Уважаемые коллеги!

Предлагаемый методический комплекс предназначен для преподавания робототехники в 6-м классе. Его задача — продолжить знакомить школьников с основами конструирования и программирования роботов, развить инженерное мышление и навыки командной работы.

Курс выстроен по принципу «от простого к сложному»: от сборки механических конструкций — к работе с электронными компонентами и датчиками, от управления одним мотором — к комплексным алгоритмам и групповым проектам. Особое внимание уделяется практической деятельности: более 70 % времени учащиеся проводят за сборкой моделей, написанием программ и тестированием роботов.

В рамках модуля рассматриваются ключевые темы:

- мобильная робототехника;
- конструирование и управление роботами;
- работа с датчиками;
- программирование движений в компьютерно-управляемой среде;
- управление сервомотором;
- реализация группового проекта и знакомство с профессиями в робототехнике.

Комплекс включает теоретические материалы, практические задания и критерии оценки. Особое внимание уделено межпредметным связям (математика, физика, информатика) и профориентационному компоненту.

Надеемся, что курс станет для вас надёжным инструментом, а для учащихся — увлекательным введением в мир робототехники.

Занятие 1 (2 часа). «Мобильная робототехника: функциональное разнообразие и устройство транспортных роботов»

Цель: сформировать представление о транспортных роботах, их назначении, конструктивных особенностях и сферах применения.

Необходимое оборудование:

- мультимедийный проектор/экран;
- образцы мобильных роботов (или макеты/модели);
- робототехнические конструкторы (для демонстрации узлов);
- раздаточные материалы (схемы, фото транспортных роботов разных типов);
- рабочие листы для практической работы;
- бланки для заполнения характеристик робота.

Ход занятия

1. Вводная часть (10 мин)

- приветствие, проверка готовности (2 мин);
- актуализация знаний: вопросы «Что такое мобильный робот?», «Где применяются роботы-перевозчики?» (3 мин);
- постановка цели и задач занятия (5 мин).

2. Теоретический блок 1: мобильная робототехника и транспортные роботы (15 мин)

- **мобильная робототехника** — роботы, способные самостоятельно перемещаться в пространстве;
- **транспортные роботы** — автономные устройства для перевозки грузов/пассажиров, сбора данных;
- **основные функции:**
    - транспортировка материалов;
    - доставка товаров;
    - перемещение внутри помещений/на открытых территориях;
    - мониторинг и сбор данных.

Демонстрация: фото/видео примеров транспортных роботов (складские тягачи, беспилотники, марсоходы).

3. Теоретический блок 2: классификация транспортных роботов (20 мин)

- **по способу перемещения грузов:**
    - тягачи (буксируют тележки);
    - погрузчики (поднимают/перемещают грузы);

- платформы (перевозят на собственной поверхности).

- **по способу управления:**
  - автономные (работают по программе);
  - дистанционно управляемые (оператор через пульт/интерфейс);
  - гибридные (автономный режим + ручное управление).
- **по конструкции:**
  - колёсные;
  - гусеничные;
  - воздушные (БПЛА);
  - подводные;
  - гибридные (например, колёсно-летающие).
- **по среде эксплуатации:**
  - внутри помещений (склады, больницы);
  - на открытых территориях (стройплощадки, поля);
  - в экстремальных условиях (зоны катастроф, космос).

4. Теоретический блок 3: гусеничные и колёсные транспортные роботы (20 мин)

- **колёсные роботы:**
  - преимущества: высокая скорость, манёвренность, простота обслуживания;
  - недостатки: низкая проходимость на неровных поверхностях;
  - сферы применения: склады, логистические центры, офисы.
- **гусеничные роботы:**
  - преимущества: проходимость по грязи, снегу, ступеням, пересечённой местности;
  - недостатки: меньшая скорость, повышенный износ гусениц;
  - сферы применения: строительство, сельское хозяйство, спасательные операции.
- **сравнительный анализ:** таблица с критериями (проходимость, скорость, стоимость, область применения).

Демонстрация: разбор конструкции колёсного и гусеничного робота-конструктора (сравнение движителей, шасси, креплений).

5. Практическая работа «Характеристика транспортного робота» (20 мин)

- **этап 1. Выбор модели (5 мин):**
  - группы получают фото/макеты 2–3 транспортных роботов (например, складской тягач, гусеничный робот-спасатель, беспилотник-доставщик);
  - задание: выбрать один робот для анализа.

- **этап 2. Заполнение бланка характеристики (15 мин):**

учащиеся вносят данные по шаблону:

  - 1. Название/тип робота: _____
  - 2. Назначение: _____
  - 3. Способ перемещения грузов: _____
  - 4. Способ управления: _____
  - 5. Тип движителя (колёса/гусеницы/др.): _____
  - 6. Особенности конструкции: _____
  - 7. Среда эксплуатации: _____
  - 8. Преимущества: _____
  - 9. Недостатки: _____
  - 10. Перспективы улучшения: _____

примеры заполнения для разных типов роботов (на доске/слайде).

6. Рефлексия и подведение итогов (5 мин)

- обсуждение:
  - «Почему для склада выбирают колёсные, а не гусеничные роботы?»,
  - «Как способ управления влияет на безопасность робота?»,
  - «Какие инновации можно внедрить в транспортные роботы?»;
- ответы на вопросы;
- анонс следующего занятия (конструирование колёсной платформы).

Домашнее задание:

1. Найти 1 пример транспортного робота, не разобранного на уроке. Описать его по плану из практической работы (10–12 предложений).
2. Нарисовать эскиз своего транспортного робота для доставки книг в школьной библиотеке. Указать:
   - тип движителя;
   - способ перевозки груза (полка, захват и т. п.);
   - 2 датчика, которые помогут роботу избегать препятствий.
3. Подготовить 1 вопрос к одноклассникам: «Как можно адаптировать робота для работы зимой?».

Критерии оценки практической работы:

- точность определения характеристик робота;

- полнота заполнения бланка (все 10 пунктов);
- аргументированность выводов при обсуждении;
- аккуратность оформления эскиза (домашнее задание).

Занятие 2 (2 часа). Гусеничные роботы: сборка и базовое управление

Цель: освоить сборку гусеничного робота и программирование базовых движений (вперёд/назад) в визуальной среде.

Необходимое оборудование:

- робототехнические конструкторы с гусеничными модулями;
- компьютеры с установленной визуальной средой программирования (Scratch, LEGO EV3, VEXcode);
- USB-кабели для подключения контроллеров;
- полигон с разметкой для тестирования;
- рабочие листы для фиксации результатов.

Ход занятия (90 мин):

1. **Вводная часть (10 мин)**
    - приветствие, проверка готовности (2 мин);
    - актуализация знаний: вопросы «В чём преимущества гусеничных роботов?», «Где они применяются?» (3 мин);
    - постановка цели и задач (5 мин).
2. **Теоретический блок: гусеничные роботы (15 мин)**
    - особенности конструкции:
        - гусеничная лента, катки, ведущие колёса;
        - повышенная проходимость, устойчивость;
    - сферы применения:
        - строительство, сельское хозяйство, спасательные операции;
        - работа на рыхлых/скользких поверхностях.
    - сравнение с колёсными роботами (таблица: проходимость vs скорость).
3. **Практическая работа: сборка модели (30 мин)**
    - этап 1. Подготовка компонентов (5 мин):
        - проверка наличия деталей (гусеницы, моторы, контроллер, шасси);
    - этап 2. Сборка ходовой части (15 мин):
        - монтаж гусеничных модулей на шасси;

        - подключение моторов к контроллеру;
    - этап 3. Проверка соединений (10 мин):
        - тестирование вращения гусениц;
        - фиксация проблем (проскальзывание, перекосы).
4. **Программирование базовых движений (25 мин)**
    - этап 1. Знакомство со средой программирования (5 мин):
        - интерфейс, блоки движения, загрузка программы;
    - этап 2. Программирование «вперёд» (10 мин):
        - блок «включить мотор на N сек»;
        - настройка мощности моторов;
    - этап 3. Программирование «назад» (10 мин):
        - изменение направления вращения;
        - синхронизация гусениц.
5. **Тестирование и рефлексия (10 мин)**
    - запуск робота на полигоне (движение по прямой);
    - обсуждение:
        - «Почему робот может поворачивать при движении вперёд?»,
        - «Как добиться прямолинейности?»,
        - «Какие проблемы возникли при сборке?».

Домашнее задание:

1. Нарисовать схему подключения моторов гусеничного робота к контроллеру (с подписями).
2. Написать алгоритм (5–7 команд) для движения вперёд на 5 сек, остановки на 2 сек, движения назад на 3 сек.
3. Найти 1 пример реального гусеничного робота (например, робот-сапёр), описать его функции (3–4 предложения).

Занятие 3 (2 часа). Колёсные роботы и оптимизация управления через переменные

Цель: собрать колёсного робота и освоить программирование поворотов и оптимизацию кода с помощью переменных.

Необходимое оборудование:

- робототехнические конструкторы с колёсными модулями;
- компьютеры с визуальной средой программирования;
- полигон с препятствиями для тестирования;

- рабочие листы с шаблонами алгоритмов.

Ход занятия (90 мин):

1. **Вводная часть (5 мин)**
    - проверка домашнего задания (обсуждение примеров гусеничных роботов, 3 мин);
    - постановка цели (2 мин).
2. **Теоретический блок: колёсные роботы (10 мин)**
    - типы колёсных схем:
        - 2-колёсная (дифференциальное управление);
        - 4-колёсная (повышенная устойчивость);
    - преимущества: скорость, манёвренность, простота обслуживания;
    - недостатки: низкая проходимость на рыхлых поверхностях;
    - примеры применения: складские роботы, курьерские платформы.
3. **Практическая работа: сборка колёсного робота (25 мин)**
    - этап 1. Выбор схемы (5 мин):
        - обсуждение 2- и 4-колёсных вариантов;
    - этап 2. Сборка шасси (15 мин):
        - монтаж колёс, моторов, контроллера;
        - балансировка конструкции;
    - этап 3. Подключение электроники (5 мин):
        - проверка вращения колёс.
4. **Теоретический блок: переменные в программировании (10 мин)**
    - что такое переменная: ячейка памяти для хранения данных (скорость, угол, время);
    - зачем нужны:
        - повторное использование значений;
        - упрощение изменений в коде;
    - пример: скорость = 50 → использование в блоках движения.
5. **Практическая работа: программирование поворотов и оптимизация (30 мин)**
    - этап 1. Базовые повороты (10 мин):
        - поворот налево/направо через разное вращение колёс;
        - калибровка угла (экспериментально).
    - этап 2. Введение переменных (10 мин):
        - создание переменной «скорость»;

        - замена числовых значений в блоках на переменную.
    - этап 3. Оптимизация программы (10 мин):
        - добавление переменной «время_движения»;
        - тестирование с разными значениями переменных.
6. **Тестирование, рефлексия и подведение итогов (10 мин)**
    - демонстрация работы робота (движение вперёд, повороты);
    - обсуждение:
        - «Как переменные упростили код?»,
        - «Что сложнее: управлять гусеничным или колёсным роботом?»,
        - «Где ещё можно применить переменные?»;
    - анонс следующего модуля (датчики и их программирование).

Домашнее задание:

1. Составить таблицу сравнения гусеничных и колёсных роботов (5 критериев: проходимость, скорость, сложность сборки, сфера применения, энергопотребление).
2. Написать программу для колёсного робота:
    - движение вперёд на время_1 сек;
    - поворот направо на угол_1 градусов;
    - движение назад на время_2 сек.
      Использовать переменные для всех числовых значений.
3. Придумать задачу для робота (например, объезд препятствия), описать алгоритм с использованием переменных (4–5 шагов).

Критерии оценки практических работ:

- правильность сборки модели;
- работоспособность робота (движение без перекосов);
- корректность программ (использование переменных, выполнение поворотов);
- аккуратность заполнения рабочих листов;
- активность в обсуждении.

Занятие 4 (2 часа). Датчик расстояния: принцип работы и программирование

Цель: изучить назначение, принцип работы датчика расстояния; освоить программирование реакций робота на сигналы датчика.

Необходимое оборудование:

- робототехнические конструкторы с датчиками расстояния;

- компьютеры с визуальной средой программирования (Scratch, LEGO EV3, VEXcode);
- USB-кабели для подключения контроллеров;
- полигон с препятствиями для тестирования;
- рабочие листы для фиксации алгоритмов и результатов.

Ход занятия (90 мин):

1. **Вводная часть (10 мин)**
    - приветствие, проверка готовности (2 мин);
    - актуализация знаний: вопросы «Для чего роботу нужны датчики?», «Какие датчики вы уже знаете?» (3 мин);
    - постановка цели и задач (5 мин).
2. **Теоретический блок 1: датчики в робототехнике (15 мин)**
    - **функции датчиков:** сбор информации об окружающей среде (расстояние, свет, касание);
    - **элементы управления схемы робота:**
        - датчик (вход) → контроллер (обработка) → мотор (выход);
    - **понятие обратной связи:**
        - робот получает данные от датчика → корректирует действия;
        - пример: «увидел препятствие → остановился».
3. **Теоретический блок 2: датчик расстояния (15 мин)**
    - **типы:** ультразвуковой, инфракрасный;
    - **принцип работы:**
        - излучение сигнала → отражение от объекта → замер времени возврата → расчёт расстояния;
    - **основные характеристики:**
        - диапазон измерений (например, 5–250 см);
        - точность (погрешность ±1 см);
        - угол обзора.
    - **сферы применения:**
        - избегание препятствий;
        - парковка;
        - навигация в помещениях.
4. **Практическая работа «Программирование работы датчика расстояния» (40 мин)**
    - **этап 1. Подключение датчика (5 мин):**

        - монтаж датчика на робота;
        - соединение с контроллером (указанный порт).
    - **этап 2. Калибровка (10 мин):**
        - измерение расстояний до объектов;
        - фиксация минимальных/максимальных значений.
    - **этап 3. Программирование алгоритма (15 мин):**
        - блок-схема:
        - если расстояние < 20 см → стоп, поворот направо на 90°;
        - иначе → движение вперёд.
        - реализация в визуальной среде (использование блока «если… то…»).
    - **этап 4. Тестирование (10 мин):**
        - запуск робота на полигоне с препятствиями;
        - фиксация сбоев (ложные срабатывания, задержки);
        - отладка программы.
5. **Рефлексия и подведение итогов (10 мин)**
    - обсуждение:
        - «Как датчик расстояния улучшает автономность робота?»,
        - «Что произойдёт, если датчик сломается?»,
        - «Где ещё применяют ультразвуковые датчики?»;
    - ответы на вопросы;
    - анонс следующего занятия (датчик линии).

Домашнее задание:

1. Нарисовать схему подключения датчика расстояния к контроллеру (с подписями портов и проводов).
2. Написать алгоритм (5–7 команд) для робота, который:
    - движется вперёд, пока расстояние > 30 см;
    - при расстоянии < 30 см останавливается, издаёт звуковой сигнал, ждёт 2 сек, поворачивает налево на 45°;
    - продолжает движение.
3. Найти 2 примера устройств (не роботов), где используется датчик расстояния. Указать их функции (2–3 предложения на пример).

Занятие 5 (2 часа). Датчик линии: принцип работы и программирование

Цель: изучить назначение и принцип работы датчика линии; освоить программирование движения робота по линии.

Необходимое оборудование:

- робототехнические конструкторы с датчиками линии;
- компьютеры с визуальной средой программирования;
- полигон с нанесёнными чёрными линиями (ширина 2–3 см);
- рабочие листы с шаблонами блок-схем.

Ход занятия (90 мин):

1. **Вводная часть (5 мин)**
    - проверка домашнего задания (обсуждение примеров устройств с датчиком расстояния, 3 мин);
    - постановка цели (2 мин).
2. **Теоретический блок: датчик линии (15 мин)**
    - **устройство:**
        - ИК-излучатель и приёмник;
        - аналоговый/цифровой выход.
    - **принцип работы:**
        - излучение ИК-луча → отражение от поверхности → замер интенсивности;
        - тёмная поверхность (чёрная линия) поглощает луч → низкий сигнал;
        - светлая поверхность отражает луч → высокий сигнал.
    - **параметры:**
        - чувствительность (регулировка порога срабатывания);
        - высота установки над поверхностью.
    - **сферы применения:**
        - сортировка объектов по цвету;
        - следование по маршруту (линия на складе, лабиринте).
3. **Практическая работа «Программирование работы датчика линии» (50 мин)**
    - **этап 1. Подключение и калибровка (10 мин):**
        - установка датчика на робот (высота 5–10 мм над поверхностью);
        - проверка сигналов на чёрной/белой поверхности (фиксация значений).
    - **этап 2. Блок-схема алгоритма (10 мин):**
        - условие:
        - если датчик видит чёрную линию → поворот налево;

        - если видит белую поверхность → поворот направо;
        - иначе → движение вперёд.
    - **этап 3. Программирование (15 мин):**
        - создание программы в визуальной среде;
        - использование циклов и условий;
        - настройка скорости моторов.
    - **этап 4. Тестирование и отладка (15 мин):**
        - запуск на полигоне (прямая линия → поворот → перекрёсток);
        - корректировка чувствительности датчика;
        - исправление ошибок (например, «проскакивание» линии).
4. **Рефлексия и подведение итогов (20 мин)**
    - демонстрация работы роботов (соревнование: кто быстрее пройдёт трассу);
    - обсуждение:
        - «Почему робот может сойти с линии?»,
        - «Как улучшить алгоритм для сложных трасс?»,
        - «Где применяют датчики линии в промышленности?»;
    - ответы на вопросы;
    - обзор следующих тем (датчики касания, программирование комплексных алгоритмов).

Домашнее задание:

1. Нарисовать схему датчика линии с подписями (излучатель, приёмник, выход).
2. Составить алгоритм (6–8 команд) для робота, который:
    - стартует на белой поверхности;
    - находит чёрную линию, движется вдоль неё 10 сек;
    - на перекрёстке (две линии) поворачивает направо;
    - продолжает движение до конца трассы.
3. Придумать задачу для робота с датчиком линии. Описать алгоритм решения (4–5 шагов).

Критерии оценки практических работ:

- правильность подключения датчика;
- работоспособность робота (стабильное следование по линии/избегание препятствий);
- корректность программы (использование условий, циклов, переменных);
- аккуратность заполнения рабочих листов;

- активность в обсуждении и отладке.

Занятие 6 (2 часа). «Управление движущейся моделью робота в компьютерно-управляемой среде»

Цель: освоить базовые инструменты визуального языка программирования для управления роботом; изучить принцип широтно-импульсной модуляции (ШИМ) и применить его на практике.

Необходимое оборудование:

- робототехнические конструкторы (с моторами, контроллером);
- компьютеры с установленной средой визуального программирования (LEGO EV3, VEXcode, Scratch с расширением для роботов);
- USB-кабели для подключения контроллеров;
- полигон с разметкой (старт, финиш, контрольные точки);
- рабочие листы для записи алгоритмов и результатов.

Ход занятия

1. Вводная часть (10 мин)

- приветствие, проверка готовности (2 мин);
- актуализация знаний: вопросы «Какие датчики мы уже изучали?», «Как робот получает команды на движение?» (3 мин);
- постановка цели и задач занятия (5 мин).

2. Теоретический блок 1: интерфейс визуального языка программирования (15 мин)

- обзор среды программирования:
    - палитра блоков (движение, датчики, логика, циклы);
    - область сборки программы;
    - панель загрузки на контроллер;
- основные команды для робота:
    - «включить мотор», «выключить мотор»;
    - «повернуть налево/направо»;
    - «ждать N сек»;
    - «если… то…».
- порядок создания программы:

1. сборка блоков в логическую последовательность;
2. проверка синтаксиса;

3. загрузка на контроллер;

4. тестирование.

Демонстрация: учитель показывает создание простой программы (движение вперёд на 3 сек).

3. Теоретический блок 2: широтно-импульсная модуляция (ШИМ) (15 мин)

- что такое ШИМ:
    - метод управления мощностью через изменение длительности импульсов;
    - аналог «открытия/закрытия крана» для мотора.
- зачем нужна:
    - плавное регулирование скорости вращения моторов;
    - экономия энергии;
    - точное позиционирование.
- параметры ШИМ:
    - частота импульсов (Гц);
    - коэффициент заполнения (0–100 %).
- пример:
    - 50 % заполнения → мотор работает на половине мощности;
    - 100 % → полная мощность.
- где применяется:
    - управление скоростью колёс;
    - регулировка яркости светодиодов;
    - контроль сервоприводов.

Наглядное пояснение: схема «сигнал ШИМ → мотор → скорость вращения».

4. Практическая работа «Программирование модели транспортного робота» (40 мин)

- **этап 1. Подготовка модели (5 мин):**
    - проверка соединений моторов и контроллера;
    - установка робота на полигон.
- **этап 2. Базовая программа движения (10 мин):**
    - задача: проехать от старта до финиша (50 см), остановиться на 2 сек, вернуться назад;
    - использование блоков:
        - «включить мотор на 50 % мощности»;
        - «ждать 3 сек»;

        - «выключить мотор».
- **этап 3. Введение ШИМ (10 мин):**
    - эксперимент:
        - программа 1 — мотор на 100 % (резкий старт/стоп);
        - программа 2 — плавное увеличение мощности от 20 % до 80 % за 2 сек;
    - сравнение результатов (плавность, точность остановки).
- **этап 4. Сложный маршрут (15 мин):**
    - задача: проехать по П-образному маршруту (вперёд → поворот направо → вперёд → поворот налево → назад);
    - использование циклов для повторения участков;
    - добавление условий («если датчик расстояния < 10 см → стоп»).

5. Тестирование, отладка и рефлексия (10 мин)

- запуск программ на полигоне (по очереди для каждой группы);
- фиксация проблем:
    - неточная остановка;
    - перекосы при повороте;
    - задержки в реакции.
- обсуждение:
    - «Как ШИМ влияет на движение робота?»,
    - «Почему важно регулировать мощность моторов?»,
    - «Какие ещё команды можно добавить в программу?»;
- ответы на вопросы.

Домашнее задание:

1. Нарисовать блок-схему алгоритма для робота, который:
    - стартует на белой поверхности;
    - движется вперёд с мощностью мотора 60 % до чёрной линии;
    - при обнаружении линии останавливается, издаёт звуковой сигнал, ждёт 1 сек, поворачивает направо на 90°;
    - продолжает движение до финиша.
2. Найти 2 примера устройств (не роботов), где используется ШИМ. Указать их функции и преимущества от применения ШИМ (3–4 предложения на пример).

3. Подготовить 1 вопрос к одноклассникам: «Как можно использовать ШИМ для улучшения работы робота-пылесоса?».

Критерии оценки практической работы:

- правильность подключения оборудования;
- работоспособность программы (выполнение маршрута без сбоев);
- корректное использование ШИМ (плавное движение, точная остановка);
- аккуратность заполнения рабочего листа (блок-схема, описание экспериментов);
- активность в обсуждении и отладке.

Занятие 7 (2 часа). Сервомотор: устройство и базовое программирование

Цель: изучить устройство сервомотора, освоить базовые приёмы программирования его движений.

Необходимое оборудование:

- робототехнические конструкторы с сервомоторами;
- компьютеры с визуальной средой программирования (LEGO EV3, VEXcode, Arduino IDE с блочным интерфейсом);
- USB-кабели для подключения контроллеров;
- угломеры или шаблоны для проверки позиций сервомотора;
- рабочие листы для записи команд и результатов.

Ход занятия (90 мин):

1. **Вводная часть (10 мин)**
    - приветствие, проверка готовности (2 мин);
    - актуализация знаний: вопросы «Чем мотор отличается от сервомотора?», «Где применяют сервоприводы?» (3 мин);
    - постановка цели и задач (5 мин).
2. **Теоретический блок: сервомотор (15 мин)**
    - **устройство:**
        - электродвигатель;
        - редуктор (понижение скорости, увеличение крутящего момента);
        - датчик обратной связи (потенциометр/энкодер);
        - управляющая плата.
    - **принцип работы:**
        - контроллер задаёт угол → сервомотор поворачивается → датчик сообщает текущую позицию → коррекция до достижения цели.

- **ключевые параметры:**
  - диапазон углов (обычно 0–180°);
  - скорость поворота (сек/60°);
  - крутящий момент (кг·см).
- **сферы применения:**
  - роботизированные захваты;
  - управление камерами;
  - модели самолётов/авто.

3. **Практическая работа «Управление одним сервомотором» (45 мин)**
   - **этап 1. Подключение сервомотора (5 мин):**
     - соединение сигнального, питания и земли (цвета проводов: оранжевый/жёлтый, красный, коричневый);
     - фиксация сервомотора на стенде.
   - **этап 2. Калибровка (10 мин):**
     - проверка диапазона движений (0°, 90°, 180°);
     - фиксация реальных углов (возможны отклонения ±5°).
   - **этап 3. Программирование базовых позиций (15 мин):**
     - создание программы для последовательного поворота:
       - 0° → ждать 2 сек → 90° → ждать 2 сек → 180°;
     - использование блоков «установить угол сервомотора».
   - **этап 4. Отладка и эксперименты (15 мин):**
     - изменение скорости поворота (добавление задержки между шагами);
     - тестирование плавности движения;
     - исправление ошибок (например, «застревание» на углах).
4. **Рефлексия и подведение итогов (20 мин)**
   - демонстрация работы сервомоторов (каждая группа показывает программу);
   - обсуждение:
     - «Почему сервомотор точнее обычного мотора?»,
     - «Как датчик обратной связи улучшает управление?»,
     - «Где ещё можно применить сервоприводы?»;
   - ответы на вопросы;
   - анонс следующего занятия (интеграция сервомотора в транспортную модель).

Домашнее задание:

1. Нарисовать схему подключения сервомотора к контроллеру (с подписями проводов и портов).
2. Составить алгоритм (6–8 команд) для сервомотора, который:
    - стартует в положении 0°;
    - плавно поворачивается до 120° за 3 сек;
    - ждёт 1 сек;
    - возвращается в 0° за 2 сек;
    - повторяет цикл 3 раза.
3. Найти 2 примера устройств (не роботов), где используется сервомотор. Указать их функции (3–4 предложения на пример).

Занятие 8 (2 часа). Интеграция сервомотора в транспортную модель

Цель: научиться комбинировать управление сервомотором с датчиками для решения прикладных задач транспортного робота.

Необходимое оборудование:

- собранные транспортные роботы с сервомоторами и датчиками (расстояния/линии);
- компьютеры с средой программирования;
- полигон с препятствиями и разметкой;
- рабочие листы с шаблонами для анализа.

Ход занятия (90 мин):

1. **Вводная часть (5 мин)**
    - проверка домашнего задания (обсуждение примеров устройств с сервомоторами, 3 мин);
    - постановка цели (2 мин).
2. **Теоретический блок: интеграция сервомотора и датчиков (10 мин)**
    - **варианты использования:**
        - сервопривод как захват груза (открытие/закрытие);
        - управление направлением движения (поворот колёс);
        - активация механизмов по сигналу датчика.
    - **алгоритмы синхронизации:**
        - «если датчик расстояния < 10 см → сервомотор 90° (захват)»;
        - «если линия обнаружена → сервомотор 0° (поворот)».

3. **Практическая работа: разработка программы (35 мин)**
    - **этап 1. Постановка задачи (5 мин):**
        - робот должен:
            - двигаться вперёд до препятствия (датчик расстояния);
            - остановиться, активировать сервомотор (захват груза);
            - развернуться, двигаться к зоне разгрузки;
            - опустить груз (сервомотор в исходное положение).
    - **этап 2. Блок-схема алгоритма (10 мин):**
        - составление последовательности действий (с условиями и циклами);
        - обозначение точек интеграции сервомотора и датчика.
    - **этап 3. Программирование (20 мин):**
        - реализация алгоритма в визуальной среде;
        - настройка углов сервомотора и чувствительности датчика;
        - добавление задержек для синхронизации действий.
4. **Практическая работа: испытания и анализ (30 мин)**
    - **этап 1. Тестирование (15 мин):**
        - запуск робота на полигоне;
        - фиксация сбоев (например, неточный захват, ложные срабатывания датчика).
    - **этап 2. Отладка (10 мин):**
        - корректировка углов сервомотора;
        - изменение порога срабатывания датчика;
        - оптимизация времени задержек.
    - **этап 3. Анализ (5 мин):**
        - заполнение таблицы:

Этап алгоритма	Ожидаемый результат	Фактический результат	Причина расхождений
Захват груза	Сервомотор 90°	Сервомотор 80°	Неточная калибровка

5. **Рефлексия и подведение итогов (10 мин)**
    - обсуждение:
        - «Как датчики улучшают работу сервопривода?»,

        - «Какие проблемы возникли при интеграции?»,
        - «Где ещё можно использовать сервомотор в транспорте?»;
    - ответы на вопросы;
    - обзор следующих тем (групповые проекты, сложные алгоритмы).

Домашнее задание:

1. Составить таблицу сравнения сервомотора и обычного мотора (5 критериев: точность, скорость, стоимость, сложность программирования, сферы применения).
2. Написать алгоритм (8–10 команд) для робота-доставщика, который:
    - движется по линии до перекрёстка;
    - останавливается, поднимает груз (сервомотор 120°);
    - продолжает движение до зоны разгрузки;
    - опускает груз (сервомотор 0°), издаёт звуковой сигнал.
3. Придумать задачу для робота с сервоприводом и датчиком расстояния (например, сортировка коробок по размеру). Описать алгоритм решения (5–6 шагов).

Критерии оценки практических работ:

- правильность подключения сервомотора и датчиков;
- работоспособность программы (выполнение всех этапов задачи);
- корректная синхронизация сервомотора с датчиками;
- аккуратность

Занятие 9 (2 часа). Профессии в робототехнике и старт группового проекта

Цель: познакомиться с профессиями в сфере робототехники; сформировать команды и заложить основу для реализации группового проекта.

Необходимое оборудование:

- мультимедийный проектор/экран;
- робототехнические конструкторы;
- компьютеры с ПО для программирования;
- бланки для планирования проекта (шаблоны: «Роль в команде», «План проекта», «Анализ ресурсов»);
- примеры успешных робототехнических проектов (презентации, видео).

Ход занятия (90 мин):

1. **Вводная часть (10 мин)**
    - приветствие, проверка готовности (2 мин);

- актуализация знаний: вопросы «Какие этапы разработки робота мы уже прошли?», «Что нужно для создания работающего прототипа?» (3 мин);
- постановка цели и задач (5 мин).

2. **Теоретический блок: профессии в робототехнике (20 мин)**
   - **мобильный робототехник:**
     - задачи: проектирование, сборка, тестирование мобильных роботов;
     - навыки: механика, программирование, работа с датчиками;
   - **робототехник в машиностроении:**
     - задачи: интеграция роботов в производственные линии, настройка манипуляторов;
     - навыки: CAD-моделирование, автоматизация, ТО оборудования;
   - **другие профессии:**
     - инженер-программист роботов (алгоритмы, ИИ);
     - техник-сборщик (монтаж, ремонт);
     - проектировщик сенсоров (разработка датчиков);
     - оператор робототехнических систем (управление, мониторинг).
   - **где работают:** заводы, НИИ, стартапы, логистические компании, космос.

Демонстрация: видеоинтервью с робототехниками, обзор кейсов (например, робот-складской погрузчик).

3. **Старт группового проекта (50 мин)**
   - **этап 1. Формирование команд (5 мин):**
     - 3–4 человека в группе;
     - обсуждение интересов (кто что хочет делать: механика, программирование, дизайн).
   - **этап 2. Определение проблемы и цели (10 мин):**
     - пример проблемы: «Нехватка автоматизированных решений для перемещения грузов в школах»;
     - цель: «Создать модель транспортного робота для доставки учебных материалов между кабинетами».
   - **этап 3. Распределение ролей (10 мин):**
     - конструктор (ответственный за сборку);
     - программист (написание кода);
     - тестировщик (проверка работы, поиск ошибок);
     - презентатор (подготовка доклада и демонстрации).

        - заполнение бланка «Роль в команде» (обязанности, зоны ответственности).
    - **этап 4. Планирование проекта (15 мин):**
        - этапы: исследование → проектирование → сборка → программирование → тестирование → доработка → защита;
        - сроки (график на 2 занятия);
        - ресурсы: конструкторы, ПО, полигон, справочные материалы.
    - **этап 5. Обоснование проекта (10 мин):**
        - ответы на вопросы:
            - «Почему это важно?»,
            - «Какие аналоги существуют?»,
            - «Чем наш робот будет лучше?».
        - запись в рабочий лист.
4. **Рефлексия и подведение итогов (10 мин)**
    - обсуждение:
        - «Какая профессия в робототехнике вам интересна и почему?»,
        - «Какие сложности могут возникнуть в проекте?»,
        - «Что нужно сделать до следующего занятия?»;
    - ответы на вопросы;
    - анонс следующего занятия (практическая реализация проекта).

Домашнее задание:

1. В команде: составить список необходимых деталей для робота (с указанием количества и функций).
2. Индивидуально: написать эссе (5–7 предложений) на тему «Моя роль в проекте: что я сделаю и чему научусь».
3. Найти 1 пример реального транспортного робота (не из урока), описать его функции и особенности конструкции (3–4 предложения).

Занятие 10 (2 часа). Реализация и защита группового проекта

Цель: завершить разработку модели транспортного робота; провести самооценку и защиту проекта.

Задачи:

- выполнить сборку и программирование робота согласно плану;
- провести испытания и устранить недочёты;
- подготовить материалы для защиты проекта;

- представить результаты работы, ответить на вопросы.

Необходимое оборудование:

- собранные на прошлом занятии заготовки (шасси, моторы, датчики);
- конструкторы и инструменты для доработки;
- компьютеры с ПО;
- полигон для тестирования (препятствия, разметка);
- шаблоны для самооценки и презентации (структура доклада, критерии оценки).

Ход занятия (90 мин):

1. **Вводная часть (5 мин)**
    - проверка домашнего задания (обсуждение списков деталей и эссе, 3 мин);
    - постановка цели (2 мин).
2. **Практическая работа: сборка и программирование (40 мин)**
    - **этап 1. Сборка модели (15 мин):**
        - монтаж шасси, моторов, датчиков;
        - крепление сервопривода (если используется для захвата);
        - проверка соединений.
    - **этап 2. Программирование (15 мин):**
        - реализация алгоритма движения (вперёд, назад, повороты);
        - интеграция датчиков (расстояния/линии) для навигации;
        - отладка кода (исправление ошибок, оптимизация скорости).
    - **этап 3. Тестирование (10 мин):**
        - запуск робота на полигоне (выполнение маршрута);
        - фиксация проблем (например, неточный поворот, сбои датчиков);
        - доработка конструкции/кода.
3. **Подготовка к защите проекта (20 мин)**
    - **этап 1. Оформление материалов (10 мин):**
        - презентация (5–7 слайдов):
            1. проблема и цель;
            2. этапы работы;
            3. роли в команде;
            4. фото/схемы робота;
            5. видео испытаний;

6. выводы.

- раздаточные материалы (описание функций, схема подключения).

- **этап 2. Репетиция выступления (10 мин):**
  - распределение времени (доклад — 5 мин, вопросы — 3 мин);
  - тренировка ответов на типовые вопросы («Почему выбрали такую конструкцию?», «Как робот решает проблему?»).

4. **Защита проекта (20 мин)**
   - выступления команд (по 8 мин на группу);
   - вопросы от одноклассников и учителя;
   - оценка по критериям (см. ниже).
5. **Самооценка и рефлексия (5 мин)**
   - заполнение бланка самооценки:
     - что получилось хорошо?;
     - какие были трудности?;
     - чему научились?;
     - как можно улучшить проект?
   - общее обсуждение: «Что было самым интересным в проекте?».

Критерии оценки проекта:

- **актуальность проблемы и обоснованность решения** (5 баллов);
- **качество сборки** (прочность, аккуратность, соответствие плану) — 10 баллов;
- **работоспособность робота** (выполнение задач без критических сбоев) — 15 баллов;
- **программирование** (логика, использование датчиков, оптимизация) — 10 баллов;
- **презентация** (структура, наглядность, ответы на вопросы) — 10 баллов;
- **командная работа** (равномерное участие, взаимопомощь) — 5 баллов;
- **самооценка** (анализ результатов, предложения по улучшению) — 5 баллов.

Максимальная сумма: 60 баллов.

Шкала оценок:

- 54–60 баллов — «отлично» (полный цикл разработки, креативность, чёткая защита);
- 45–53 балла — «хорошо» (работоспособный робот, небольшие недочёты в презентации);
- 36–44 балла — «удовлетворительно» (базовые функции реализованы, есть ошибки в коде/сборке);
- < 36 баллов — «требуется доработка» (критические сбои, отсутствие ключевых компонентов).