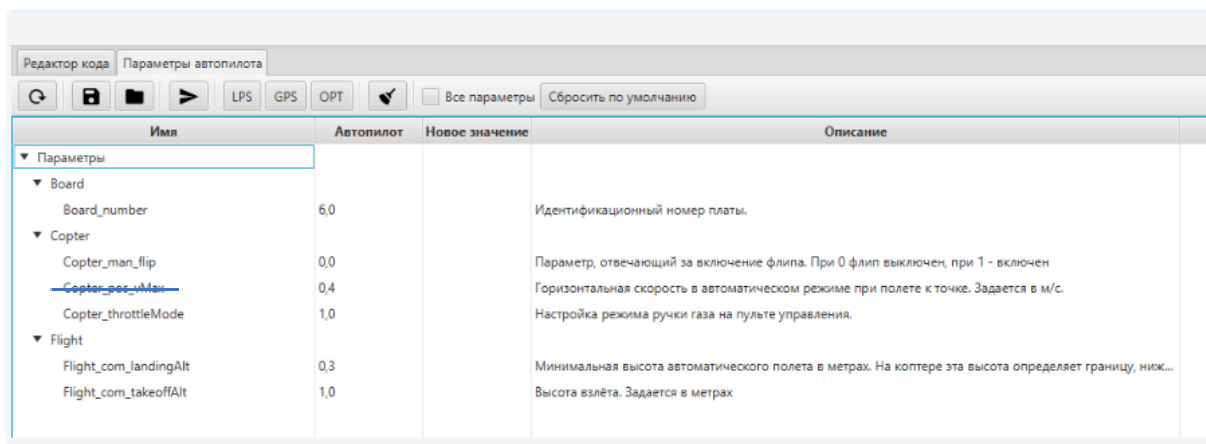
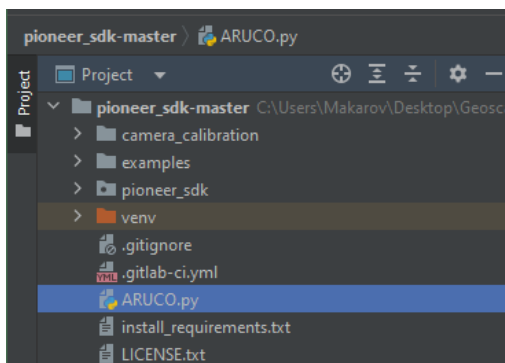


1. На компьютере установите рабочую среду [PyCharm Community](#), приложение для работы с Пионером [Pioneer Station](#), а также интерпретатор [Python 3.9](#).
2. Скачайте прошивку для автопилота [Pioneer Mini 1.6.9202](#). Подключите Пионер Мини к компьютеру с помощью micro USB.
3. Обновите прошивку автопилота согласно [инструкции](#).
4. С помощью утилиты NJET обновите прошивку ESP-32, [по инструкции](#).
5. Установите последнюю версию библиотеки [pioneer_sdk](#) и настройте среду PyCharm. Для этого следуйте следующей [инструкции](#). Пункт 9-11 выполнять необязательно. В качестве интерпретатора выберите Python 3.9 (в случае если он не был выбран автоматически).
6. В Pioneer Station подключитесь к дрону по micro USB и перейдите в параметры автопилота. В разделе Copter отрегулируйте значение Copter_pos_vMax, чтобы задать дрону скорость. Рекомендуется 0,4.



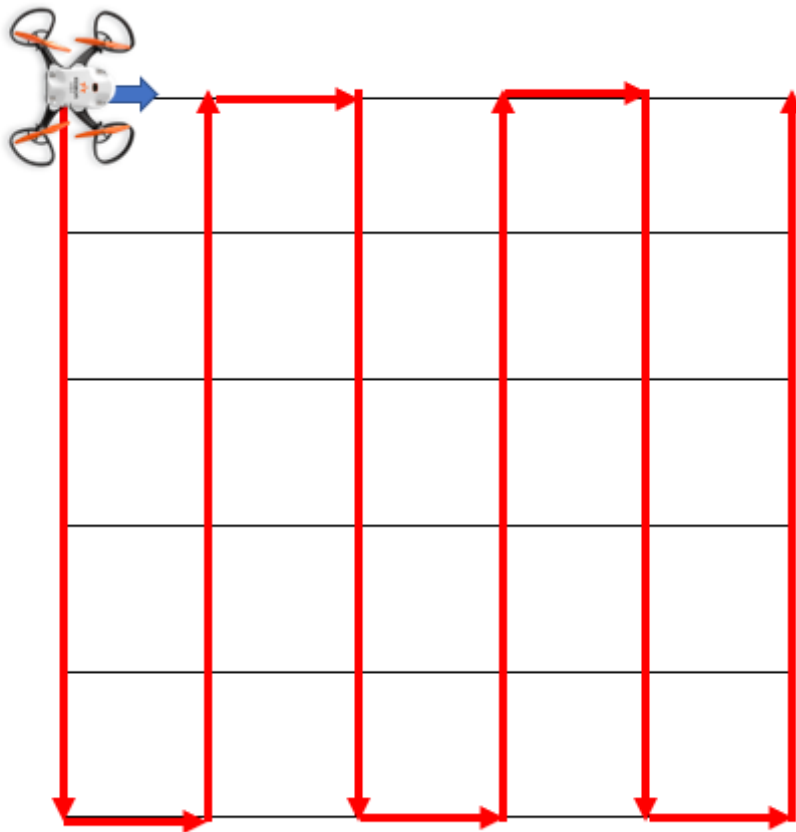
Имя	Автопилот	Новое значение	Описание
Board			
Board_number	6.0		Идентификационный номер платы.
Copter			
Copter_man_flip	0.0		Параметр, отвечающий за включение флипа. При 0 флип выключен, при 1 - включен
<u>Copter_pos_vMax</u>	0.4		Горизонтальная скорость в автоматическом режиме при полете к точке. Задается в м/с.
Copter_throttleMode	1.0		Настройка режима ручки газа на пульте управления.
Flight			
Flight_com_landingAlt	0.3		Минимальная высота автоматического полета в метрах. На коптере эта высота определяет границу, ниж...
Flight_com_takeoffAlt	1.0		Высота взлёта. Задается в метрах

7. Поместите приложенный скрипт ARUCO.py в папку pioneer_sdk-master созданную в результате выполнения пункта 5. В PyCharm в дереве с файлами выберите загруженный скрипт.



8. Проверьте скрипт на наличие предупреждений (выделенный красной линией код). Если они имеются, `pioneer_sdk` был инициализирован неправильно или некоторые библиотеки не смогли загрузиться.

9. В случае отсутствия предупреждений, изучите описание функции на строчках 19-21 и отредактируйте параметры на строчках 23-25, 27 при необходимости. Изначально полет рассчитан на поле 3x3 метра. Сам дрон при этом облетает траекторию 2,5x2,5 метра. Дрон располагается и летит так, как показано на рисунке (сторона одного квадрата – 0,5 метра):

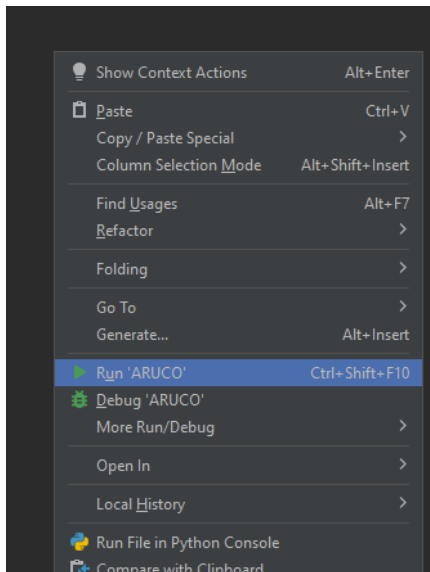


10. Подготовьте поле. Подробная инструкция представлена в [описании кейса](#), пункте 3. ArUco-метки должны быть зафиксированы, чтобы поток воздуха от лопастей не сдувал их.

11. Проверьте целостность дрона, в частности лопастей и батареи. Запитайте его и включите. Расположите на указанной выше позиции в указанном направлении. Камера дрона должна быть повернута вниз.

12. Подключитесь к дрону через сеть Wi-Fi. Пароль от сети по умолчанию – 12345678.

13. Запустите скрипт. Для этого в свободном месте в окне для редактирования кода нажмите ПКМ и выберите Run 'ARUCO'.



14. Дрон начнет выполнять скрипт и двигаться по заданной траектории. На компьютере появится окно с видео от камеры на Пионере, а в терминале PyCharm будут высвечиваться данные о полёте и процессе сканирование и сохранения найденных ArUco-меток.

При обнаружении метки дрон: обведет найденную метку на кадре, сохранит её ID в буфер (выведется по окончании программы в терминале), сохранит ее фотографию в папку ARUCO (создастся сама в папке pioneer_sdk-master) под названием «markerid_X», где X – ID найденной метки, помигает светодиодом.

По завершению полета дрон самостоятельно сядет и завершит скрипт через 5 секунд.