

# ТЕХНОЛОГИИ ГЕОСКАНА



Применение операционной системы PioneerOS  
в образовательной робототехнике

# Подходы к программированию БВС (и не только)



Программа на борту

## Плюсы

- Нет задержек выполнения команд

## Минусы

- Нужно приложение загрузки программы
- Сложно отлаживать программу
- Нельзя обрабатывать видеоизображение в режиме реального времени
- Никакого ИИ



Управление с внешнего устройства

## Плюсы

- Можно обрабатывать все что угодно
- Легко отлаживать программу
- Низкий порог входа

## Минусы

- Нужен канал связи
- Есть задержки выполнения команд
- Необходимо настраивать окружение



Бортовой вычислитель

## Плюсы

- Можно обрабатывать почти все что угодно
- Легко отлаживать программу
- Нет задержек выполнения команд

## Минусы

- Сложно настраивается бортовой вычислитель
- Высокий порог входа

# Почему бортовой вычислитель в образовании — сложно

Возникающая проблема	Что нужно педагогу
Нет настроенного образа ОС	Знания в области ИТ
Непонятно, как с ним работать	Много времени, чтобы разобраться
При переходе на другой вычислитель надо проходить все заново	Знания в области ИТ Много времени, чтобы разобраться
На компьютере работает, на бортовом вычислителе — нет	Знания в области ИТ
Надо все вручную настраивать	Много времени, чтобы разобраться
Почему-то камеры и датчики не работают	Инженерные навыки Много времени, чтобы разобраться



Несколько примеров микрокомпьютеров.  
Для каждого нужно искать и настраивать ОС

# Наш ответ — PioneerOS

PioneerOS — это первая в России операционная система, разработанная для образовательных квадрокоптеров серии «Геоскан Пионер». Она представляет собой дистрибутив Linux, основанный на актуальной версии Ubuntu.

# PioneerOS

v0.5.0-dev for Geoscan Pioneer Mini 2

Packages: Ubuntu stable (noble)  
Support: <https://t.me/GeoscanPioneer>  
IP addresses: (LAN) IPv4: 10.42.0.1 IPv6: fe80::343a:7361:2a1f:f486  
Performance:

Load:	14%	Up-time:	0 min
Memory usage:	14% of 3.81G		
CPU temp:	48°C	Usage of /:	17% of 29G

Commands:

**Network** : sudo pionet  
**Upgrade** : sudo apt-get upgrade  
**Monitoring** : http

Last login: Tue Nov 25 14:59:03 2025 from 10.42.0.39  
pioneermini@pioneermini:~\$ █

# PioneerOS — ответ на все сложное

## Зачем нужна PioneerOS

- Облегчить жизнь пользователю — все уже настроено
- Предусмотрены разные инструменты программирования для разных возрастных групп
- Все ПО оптимизировано для работы с Пионерами
- Есть все необходимое для программирования, в том числе для работы с компьютерным зрением и ИИ
- Нет необходимости что-либо настраивать на компьютере
- Сложно сломать саму ОС
- Новые инструменты, расширяющие возможности программирования



Легкий старт



Вычисления  
на борту



Все настроено  
«из коробки»



Для работы нужен  
только браузер

# Для кого подойдет PioneerOS

## Начинающий

- Первый раз видит квадрокоптер
  - Не знает, что такое ИИ
  - Не владеет языками программирования
  - Нужно освоить дроны и ИИ
- 
- Блочное программирование
  - Встроенные нейросети

## Продвинутый

- Имеет базовые навыки Python
  - Пользовался нейронными сетями в повседневных задачах
  - Должен побеждать в соревнованиях
- 
- Программирование через Web IDE
  - Обучение своих нейросетей
  - Шоу дронов

## Профессионал

- Исследователь в области робототехники
  - Ищет инструмент для реализации своих идей
  - Глубокое понимание кода и технологий
- 
- Работа с консолью
  - Тонкая настройка алгоритмов ИИ
  - Загрузка собственных библиотек на дрон

# Возможности PioneerOS

## Библиотеки

- **Pioneer SDK2** — работа с полетным контроллером
- **Pioneer RKNN** — работа с ИИ

## Pioneer Code

- **Pioneer Bricks** — блочное программирование
- **Code OSS** — текстовое программирование
- **Регистр моделей ИИ** — управление файлами ИИ
- **Галерея** — управление фото и видео

## Система

- **Медиасервер** — работа с камерой, трансляция любого видеопотока
- Обновление прошивок и параметров АП по Wi-Fi через Pioneer Station 2
- Работа с полетным контроллером из нескольких программ
- Работа с видеопотоком камер из нескольких программ
- Можно поставить любую библиотеку

# Новые библиотеки: Pioneer SDK2

Pioneer SDK2 — принципиально новая библиотека для работы с полетным контроллером.

## Особенности:

- Быстрее SDK1
- Минимальное время задержки между командами
- Обратно совместима с SDK1
- Позволяет работать с полезными нагрузками
- Настроена под конкретный борт
- Получение изображения со всех камер дрона (если их несколько)
- Запуск трансляции видео в браузер
- Защита от неправильной последовательности действий
- Несколько способов работы с событиями АП (как в LUA, так и в SDK1)
- Много методов, которых не хватало в первой версии

```
from pioneer_sdk2 import Pioneer, Event
from pioneer_sdk2 import Camera, CameraType

def event_callback(event: Event):
    if event == Event.ENGINES_STARTED:
        print(f'Engines started')
    elif event == Event.TAKEOFF_COMPLETE:
        print(f'Takeoff completed')

pioneer = Pioneer()
pioneer.subscribe(event_callback, Event.TAKEOFF_COMPLETE)
pioneer.subscribe(event_callback, Event.ENGINES_STARTED)
camera_main = Camera(CameraType.MAIN)
camera_opt = Camera(CameraType.OPT)

pioneer.arm()
pioneer.takeoff()

pioneer.grab_open()
print(pioneer.get_fly_state())
print(pioneer.get_motors_rpm())
pioneer.set_param('Copter_man_rcMode0', 1)
print(pioneer.get_nav_system())

pioneer.land()
pioneer.close_connection()
```

# Кто же такой этот ваш ИИ: Pioneer RKNN

Pioneer RKNN — библиотека для выполнения моделей ИИ на нейровычислительном модуле (NPU).

## Особенности:

- Запуск ИИ двумя строчками кода
- Поддерживает «из коробки» распознавание объектов, поз, текста
- Имеет собственное хранилище файлов моделей (не надо помнить, где лежит файл модели)
- Интеграция с популярными библиотеками компьютерного зрения (OpenCV)
- Можно добавлять свои модели, решающие другие задачи (конечно, надо понимать, что делаешь)

**Важно: обучать модели ИИ  
все еще надо на компьютере**



```
import cv2, numpy as np
from pioneer_rknn import Yolo
from pioneer_sdk2 import Camera

def resize_img(img):
    img_copy = cv2.resize(img, (640, 640))
    return np.expand_dims(img_copy, 0)

camera = Camera()
img = resize_img(camera.get_cv_frame())

model = Yolo(model_name="yolov8n")
boxes, classes, scores = model.run([[img]])
print(boxes, classes, scores)
```

# Платформа Pioneer Code

**Pioneer Code** — единая среда разработки, в которой можно выполнять все необходимые действия по настройке, проверке и работе с коптером.

## Что в нее входит:

- Pioneer Bricks — среда блочного программирования
- Code OSS IDE — среда текстового программирования
- Регистр моделей ИИ — хранилище файлов моделей ИИ
- Галерея — хранилище фото и видео

## Особенности:

- Работает на борту
- Чтобы зайти, нужно любое устройство (неважно какое) с браузером
- Красивый и понятный дизайн

Чтобы начать работу, нужно подключиться к бортовому вычислителю по Wi-Fi и в браузере ввести одну строчку.

The screenshot displays the Pioneer Code web interface. On the left is a sidebar menu with the following items: Pioneer Code (with a gear icon), Pioneer Bricks (with a puzzle piece icon), Среда разработки (with a code editor icon), Модели ИИ (with a brain icon), Галерея медиа (with a photo gallery icon), and О приложении (with an information icon). The main content area is divided into four quadrants, each featuring an illustration and a text block:

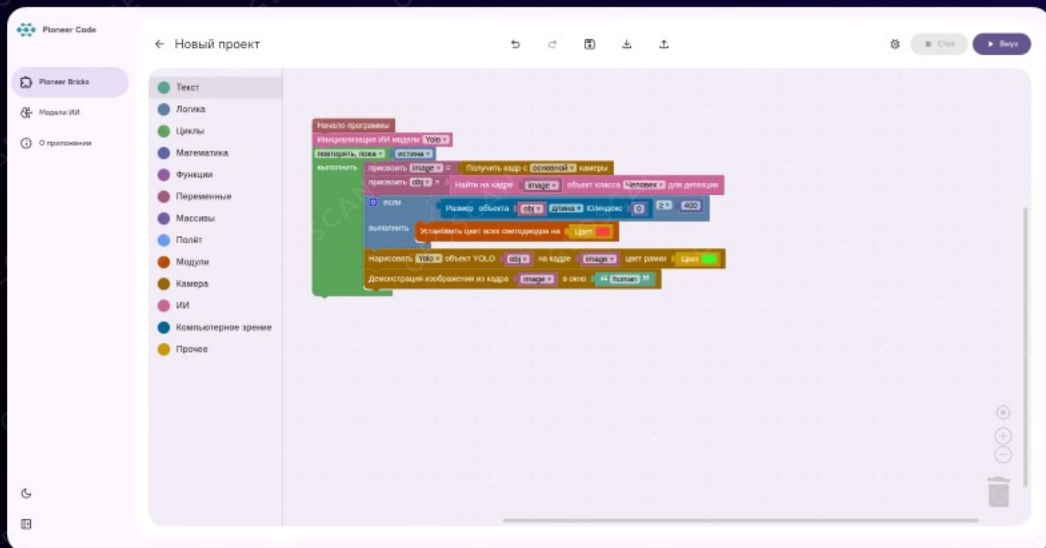
- Top Left:** Illustration of a person with puzzle pieces. Title: **Pioneer Bricks**. Text: "Собирай код как пазл! Освой азы программирования в визуальной среде. Перетаскивай блоки, управляй дронами и сразу смотри на результаты." Subtext: "Начни кодить с «Pioneer Bricks»!"
- Top Right:** Illustration of a person at a laptop. Title: **Среда разработки**. Text: "Пиши код как профи! Используй мощный редактор с полноценной подсветкой синтаксиса и подсказками. Получи полный контроль через терминал Linux." Subtext: "Реализуй свою любую идею!"
- Bottom Left:** Illustration of a person with a laptop and AI icons. Title: **Модели ИИ**. Text: "Обучай и загружай ИИ! Добавляй и управляй моделями машинного обучения для компьютерного зрения прямо на борту." Subtext: "Сделай своего «Пилонера» умнее!"
- Bottom Right:** Illustration of a person with a laptop and media icons. Title: **Галерея медиа**. Text: "Сохраняй лучшие кадры! Просматривай и управляй фото и видео с камеры дрона. Сохраняй яркие моменты и очкай память устройства." Subtext: "Просмотри свои открытия!"

# Блочное программирование: Pioneer Bricks

**Pioneer Bricks** — среда блочного программирования на основе Blockly.

## Особенности:

- Это Blockly — знакомый для многих интерфейс
- Есть блоки для работы с ИИ
- Есть блоки для работы с компьютерным зрением
- При запуске трансляции можно сразу ее посмотреть
- Несколько пользователей могут работать одновременно
- Есть вывод программы
- Есть блоки для работы с полезными нагрузками
- Можно экспортировать или импортировать проект
- Набор блоков предоставляется под конкретный борт



# Текстовое программирование: Code OSS

Code OSS — веб IDE на основе  
Visual Studio Code.

## Особенности:

- Привычный инструмент для тех, кто программирует
- Все настроено для программирования на Python
- Запуск/остановка программ по одной кнопке
- Подсветка кода
- Подсказки по коду
- Если очень хочется, можно программировать и не на Python
- Если очень хочется, есть доступ к терминалу PioneerOS
- Полностью на русском языке (доступно переключение на английский)
- Настроены инструменты отладки

```

1 import cv2
2 import argparse
3 import numpy as np
4 from time import sleep
5 from threading import Thread
6
7 from pioneer_3dx2 import Camera, ImageViewer
8 from pioneer_3dx2 import Pioneer
9 from pioneer_rkm import Yolo
10
11 # Input size
12 IMG_SIZE = [640, 640]
13
14 # All classes
15 classnames = ["person", "bicycle", "car", "motorbike", "aeroplane", "bus", "train", "truck", "boat",
16 "traffic light", "fire hydrant", "stop sign", "parking meter", "bench", "bird", "cat",
17 "dog", "horse", "sheep", "cow", "elephant", "bear", "zebra", "giraffe", "backpack", "umbrella",
18 "handbag", "tie", "suitcase", "frisbee", "skis", "snowboard", "sports ball", "kite", "baseball bat",
19 "baseball glove", "skateboard", "surfboard", "tennis racket", "bottle", "wine glass", "cup",
20 "fork", "knife", "spoon", "bowl", "banana", "apple", "sandwich", "orange", "broccoli",
21 "carrot", "hot dog", "pizza", "donut", "cake", "chair", "sofa", "pottedplant", "bed",
22 "diningtable", "toilet", "tvmonitor", "laptop", "mouse", "remote", "keyboard", "cell phone",
23 "microwave", "oven", "toaster", "sink", "refrigerator", "book", "clock", "vase", "scissors",
24 "teddy bear", "hair drier", "toothbrush"]
25
26
27 # Define the dictionary and parameters
28 aruco_dict = cv2.Aruco_getPredefinedDictionary(cv2.Aruco_DICT_ARUCO_ORIGINAL)
29 parameters = cv2.Aruco_DetectorParameters()
30
31 only_class_detected = False
32
33 # Draw boxes
34 def yolo_draw(image, boxes, scores, classes, only_class = 0, yolo_scale = 1):
35     global only_class_detected
36
37     for box, score, cl in zip(boxes, scores, classes):
38         if only_class == cl or only_class == -1:
39             left, top, right, bottom = [int(b) for b in box]
40             top = int(top + yolo_scale[1])
41             bottom = int(bottom + yolo_scale[1])
42             left = int(left + yolo_scale[0])
43             right = int(right + yolo_scale[0])
44             print("%0.1f %0.1f %0.1f %0.1f %0.1f" % (classnames[cl], top, left, right, bottom, score))
45             cv2.rectangle(image, (left, top), (right, bottom), (0, 255, 0))
46             cv2.putText(image, classnames[cl],
47                         (left, top - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 255, 0))

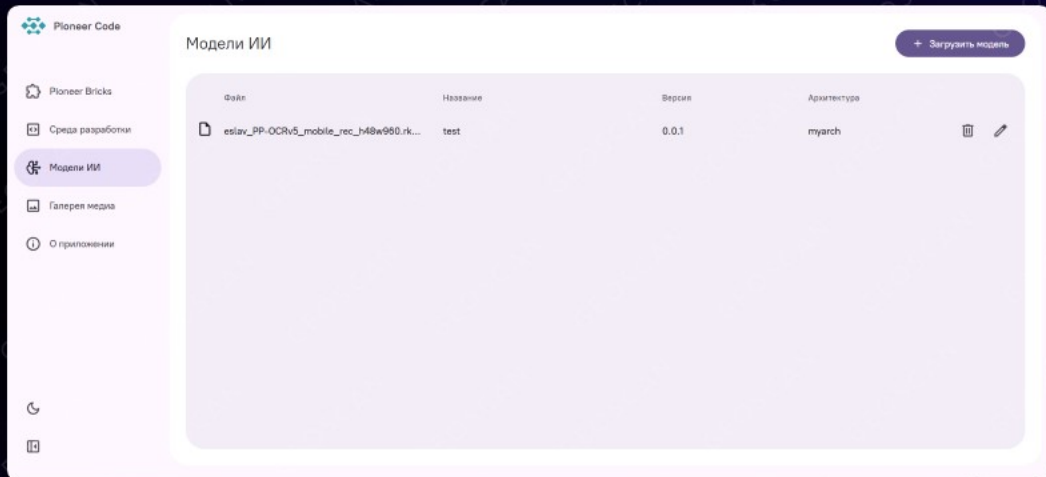
```

# Регистр моделей ИИ

Регистр моделей ИИ — особое хранилище для файлов моделей.

## Особенности:

- Позволяет добавлять/удалять модели с борта
- Позволяет посмотреть все установленные модели
- Поддерживает интеграцию с Pioneer RKNN

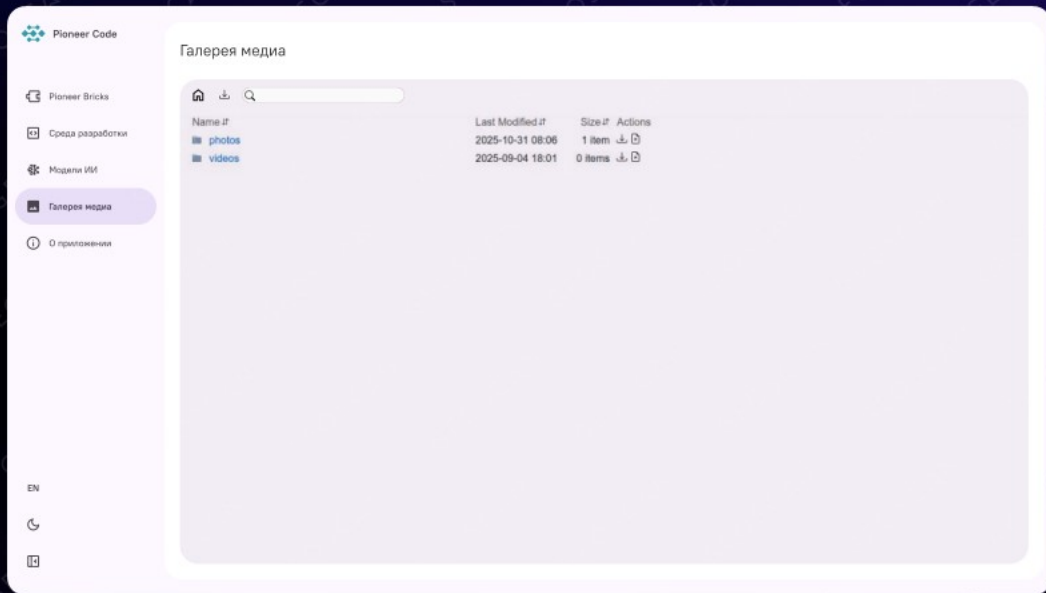


# Галерея

Галерея — единое хранилище фото и видео.

## Особенности:

- Позволяет сохранять фото и видео из кода, блоков
- Позволяет скачивать на любое устройство фото или видео
- Есть предпросмотр фото и видео



# Устройства с PioneerOS



Геоскан Пионер Мини 2



Геоскан Пионер\*

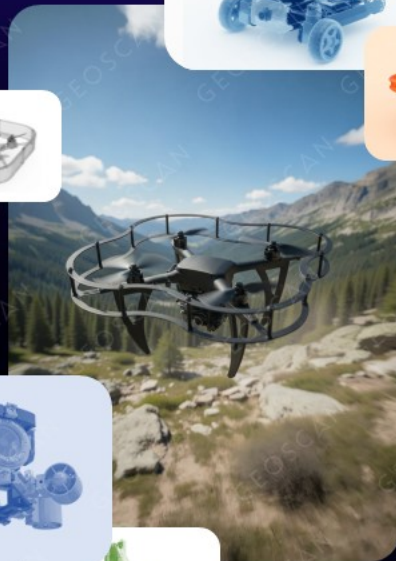


Концепция «Пионер X»



Попробовать  
возможности PioneerOS  
на своих устройствах

# PioneerOS — скоро во всех роботах страны



Спасибо за внимание!



GEOSCAN

**Илья Даниленко**

Руководитель отдела разработки  
платформенных решений  
и автономной навигации  
ГК «Геоскан»

Санкт-Петербург, ул. Политехническая, д. 22л

Москва, Колпачный переулок, д. 6, стр. 3

8 800 333-84-77, +7 812 363-33-87

[info@geoscan.ru](mailto:info@geoscan.ru)

[geoscan.ru](http://geoscan.ru)