

Министерство образования Псковской области
Государственное бюджетное образовательное учреждение
дополнительного профессионального образования Псковской области «Центр
оценки качества образования»

РАССМОТРЕНО

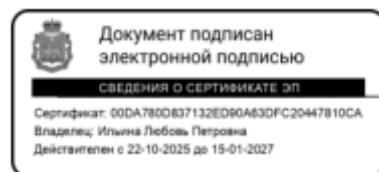
Научно-методическим советом

протокол № 7

от 31.08.2026 г.

УТВЕРЖДАЮ

Директор ГБОУ ДПО ПО «ЦОКО»



Л.П. Ильина

« 01 » _____ 09 _____ 2026 г.

Дополнительная общеразвивающая программа для детей
«Программирование роботов»

Уровень: стартовый

Направленность: углубленный

Возраст обучающихся: 10-13 лет

Срок реализации: 1 год

Составитель:

Сапаров Виталий Александрович,
педагог дополнительного образования
центра цифрового образования «IT-куб»

г. Псков, 2026 г.

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Программа «Программирование роботов» разработана в соответствии с требованиями нормативных документов:

- Федерального закона «Об образовании в Российской Федерации» от 29.12.2012 № 273-ФЗ;
- Распоряжения Правительства РФ от 31.03.2022 № 678-р «Об утверждении Концепции развития дополнительного образования детей до 2030 года и плана мероприятий по ее реализации» (с изменениями и дополнениями);
- Приказа Минпросвещения России от 27.07.2022 № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» (Зарегистрировано в Минюсте России 26.09.2022 № 70226);
- Постановления Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи» (вместе с «СП 2.4.3648-20. Санитарные правила...») (Зарегистрировано в Минюсте России 18.12.2020 № 61573);
- Постановления Правительства РФ от 26.12.2017 № 1642 (с изменениями и дополнениями) «Об утверждении государственной программы Российской Федерации «Развитие образования»
- Распоряжения Правительства РФ от 29.05.2015 № 996-р «Об утверждении Стратегии развития воспитания в Российской Федерации на период до 2025 года».

Направленность программы: техническая.

Уровень освоения программы: повышенный

Актуальность программы обусловлена стремительным развитием концепции Индустрии 4.0, интернета вещей (IoT), киберфизических и беспилотных автономных систем. Современное производство требует от специалистов навыков работы на стыке электроники, механики и высокоуровневого программирования. Данная программа позволяет преодолеть разрыв между базовой учебной робототехникой (конструированием) и промышленными технологиями автоматизации.

Новизна программы заключается в реализации **гибридного подхода** к проектированию роботов. Вместо изоляции платформ учащиеся объединяют конструктивные преимущества блочного конструктора [LEGO Mindstorms EV3](#) (стабильная кинематика, надёжные редукторы, точные энкодеры) и вычислительную мощность одноплатного компьютера Raspberry Pi под управлением Windows. Это позволяет интегрировать в проекты школьников технологии компьютерного зрения (OpenCV), сложные сетевые протоколы связи (MQTT, WebSockets) и распределенные вычисления.

Цель и задачи программы

Цель программы: Формирование инженерно-технических компетенций учащихся в области проектирования, конструирования и высокоуровневого программирования гибридных автономных робототехнических систем.

Задачи:

- **Обучающие (предметные):**
 - Освоить синтаксис языка программирования Python на продвинутом уровне.
 - Изучить архитектуру ОС Windows, принципы администрирования через командную строку.
 - Освоить принципы работы с GPIO-интерфейсами (ШИМ, I2C, SPI, UART).
 - Изучить методы цифровой обработки изображений и видеопотока с помощью OpenCV.
 - Освоить программирование сетевого взаимодействия (клиент-серверная архитектура на сокетах и MQTT).
- **Развивающие (метапредметные):**
 - Развить алгоритмическое, логическое и пространственное мышление.
 - Формировать исследовательские навыки, умение анализировать и фильтровать физические данные (подавление дребезга кнопок, сглаживание шумов датчиков).
 - Развить навыки работы над комплексными проектами (от технического задания до финальных испытаний).
- **Воспитательные (личностные):**
 - Воспитывать культуру командной работы, лидерские качества и умение распределять задачи в группе.
 - Формировать целеустремленность, усидчивость и аккуратность при сборке электрических схем и отладке программного кода.

Категория обучающихся: программа предназначена для учащихся в возрасте от 10 до 13 лет, не требует предварительных знаний и входного тестирования.

Срок реализации программы: программа рассчитана на 1 год, количество учебных часов – 144 (из расчёта 6 учебных часов в неделю).

Формы и режим занятий

- **Форма обучения:** очная, очная с применением дистанционных технологий.
- **Режим занятий:** занятия проводятся в группах до 12 человек, длительность одного занятия составляет 3 академических часа, периодичность занятий – 2 раза в неделю. Продолжительность одного академического часа – 35 минут. После окончания одного академического часа организовывается перерыв длительностью 5 минут для проветривания помещения и отдыха обучающихся.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ

Личностные результаты:

- Сформированность навыков критического анализа результатов своей работы и выявления причин программных или конструктивных сбоев.
- Готовность к продуктивному профессиональному самоопределению в сфере ИТ и инженерных технологий.
- Умение эффективно взаимодействовать в коллективе, брать на себя ответственность за стабильность части общего программного кода команды.

Метапредметные результаты:

- Умение самостоятельно формулировать алгоритмы решения нестандартных задач автоматизации.
- Опыт применения математического аппарата (геометрия кадра, тригонометрические расчеты расстояний, дифференциальные коэффициенты регуляторов) в прикладных задачах.
- Навык поиска, извлечения и структурирования технической информации из официальной англоязычной документации (API библиотек, документация Linux).

Предметные результаты:

- Свободное владение инструментами командной строки ОС Linux для администрирования робота.
- Умение писать, оптимизировать и отлаживать скрипты на языке Python средней и высокой сложности.

- Опыт проектирования электронных схем с использованием GPIO портов, транзисторных ключей, АЦП и различных шин обмена данными.
- Способность программировать системы компьютерного зрения реального времени с фреймрейтом (FPS), достаточным для управления физическим роботом.
- Знание принципов построения локальных компьютерных сетей и умение организовывать межплатформенное сетевое взаимодействие (Sockets, MQTT).

СОДЕРЖАНИЕ ПРОГРАММЫ

2.1 Учебно-тематический план (144 часа)

№	Название модуля	Всего (ч)	Теория (ч)	Практика (ч)	Форма контроля
1	Знакомство с платформами, ОС Windows и расширенным Python	24	7	17	Тестирование, мини-скрипты
2	Сложная микроэлектроника и физические интерфейсы GPIO	24	5	19	Защита практической схемы
3	Низкоуровневое программирование LEGO EV3 на базе ОС ev3dev	28	5	23	Квалификационные заезды
4	Распределенные вычисления и сетевая синхронизация	20	4	16	Проверка сетевого обмена
5	Компьютерное зрение OpenCV и пространственная навигация	24	4	20	Выполнение мини-кейсов
6	Инженерно-проектный практикум и интеграция систем	24	1	23	Публичная защита проекта
	ИТОГО	144	26	118	

2.2. Учебный план

№ зан.	Тема занятия	Теория (ч)	Практика (ч)	Форма контроля
Модуль 1	Знакомство с платформами, экосистемой Windows и расширенным Python	7	17	
1	Введение в курс. Инструктаж по ТБ. Архитектура гибридных систем (Raspberry Pi + LEGO EV3).	1	1	Входное собеседование
2	Аппаратное устройство Raspberry Pi. Разъемы, интерфейсы, периферия.	1	1	Устный опрос (компоненты платы)
3	Подготовка карт памяти в ОС Windows. Использование утилиты Raspberry Pi Imager для Windows.	0.5	1.5	Практическая работа (первый запуск робоплатформы)
4	Настройка удаленного доступа из Windows. Работа с клиентами PuTTY (SSH) и VNC Viewer для Windows.	0.5	1.5	Наблюдение (подключение к плате с ПК)
5	Основы сетевого взаимодействия в Windows. Командная строка (CMD) и Windows PowerShell. Команда ping.	1	1	Работа в консоли Windows
6	Удаленное управление файлами робота из Windows. Работа по протоколу SFTP через FileZilla / WinSCP.	0.5	1.5	Экспресс-тест (перенос файлов с ПК на робота)
7	Среда разработки Python под Windows. Настройка IDLE, Thonny и VS Code в Windows-окружении.	1	1	Проверка настройки IDE на ПК
8	Управляющие конструкции Python: Сложные ветвления (условия if-elif-else).	0.5	1.5	Защита мини-задачи на ветвление кода
9	Организация циклов в Python (for, while), операторы управления циклом.	0.5	1.5	Оценка алгоритмической логики скрипта
10	Сложные коллекции данных: Списки, кортежи, словари в Python.	0.5	1.5	Защита задачи на обработку массивов

11	Функциональное программирование. Создание собственных функций. Пакетный менеджер pip в Windows.	0.5	1.5	Проверка модульности кода
12	Конструкция обработки исключений try-except и чтение конфигурационных файлов.	0.5	1.5	Зачетный мини-хакатон по Модулю 1
Модуль 2	Сложная микроэлектроника и физические интерфейсы GPIO	5	19	
13	Порты GPIO на Raspberry Pi. Физика интерфейса, распиновка, ограничения по току.	1	1	Фронтальный опрос по схеме распиновки
14	Цифровой вывод. Расчет токоограничительных резисторов. Сборка схем со светодиодами.	0.5	1.5	Контроль собранной на Breadboard схемы
15	Цифровой ввод. Подключение тактовых кнопок. Борьба с дребезгом контактов.	0.5	1.5	Проверка скрипта (обработка нажатий)
16	Широтно-импульсная модуляция (ШИМ). Плавное управление яркостью.	0.5	1.5	Демонстрация работы ШИМ-эффекта
17	Протоколы последовательной передачи данных. Архитектура и принципы шины I2C.	0.5	1.5	Детекция адресов I2C-устройств
18	Работа с I2C-устройствами: Считывание цифровых датчиков (BMP280/BH1750).	0.5	1.5	Проверка вывода телеметрии в консоль
19	Шина SPI и аналогово-цифровое преобразование. Изучение чипа MCP3008.	0.5	1.5	Защита электрической схемы подключения АЦП
20	Чтение показаний аналоговых датчиков (потенциометры, фоторезисторы) через АЦП.	0.25	1.75	Анализ графиков изменения аналоговых величин
21	Ультразвуковой дальномер HC-SR04. Физика ультразвука, расчет делителя напряжения.	0.5	1.5	Измерение эталонных расстояний датчиком
22	Управление сервоприводами. Задание углов поворота через ШИМ-сигнал.	0.25	1.75	Проверка точности позиционирования вала
23	Интерфейс UART. Принципы организации последовательного	0.5	1.5	Проверка текстового обмена между устройствами

	соединения COM-портов в Windows.			
24	Практикум: Сборка комплексной метеостанции с выводом тревоги на светодиоды.	0	2	Защита практической схемы по Модулю 2
Модуль 3	Программирование LEGO EV3 (EV3 Classroom & расширенный код)	6	22	
25	Установка и знакомство с интерфейсом EV3 Classroom для Windows. Палитра блоков Scratch.	1	1	Проверка настройки конфигурации шасси
26	Управление перемещением в EV3 Classroom. Движение по времени, оборотам и градусам.	0.5	1.5	Выполнение упражнения «Квадрат»
27	Работа с датчиками расстояния (УЗ и ИК) в среде EV3 Classroom. Вложенные условия.	0.5	1.5	Остановка робота строго перед преградой
28	Движение по лабиринту в EV3 Classroom: реализация «Правила правой руки» на блоках.	0.5	1.5	Зачетный заезд: выход из лабиринта на Scratch
29	Подготовка SD-карт расширения для EV3 из-под Windows. Форматирование FAT32/NTFS.	0.5	1.5	Прошивка микропроцессорного блока EV3
30	Настройка сетевого общего доступа к Интернету (ICS) в Windows для подключения EV3 по USB.	0.5	1.5	Проверка пинга (ping) между ПК и платами
31	Настройка расширения "EV3 MicroPython" / "ev3dev" в среде VS Code под Windows.	0.5	1.5	Проверка связи VS Code с роботом
32	Управление исполнительными механизмами на Python: классы LargeMotor и MediumMotor.	0.5	1.5	Наблюдение за вращением моторов на Python
33	Чтение показаний энкодеров. Точное перемещение робота по градусам и оборотам в коде.	0.25	1.75	Измерение точности проезда дистанции
34	Обработка показаний дискретных датчиков (касание) и дальномеров на Python.	0.25	1.75	Проверка скрипта аварийного торможения
35	Датчик цвета EV3. Калибровка режимов отраженного света и чтения RGB-составляющих.	0.25	1.75	Проверка вывода названий цветов в терминал

36	Использование гироскопического датчика на Python. Борьба с аппаратным дрейфом нуля.	0.5	1.5	Проверка разворота робоплатформы на 90°
37	Алгоритмы движения: Пропорционально-Дифференциальный (ПД) регулятор следования по линии.	0.5	1.5	Расчет дифференциальной составляющей в коде
38	Практикум: Сборка и калибровка мобильного шасси робота для соревнований по линии.	0	2	Квалификационные заезды по линии на Python
Модуль 4	Распределенные вычисления и сетевая синхронизация	4	16	
39	Концепция Master-Slave (Ведущий-Ведомый) в робототехнике. Распределение ролей.	1	1	Устный опрос (архитектурная схема задач)
40	Настройка беспроводной Wi-Fi сети через Windows-маршрутизатор (точку доступа).	0.5	1.5	Проверка распределения IP-адресов на платах
41	Введение в сетевые сокеты (Sockets). Принципы работы протоколов TCP и UDP.	0.5	1.5	Тестирование отправки строк через сокет-клиент
42	Разработка серверной части сетевого интерфейса на Python (на стороне Raspberry Pi).	0.25	1.75	Проверка готовности сервера слушать порты
43	Разработка клиентского скрипта на Python (на стороне блока LEGO EV3).	0.25	1.75	Регистрация успешного сокет-соединения
44	Передача управляющих команд движения с Raspberry Pi и их отработка моторами EV3.	0.25	1.75	Оценка задержки от команды до старта моторов
45	Передача пакетов телеметрии датчиков с EV3 на Raspberry. Использование JSON.	0.25	1.75	Проверка валидности структуры JSON-пакета
46	Введение в IoT-протоколы. Архитектура MQTT. Установка Windows-версий MQTT-клиентов (MQTT Explorer).	0.5	1.5	Мониторинг топиков брокера из-под Windows
47	Реализация подписок на топики управления с помощью библиотеки raho-mqtt.	0.5	1.5	Проверка отправки данных через MQTT-клиент

48	Практикум: Дистанционное управление движением робота EV3 через терминал Raspberry Pi.	0	2	Проверка работы сетевого обмена данными
Модуль 5	Компьютерное зрение OpenCV и пространственная навигация	4	20	
49	Подключение модулей камер (CSI/USB) к Raspberry Pi. Захват кадра средствами Python.	0.5	1.5	Вывод статичного изображения с камеры на экран
50	Знакомство с библиотекой OpenCV. Матрицы изображений, базовые типы данных (cv2.Mat).	0.5	1.5	Проверка изменения разрешения видеопотока
51	Цветовые пространства цифровых кадров. Перевод изображения из BGR в HSV.	0.5	1.5	Подбор цветовых диапазонов с помощью трекбаров
52	Пороговая обработка кадра (Бинаризация). Морфологические операции фильтрации шумов.	0.5	1.5	Оценка чистоты бинарной маски объекта
53	Анализ контуров объектов. Поиск центраида через моменты изображения.	0.25	1.75	Отрисовка маркера центра объекта на видео
54	Создание алгоритма слежения за цветным объектом (теннисным мячом) в кадре.	0.25	1.75	Вычисление величины отклонения от центра
55	Интеграция зрения и движения: Расчет рулевого усилия для EV3 на основе видеоданных.	0.25	1.75	Удержание роботом дистанции до мяча
56	Геометрический анализ кадра. Выделение прямых линий с помощью преобразования Хафа.	0.5	1.5	Проверка детекции краев реальной полосы
57	Написание алгоритма движения гибридного робота по широкой видеолнии (камера).	0.25	1.75	Тестовый проезд робота по трассе по камере
58	Распознавание локационных маркеров ArUco. Чтение ID маркера и расчет расстояния.	0.25	1.75	Вывод ID распознанного маркера в окно OpenCV
59	Детекция дорожных знаков («Стоп», «Поворот») с использованием шаблонов.	0.25	1.75	Проверка качества срабатывания при разном свете
60	Практикум: Робот-курьер, ориентирующийся по видеоразметке и знакам дорожного движения.	0	2	Выполнение миникейса (остановка на «Стоп»)

Модуль 6	Инженерно-проектный практикум и интеграция систем	1	23	
61	Постановка комплексных инженерных задач. Разбиение на команды, составление ТЗ.	1	1	Защита бланка технического задания (ТЗ)
62	Конструирование гибридной платформы. Размещение плат, Raspberry Pi и АКБ на шасси EV3.	0	2	Контроль жесткости конструкции и камеры
63	Электротехническая интеграция. Пайка цепей отдельного электропитания робота.	0	2	Проверка отсутствия КЗ мультиметром
64	Разработка низкоуровневого ПО для блока EV3 (прием пакетов, аварийная защита).	0	2	Проверка отказоустойчивости прошивки EV3
65	Разработка высокоуровневого ПО на Raspberry Pi (архитектура конечных автоматов, отладка через удаленный рабочий стол Windows).	0	2	Защита алгоритмической блок-схемы проекта
66	Синхронизация систем верхнего и нижнего уровней. Минимизация пинга. Анализ сетевых логов в консоли Windows PowerShell.			Анализ задержек (Latency) обмена данными
	Комплексирование данных (Sensor Fusion) со зрения, УЗ-датчиков и энкодеров. Мониторинг потока данных через ПО под Windows.			Наблюдение за реакцией на динамические преграды
	Комплексные ходовые испытания гибридных роботов на испытательном полигоне. Выявление аппаратных и программных люфтов.			Фиксация замечаний и ошибок в бортовом журнале
	Оптимизация программного кода. Многопоточность (threading) для ускорения OpenCV процессов. Замеры производительности.			Измерение прироста частоты кадров (FPS)
	Финальная отладка алгоритмов пространственной навигации и точная подгонка регуляторов движения на соревновательной трассе.			Контрольный предзащитный заезд робоплатформы

	Подготовка презентационных материалов в Microsoft PowerPoint, графическое оформление электрических схем в Windows-редакторах.			Проверка комплектности проектной документации
	Итоговое занятие. Демо-день. Публичная защита созданных робототехнических систем. Прохождение соревновательного полигона.			Публичная защита проекта перед жюри
	ИТОГО ПО МОДУЛЮ 6:	1	23	
	ВСЕГО ПО ПРОГРАММЕ:	26	118	ОБЩИЙ ОБЪЕМ: 144 ЧАСА

2.3. Содержание

Модуль 1. Знакомство с платформами, ОС Linux и расширенным Python (24 часа)

- **Теория:** Понятие распределенных систем. Архитектура Raspberry Pi и ARM-процессоров. Файловая система Windows. Синтаксис Python, типы данных, области видимости, основы модульного программирования и обработки ошибок.
- **Практика:** Создание загрузочной флеш-карты в [Raspberry Pi Imager](#), удаленная настройка по SSH/VNC. Управление системой через Bash-терминал. Создание программ на Python с использованием ветвлений, циклов, функций, коллекций (list, dict) и внешних пакетов (pip).

Модуль 2. Сложная микроэлектроника и физические интерфейсы GPIO (24 часа)

- **Теория:** Физика полупроводников, законы Ома и Кирхгофа применительно к слаботочным цепям. Назначение GPIO. Цифровые и аналоговые сигналы. Принципы работы интерфейсов I2C, SPI, UART.
- **Практика:** Сборка схем на беспаячных макетных платах (Breadboard). Подключение светодиодов (расчет токоограничивающих резисторов), аппаратное и программное устранение дребезга кнопок. Генерация сигналов ШИМ. Считывание аналоговых сенсоров через внешний АЦП MCP3008. Работа с ультразвуковым датчиком HC-SR04, сервоприводами и I2C-датчиками климата/освещенности.

Модуль 3. Низкоуровневое программирование LEGO EV3 на базе ОС ev3dev (28 часов)

- **Теория:** Ограничения стандартной микропрограммы LEGO. Операционная система ev3dev (Debian Linux для роботов). Теория

автоматического управления: релейный, пропорциональный (П) и пропорционально-дифференциальный (ПД) регуляторы движения.

- **Практика:** Прошивка блока EV3, сетевое сопряжение по USB/Wi-Fi. Использование библиотеки `python-ev3dev`. Программирование моторов (LargeMotor, MediumMotor), считывание энкодеров для точного движения по углам. Работа с фирменными датчиками (касание, гироскоп, ультразвук, цвет в режиме RGB). Написание алгоритма скоростного и плавного движения по сложной линии на базе ПД-регулятора.

Модуль 4. Распределенные вычисления и сетевая синхронизация (20 часов)

- **Теория:** Архитектура Master-Slave. Основы компьютерных сетей: модель TCP/IP, IP-адресация, порты, сокет. Протокол промышленного интернета вещей MQTT (концепция Publisher/Subscriber).

- **Практика:** Создание беспроводного моста связи между Raspberry Pi и EV3. Разработка клиент-серверного приложения на сокетах в Python для трансляции управляющих команд и телеметрии датчиков. Установка MQTT-брокера Mosquitto на Raspberry Pi. Сериализация пакетов данных в формат JSON.

Модуль 5. Компьютерное зрение OpenCV и пространственная навигация (24 часа)

- **Теория:** Матричная структура цифрового кадра. Цветовые пространства (RGB/BGR, HSV). Фильтрация шумов, морфология. Поиск контуров. Преобразование Хафа для выделения прямых линий. Принципы распознавания маркеров (ArUco) и шаблонов.

- **Практика:** Настройка модуля CSI-камеры. Цветовое маскирование (выделение объектов по цвету). Определение геометрического центра объекта (центроида) по моментам изображения. Написание алгоритма следования робота за цветным мячом. Реализация видеодатчика линии. Детекция знаков дорожного движения и считывание ArUco-маркеров для точной парковки робота.

Модуль 6. Инженерно-проектный практикум и интеграция систем (24 часа)

- **Теория:** Жизненный цикл инженерного проекта. Составление ТЗ. Принципы комплексирования данных со множества разнородных датчиков (Sensor Fusion).

- **Практика:** Командное проектирование сложного робота-доставщика или автономного транспортного средства. Разработка общей механической платформы, монтаж Raspberry Pi и силовой электрической цепи (раздельное питание логики от PowerBank и моторов от АКБ LEGO).

Интеграция программных модулей, устранение сетевых задержек. Отладка робота на комплексном полигоне (линии, перекрестки, препятствия, дорожные знаки). Подготовка презентации и технической документации.

3. ФОРМЫ АТТЕСТАЦИИ И ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

Формы контроля:

1. **Входной контроль:** Собеседование или экспресс-тестирование базовых навыков логики и конструирования.
2. **Текущий контроль:** Оценка выполнения практических кейсов в конце каждого подмодуля (например, «Подключить датчик I2C и вывести графики», «Добиться прохождения роботом EV3 ломаной линии за нормативное время»).
3. **Итоговый контроль: Демо-день (публичная защита проектов).** Робот команды должен в полностью автономном режиме пройти соревновательный полигон, имитирующий элементы умного города или склада (чтение знаков, движение по разметке, объезд препятствий, выгрузка на финише), а команда обязана представить презентацию и защитить архитектуру своего кода перед экспертной комиссией.

4. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

1. Оборудование и электроника

- **Рабочие станции:** 6 ноутбуков или ПК с ОС Windows 10/11, модулем Wi-Fi и SSD-накопителем.
 - **Робототехника:** 6 базовых наборов LEGO Mindstorms EV3 (45544) с полным комплектом штатных датчиков и моторов.
 - **Микрокомпьютеры:** 6 плат Raspberry Pi (модели 3B+ / 4 / 5) в защитных корпусах с блоками питания.
 - **Память:** 12 карт памяти MicroSD (16–32 Гб, Class 10) и 3–6 USB-картридеров.
 - **Зрение и питание:** 6 камер (CSI или USB) и 6 внешних аккумуляторов (PowerBank, выход от 2.1A).
 - **Сеть:** 1 беспроводной Wi-Fi роутер для создания локальной сети класса.
- #### **2. Инструменты для пайки и интеграции (на класс)**
- **3 паяльные станции** с цифровой регулировкой температуры (200–480°C).

- Расходники для пайки: оловянно-свинцовый **припой** (ПОС-61) и жидкий спиртоканифольный **флюс** (СКФ).

- Инструменты: 3 мультиметра, стрипперы для зачистки проводов, кусачки, 6 беспаячных макетных плат (Breadboard) и комплекты проводов-перемычек.

3. Программное обеспечение (ПО для Windows)

- **Среда Scratch:** [LEGO Education EV3 Classroom для Windows](#).
- **Среды разработки:** Thonny Python IDE и Visual Studio Code (VS Code) с расширением *ev3dev-browser*.
- **Утилиты и связь:** [Raspberry Pi Imager](#) (запись ОС), **PuTTY** / PowerShell (SSH-терминал), [VNC Viewer](#) (удаленный рабочий стол), **WinSCP** (передача файлов по SFTP), **MQTT Explorer** (отладка сети).

4. Испытательный полигон

- Виниловые баннеры с черными линиями для движения по разметке.
- Комплект съемных пластиковых или картонных стенок для сборки лабиринта.
- Набор карточек со знаками дорожного движения и ArUco-маркерами для калибровки камер.

5. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

Программа реализуется при наборе группы в течение учебного года. В очной форме проводится на базе Центра цифрового образования «ИТ-куб» ГБОУ ДПО ПО «Центр оценки качества образования» и общеобразовательных организаций при наличии технических возможностей по договору сетевого взаимодействия с ОО.

6. СПИСОК ЛИТЕРАТУРЫ

1. **Монк, С.** Робототехника на Raspberry Pi для юных конструкторов и программистов / С. Монк ; пер. с англ. — СПб. : БХВ-Петербург, 2020. — 208 с.
2. **Монк, С.** Мейкерство. Arduino и Raspberry Pi / С. Монк. — СПб. : БХВ-Петербург, 2023. — 352 с.
3. **Платт, Ч.** Электроника для начинающих / Ч. Платт ; пер. с англ. — 3-е изд. — СПб. : БХВ-Петербург, 2024. — 416 с.

4. **Колисниченко, Д. Н.** Командная строка Linux и автоматизация рутинных задач / Д. Н. Колисниченко. — СПб. : БХВ-Петербург, 2022. — 304 с. [[1](#), [2](#)]
5. **По программированию на языке Python (Модули 1 и 4):**
6. **Доусон, М.** Програмируем на Python / М. Доусон. — СПб. : Питер, 2023. — 416 с.
7. **Севериансе, Ч.** Python для всех. Введение в программирование и анализ данных / Ч. Севериансе. — М. : ДМК Пресс, 2021. — 242 с.
8. **Уизерилл, Д.** Python для хакеров. Нетривиальные задачи и проекты / Д. Уизерилл. — СПб. : Питер, 2023. — 400 с. [[1](#)]
9. **По робототехнике LEGO Mindstorms и Linux (Модуль 3):**
10. **Филиппов, С. А.** Робототехника для детей и родителей / С. А. Филиппов. — 5-е изд. — СПб. : Наука, 2017. — 320 с. (*Рекомендовано для изучения теории ПД-регуляторов и кинематики*).
11. **Васильев, М. В.** Программирование учебных роботов на базе Lego Education EV3 на языке Python / М. В. Васильев // Вестник Шадринского государственного педагогического университета. — 2022. — № 4 (56). — С. 15–20. [[1](#), [2](#)]
12. **По компьютерному зрению OpenCV (Модуль 5):**
13. **Кэпитал, Д.** Изучаем OpenCV 4 / Д. Кэпитал, Э. Кауэр. — М. : ДМК Пресс, 2020. — 540 с.
14. **Лутц, М.** Обработка изображений на Python. Компьютерное зрение и OpenCV / М. Лутц. — М. : ДМК Пресс, 2022. — 280 с.
15. **Официальный сайт Raspberry Pi Foundation** — raspberrypi.org.
Руководства по установке ОС, документация по портам ввода-вывода (GPIO).
16. **Портал ОС ev3dev (Debian Linux для LEGO)** — ev3dev.org.
Официальная база знаний, инструкции по установке ядра ОС и API-библиотеки для Python.
17. **Официальная документация библиотеки OpenCV** — opencv.org.
Примеры реализации цветовой фильтрации (HSV), поиска контуров и трекинга объектов.
18. **Репозиторий библиотек Python (PyPI)** — <https://pypi.org>. Поиск и установка специфических модулей для работы с датчиками (например, raho-mqtt, RPi.GPIO).
19. **Инженерный портал по робототехнике Robototehnika.ru** — <http://robototehnika.ru>. Методические статьи по теории автоматического

управления (ТАУ) и конструированию мобильных робоплатформ. [[1](#), [2](#), [3](#), [4](#)]