Тема 5 · Урок 5.2

Захват: берём и ставим

Робот подъезжает к грузу, берёт его захватом, увозит и ставит в нужном месте.

2 × 45 мин | 1–2 ученика на робота | Практика

РЕЗУЛЬТАТ УРОКА

Робот подъезжает к грузу по ультразвуку, берёт её, разворачивается, везёт по линии и ставит на другом конце. Действия оформлены функциями pickUp() и putDown().

Ты научишься

- Объяснить, как сервопривод поворачивается на заданный угол.
- Найти углы «открыто» и «закрыто» для своего захвата опытом.
- Соблюдать порядок: остановиться — закрыть; открыть — подождать — отъехать.
- Собрать перевозку груза из функций прошлых уроков.

Нужно знать

- 5.1 Ультразвук: видим груз
- 4.2 Повороты на перекрёстке и маршрут

Понадобится

- Захват с сервоприводом
- Ультразвуковой датчик расстояния
- Датчик линии (аналоговый, Trema) × 2

Зачем этот урок

Робот умеет доезжать до груза. Остался последний навык курьера — взять груз, довезти и аккуратно поставить. Сегодня робот соберёт всё вместе: линия, ультразвук и захват.

РАЗМИНКА · ВСПОМНИ

1. Какое условие остановки перед грузом правильное и почему?
2. Как робот разворачивается на линии?
3. Что делает `followLineFor(1500)`?

БЕЗОПАСНОСТЬ

Не держи пальцы в захвате, когда включено питание. Сервопривод может резко повернуться при загрузке программы.

ЗАНЯТИЕ 1 · 45 МИН

1 Сервопривод и захват

1 Как работает сервопривод

8 мин

Что должен делать робот. Курьер должен взять груз, удержать его при перевозке и отпустить в нужном месте. Для этого добавим механический захват с сервоприводом. Ультразвук помогает подъехать, но сам ничего не захватывает.

Что используем и как это работает. Внутри сервопривода есть мотор, редуктор, датчик положения и управляющая схема. По управляющим импульсам он получает заданное положение вала и старается его удержать. Библиотека Servo формирует эти импульсы; gripper.write задаёт положение, а не скорость вращения. Рычаг превращает поворот вала в движение губок захвата.

Как связать измерение с действием. Углы «открыто» и «закрыто» подбираем для своей механики небольшими изменениями; чужие числа могут упереть механизм в ограничитель. Сначала останавливаем колёса, затем закрываем захват и даём ему время. Команда закрытия не доказывает, что груз удерживается: это проверяем наблюдением. Подключение и питание сверяем с подтверждённой схемой комплекта.

Сервопривод поворачивается на заданный угол и держит его: внутри мотор, шестерёнки и датчик положения.

1. Посмотри контакт сервопривода на карточке у робота и впиши его в скетч пробы угла.
2. Сделай опыт: `gripper.write(40)`, пауза, `gripper.write(90)`, пауза. Пробуй паузы 100, 300 и 600 мс. При какой паузе захват успевает дойти до угла?
3. Запиши, сколько примерно времени нужно сервоприводу на 50° и почему программа должна ждать после команды.

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

В журнале — опыт с паузами и наименьшая пауза, при которой захват доходит до угла.

2 Найди углы «открыто» и «закрыто»

22 мин

Посмотри контакт сервопривода на карточке у робота.

Требование 1. Скетч `L52_Servo_Test` для подбора углов: угол вводится числом в монитор порта, и захват поворачивается сразу, без новой загрузки программы.

- Принимаются только числа от 0 до 180; на другое число программа отвечает, что так нельзя, и захват не двигается.
- После каждого поворота программа печатает, какой угол сейчас.
- Без номера контакта программа не трогает сервопривод.

Как это устроено: Справка по командам

Команда	Что делает
`#include <Servo.h>`, `Servo gripper;`	Подключить библиотеку и создать сервопривод
`gripper.attach(pin)`	Связать сервопривод с контактом
`gripper.write(angle)`	Повернуть на угол 0...180
`Serial.available()`	Сколько символов пришло из монитора порта
`Serial.parseInt()`	Прочитать из пришедших символов целое число

Требование 2. Углы «открыто» и «закрыто» найдены с запасом и проверены:

- «открыто» — груз входит в захват, даже если робот остановился на 1 см левее или правее середины;
- «закрыто» — найди угол, где захват упирается в груз, и отступи от него так, чтобы сервопривод не гудел; запиши, на сколько градусов отступил;
- груз не выпадает, когда робота с грузом наклоняют на 30° вперёд и назад.

Требование 3. Функция `moveGripper(target)` поворачивает захват к углу плавно — шагами по 2° — а не одним рывком. Сравни, как ведёт себя груз при резком и плавном закрытии.

Подсказка: Идея (1 из 2)

Помни текущий угол в переменной. Пока он не равен цели, сдвигай его на шаг в сторону цели и делай короткую паузу.

Подсказка: Фрагмент кода (2 из 2)

`currentAngle += (target > currentAngle) ? 2 : -2;` — что будет, если цель отличается на нечётное число градусов?

Подсказка: В мониторе порта ничего не происходит

Проверь скорость 9600 и что монитор отправляет строку по Enter.

ДОЛЖНО ПОЛУЧИТЬСЯ

Скетч подбора углов выполняет требование 1. В журнале — углы, запас «закрыто» в градусах, проверка наклоном и сравнение резкого и плавного закрытия. `moveGripper` работает.

3 Порядок действий важен

15 мин

НАЙДИ ОШИБКУ · ГРУЗ УЕЗЖАЕТ С РОБОТОМ

У соседей `putDown()` такая: `gripper.write(OPEN_ANGLE);` и сразу `drive(BACKWARD, …)`. Груз то падает, то уезжает вместе с роботом. Почему?

1. Запиши правила порядка для «взять» и «поставить».
2. Найди опытом наименьшую паузу после открытия, при которой груз остаётся на месте: 100, 200, 400, 600 мс, по три раза каждую.
3. Какую паузу ты возьмёшь в программу и почему не самую короткую из удачных?

ДОЛЖНО ПОЛУЧИТЬСЯ

Правила записаны, таблица опыта и выбранная пауза с обоснованием.

ЗАНЯТИЕ 2 · 45 МИН

2 Перевозка груза

4 Взять и поставить

20 мин

Требование 4. Продолжи свою программу `L51_Stop_At_Can` и сохрани как `L52_Pick_Place`. Робот выполняет доставку по линии:

- `pickUp()` и `putDown()` соблюдают порядок: остановиться — закрыть; открыть — подождать — отъехать;
- захват двигается твоей функцией `moveGripper`;
- без контакта сервопривода программа не начинает движение;
- в `setup()`: подъехать к грузу, взять, развернуться на линии, проехать с грузом, поставить.

Приёмка. Груз в конце линии, робот — на другом конце: доставка без помощи рук.

Подсказка: Идея (1 из 2)

Каждое действие доставки — уже готовая функция из прошлых уроков или этого урока. Осталось соединить их по порядку.

Подсказка: План (2 из 2)

1. `followLineToCan(STOP_CM)`.
2. `pickUp()`.
3. `turnAroundOnLine()`.
4. Проехать по линии `CARRY_MS`.
5. `putDown()`.

ДОЛЖНО ПОЛУЧИТЬСЯ

Программа выполняет требования 1–4, доставка проходит без помощи рук.

5 Надёжность

12 мин

Измерь надёжность доставки и улучши её **по данным**.

1. Сделай 10 доставок подряд и отметь для каждой, на каком этапе она сорвалась: подъезд, захват, разворот, перевозка, постановка.
2. Посчитай, сколько доставок из 10 прошли. Какой этап даёт больше всего срывов?
3. Исправь только этот этап и снова сделай 10 доставок. Стало лучше? На сколько?

Серия	Успешно из 10	Подъезд	Захват	Разворот	Перевозка	Постановка
До						
После						

Подсказка: Если срывов нет

Усложни условия: поставь груз на 1 см в сторону от линии или подними `baseSpeed`. Где сорвётся первым?

ДОЛЖНО ПОЛУЧИТЬСЯ

Две серии по 10 доставок, этап с больше всего срывов, что изменено и насколько выросла надёжность.

6 Задания по уровням

13 мин

★★ **ПРОДВИНУТЫЙ УРОВЕНЬ · ФУНКЦИЯ DELIVER**

Напиши функцию `deliver(маршрут до груза, маршрут до места, время до места)`, которая выполняет всю доставку. Маршруты передавай строками решений, например `"LS"`. Тогда в итоговой задаче доставка будет одной строкой программы.

ПРОВЕРЬ СЕБЯ

- [ ] Мои углы «открыто» и «закрыто» записаны в журнале.
- [ ] Я соблюдаю порядок: остановиться — закрыть; открыть — подождать — отъехать.
- [ ] Робот перевозит груз без помощи рук.

Проверь себя

★★ Продвинутый уровень

1. Что в этом вызове означает строка "LS"?

```
1 deliver("LS", "R", 1500);
```

- ○ Маршрут до груза: на первом перекрёстке налево, на втором прямо
- ○ Маршрут до места доставки
- ○ Скорость моторов

2. Зачем оформлять доставку функцией deliver()?

- ○ В итоговой задаче доставка будет одной строкой, а исправлять её нужно в одном месте
- ○ Без функции захват не закроется
- ○ Функция ускоряет моторы

3. currentAngle = 90, target = 41. Что произойдёт?

```
1 void moveGripper(int target) {
2   while (currentAngle != target) {
3     currentAngle += (target > currentAngle) ? 2 : -2;
4     gripper.write(currentAngle);
5     delay(15);
6   }
7 }
```

- ○ Цикл не закончится: угол перескакивает 41 (42 → 40 → 42 …)
- ○ Захват остановится на 42
- ○ Захват дойдёт ровно до 41

4. Серия из 10 доставок: срывов на подъезде — 0, на захвате — 1, на развороте — 4, при перевозке — 1, на постановке — 0. Что улучшать первым?

- ○ Всё сразу
- ○ Разворот: там больше всего срывов
- ○ Захват

5. После исправления разворота успешных доставок стало 7 из 10 вместо 4. Как убедиться, что помогло именно исправление?

- ○ Сделать ещё одну доставку
- ○ Никак: 7 больше 4, этого достаточно
- ○ Сравнить срывы по этапам: на развороте их должно стать меньше, на остальных — столько же

Отвечай на вопросы своего уровня. Нужно верных ответов: Начальный — 4 из 5; Средний — 4 из 5; Продвинутый — 4 из 5. Ответы проверит преподаватель или программа курса.

Итог

У робота есть все умения итоговой задачи: ехать, поворачивать, держаться линии, находить перекрёстки, видеть груз и перевозить его. Дальше — итоговый проект.

Как подтвердить результат

Результат опыта связывает условия, версию программы, наблюдение и вывод. Сравнение попыток с одним изменением помогает проверить причину, включая неудачные попытки. Отметка относится к уже выполненной практике; повтор ради отметки не нужен. Расхождение с прогнозом задаёт следующую проверяемую гипотезу. Модель, измерения настоящего робота и объяснение — разные виды доказательств.

Словарь урока

сервопривод	Мотор, который поворачивается на заданный угол и держит его.
угол	Положение рычага сервопривода в градусах, обычно от 0 до 180.
библиотека	Готовый набор команд, который подключают строкой #include. Servo — библиотека для сервоприводов.