Тема 1 · Урок 1.1

Собираем робота и оживляем плату

Собери робота и проверь каждый установленный узел диагностическим скетчем.

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

РЕЗУЛЬТАТ УРОКА

Собранный робот: оба мотора проверены в обе стороны и останавливаются, оба датчика линии реагируют на белое и чёрное. Результаты проверки записаны в журнал.

Ты научишься

- Собрать ходовую часть робота по инструкции и проверить каждый узел.
- Различать перед, зад, левую и правую стороны робота.
- Загрузить программу в Arduino и объяснить, что делают setup() и loop().
- Проверить оба мотора, их остановку и оба датчика линии; отличать команду программы от наблюдаемого результата.

Нужно знать

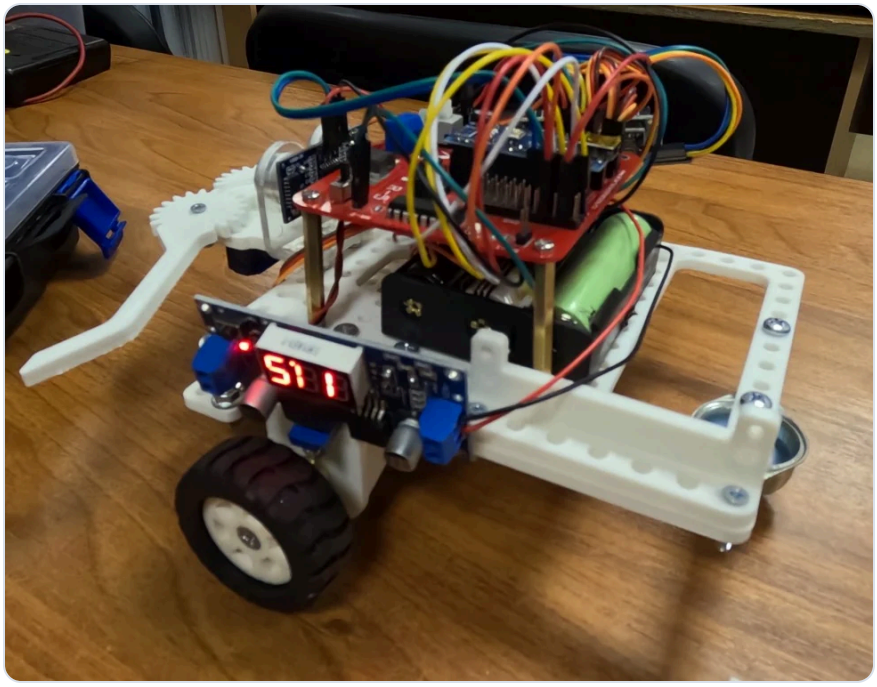
Ничего — это первый урок.

Понадобится

- Основа 130 × 150
- Кронштейн мотора
- Кронштейн мотора (зеркальный)
- Мотор-редуктор × 2
- Колесо × 2
- Шаровая опора
- Держатель Arduino
- Arduino Uno R3
- Шилд моторов L298P (Амперка)
- Планка датчиков
- Датчик линии (аналоговый, Trema) × 2
- Крепление выключателя
- Батарейный отсек моторов с выключателем
- USB-кабель для Arduino
- Крепёж: винты, гайки, шайбы, стойки
- Отвёртка

Зачем этот урок

Весь курс ты будешь работать с одним роботом — своим. Сегодня соберёшь его ходовую часть, установишь «мозг» робота — плату Arduino — и напишешь первую программу. К концу урока ты проверишь каждый установленный узел и запишешь, что он действительно работает.



Так робот будет выглядеть в конце курса. Сегодня собираем основу: платформу, моторы, колёса, плату и датчики линии.

БЕЗОПАСНОСТЬ

Пока идёт сборка, батарейный отсек отключён, а USB-кабель не подключён к компьютеру. Батарейный отсек к шилду заранее подключает взрослый. Моторы ты включаешь и выключаешь только выключателем.

ЗАНЯТИЕ 1 · 45 МИН

1 Ходовая часть

1 Распределите роли

3 мин

Что должен делать робот. Сначала роботу нужно правильно выполнять отдельные команды: вращать каждое колесо, останавливаться и сообщать показания датчиков. До самостоятельной поездки проверим каждый узел по очереди, чтобы искать причину ошибки в одном месте.

Что используем и как это работает. Arduino Uno выполняет программу. Драйвер L298P по её сигналам управляет питанием моторов: вывод DIR задаёт направление, PWM — долю времени подачи питания. Мотор превращает электрическую энергию во вращение, а колесо — в движение. Два датчика отражения снизу помогают различать светлую поверхность и чёрную линию.

Как связать измерение с действием. Команда и результат — разные вещи. Надпись в мониторе означает, что программа дошла до печати; вращение проверяем глазами. На подставке проверим оба направления каждого колеса и остановку, затем сравним показания датчиков над белой и чёрной карточками. Так узнаем, как именно собран наш робот.

Вы работаете вдвоём. Один — **Пилот**: держит детали и отвёртку. Второй — **Штурман**: читает шаг вслух, подаёт детали и проверяет результат по картинке.

1. Решите, кто начинает Пилотом.
2. После каждой детали, которая закреплена винтами, меняйтесь ролями.

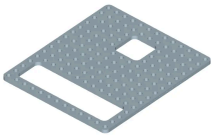
✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

Каждый знает свою роль. Штурман держит инструкцию, Пилот — инструмент.

2 Разложи детали

7 мин

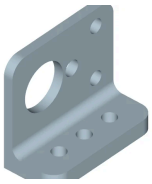
Прежде чем собирать, найди каждую деталь. Так ты не перепутаешь похожие детали и сразу увидишь, если чего-то не хватает.



Основа 130 × 150
1 шт.



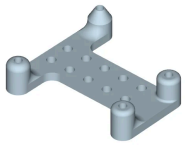
Кронштейн мотора
1 шт.



Кронштейн мотора (зеркальный)
1 шт.



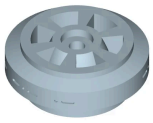
Планка датчиков
1 шт.



Держатель Arduino
1 шт.



Крепление выключателя
1 шт.



Колесо
2 шт.

1. Разложи печатные детали в ряд, как на карточках.
2. Найди два кронштейна моторов. Они похожи, но **зеркальные**: положи их рядом и посмотри — полочки смотрят в разные стороны.
3. Рядом положи покупные детали: Arduino, шилд моторов, два мотора, шаровую опору, два датчика линии, батарейный отсек и пакеты с крепежом.

ДОЛЖНО ПОЛУЧИТЬСЯ

Все детали лежат перед тобой. Ты можешь показать оба зеркальных кронштейна.

Подсказка: Не хватает детали

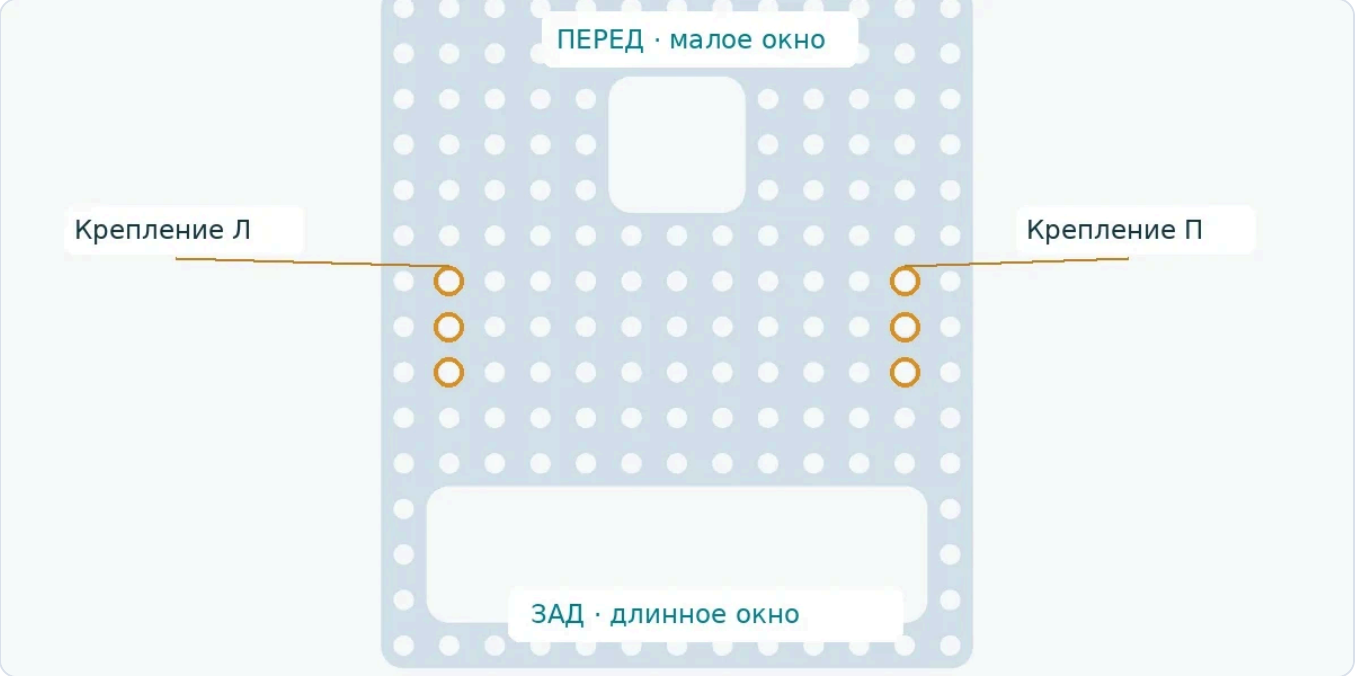
Не заменяй её похожей. Сверь пакет с карточками ещё раз; если детали нет — сообщи взрослому.

3 Потренируйся на одном винте

4 мин

Почти все детали робота держатся на винтах и гайках. Соберём одно соединение на столе.

1. Возьми винт, две шайбы и гайку из пакета «Тренировка».
2. Надень шайбу на винт, проведи винт через две детали, надень вторую шайбу и накрути гайку пальцами.
3. Придержи гайку пальцем и затяни винт отвёрткой. Остановись, как только детали перестали шевелиться.



Сетка отверстий основы. Через эти отверстия будем крепить все узлы.

ДОЛЖНО ПОЛУЧИТЬСЯ

Порядок соединения: винт → шайба → деталь → деталь → шайба → гайка. Детали не болтаются.

Как это устроено: Почему нельзя затягивать изо всех сил

Детали напечатаны из пластика. Если перетянуть винт, пластик продавится или треснет. Признак перетяжки — деталь прогибается или белеет вокруг винта.

4 Договоритесь, где перед

3 мин

Чтобы «левое колесо» всегда значило одно и то же, отметим перед робота.

1. Положи основу плоско: маленькое квадратное окно — дальше от тебя, длинное прямоугольное — ближе.
2. Дальняя сторона — **перед**. Наклей на неё бумажную стрелку «вперёд».
3. Лево и право определяй как водитель, который сидит в роботе и смотрит по стрелке.

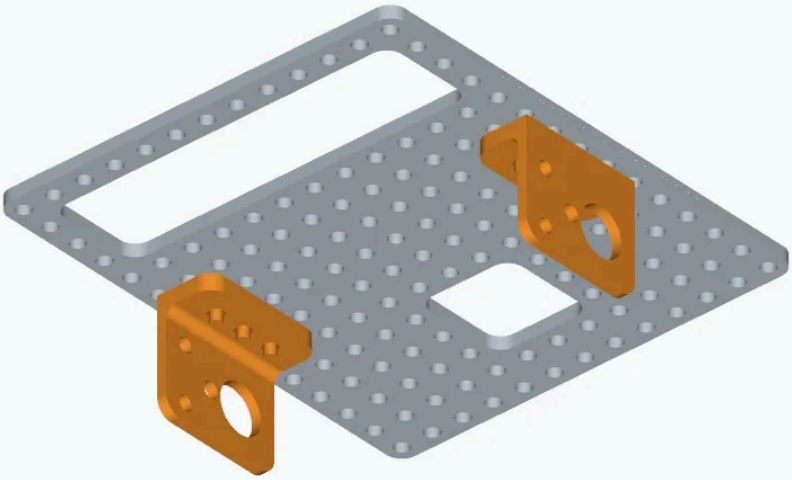
ДОЛЖНО ПОЛУЧИТЬСЯ

На основе есть стрелка «вперёд». Вы оба показываете левую сторону робота одинаково.

5 Прикрепи кронштейны моторов

8 мин

Кронштейны держат моторы снизу основы.



Вид снизу. Новые детали выделены оранжевым.

1. Переверни основу так, чтобы стрелка осталась видна сбоку.
2. Приложи кронштейны к нижней стороне, как на рисунке. Большие круглые отверстия смотрят **наружу** — туда будут выходить оси колёс.
3. Вставь винты сверху, снизу надень шайбы и гайки. Сначала наживи все винты пальцами, потом затяни.

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

Два кронштейна стоят друг напротив друга, большие отверстия — снаружи. Кронштейны не качаются.

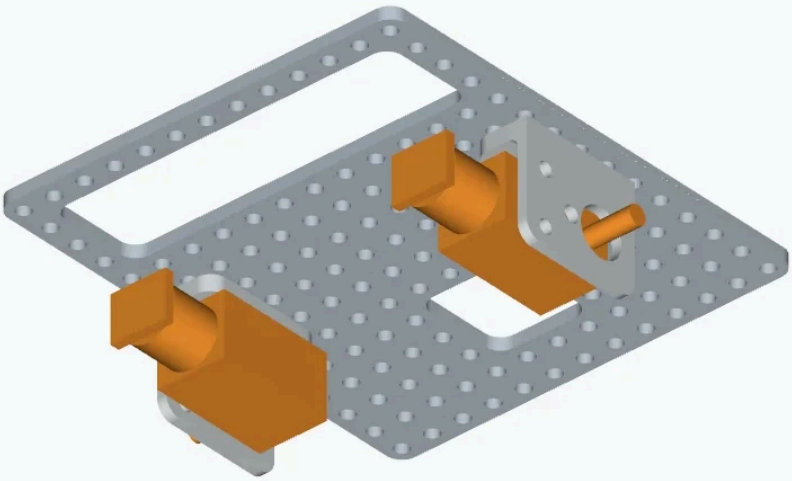
ПРОВЕРЬ СЕБЯ

- [ ] Оба кронштейна смотрят отверстиями наружу.
- [ ] Ни один винт не болтается.

6 Установи моторы

8 мин

Ось мотора — вращающийся стержень, на который надевается колесо. Она должна выйти наружу через отверстие кронштейна.



Корпус мотора внутри, ось — снаружи.

1. Приложи первый мотор к кронштейну изнутри. Ось проходит через большое отверстие наружу.
2. Закрепи мотор винтами из пакета «Моторы». Держи мотор рукой, пока затягиваешь.
3. Так же установи второй мотор.
4. Наклей на провода метки: **Л** — левый мотор, **П** — правый. Сверься со стрелкой «вперёд».

ДОЛЖНО ПОЛУЧИТЬСЯ

Оба мотора держатся крепко, оси смотрят влево и вправо, провода подписаны.

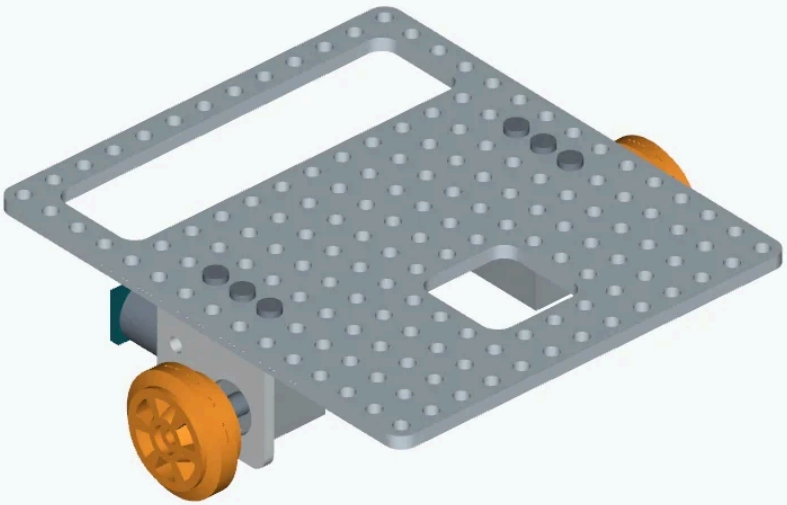
Подсказка: Мотор не встаёт в кронштейн

Не дави и не подпиливай. Проверь, тот ли это кронштейн: может быть, ты взял зеркальный для другой стороны.

7 Надень колёса

5 мин

1. Совмести отверстие колеса с осью мотора. Форма отверстия должна совпасть с формой оси.
2. Надень колесо, придерживая мотор за корпус. Не бей по колесу.
3. Посмотри сбоку: колесо не должно касаться кронштейна или основы.



Между колесом и кронштейном виден зазор.

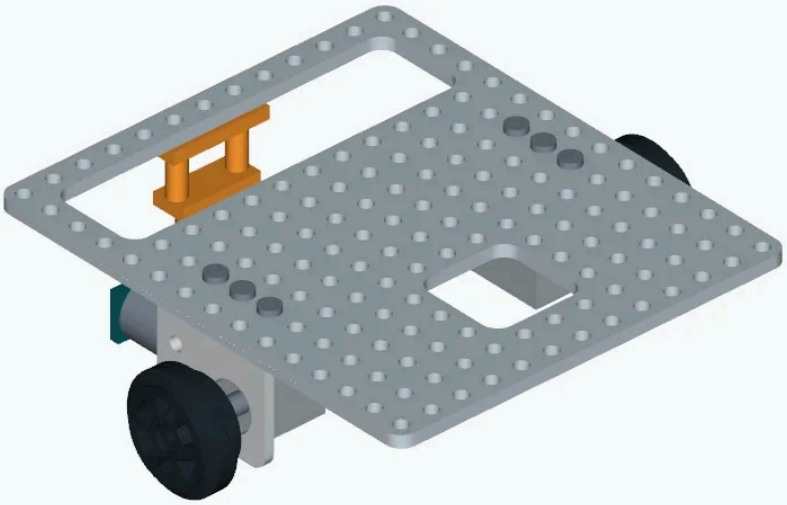
ДОЛЖНО ПОЛУЧИТЬСЯ

Два колеса стоят на осях и ничего не задевают.

8 Добавь заднюю опору и проверь робота

7 мин

Колёс два, поэтому роботу нужна третья точка опоры сзади — шаровая опора.



Опора крепится снизу у задней стороны основы на стойках.

1. Закрепи шаровую опору снизу у задней стороны основы. Стойки из пакета «Опора» подобраны по высоте.
2. Поставь робота на стол. Он должен стоять ровно на двух колёсах и опоре.
3. Аккуратно покачай робота. Ничего не должно болтаться.

ДОЛЖНО ПОЛУЧИТЬСЯ

Робот стоит устойчиво и ровно. Ходовая часть готова.

ПРОВЕРЬ СЕБЯ

- [ ] Я могу показать перед, левый и правый мотор.
- [ ] Колёса ничего не задевают.
- [ ] Робот стоит ровно.

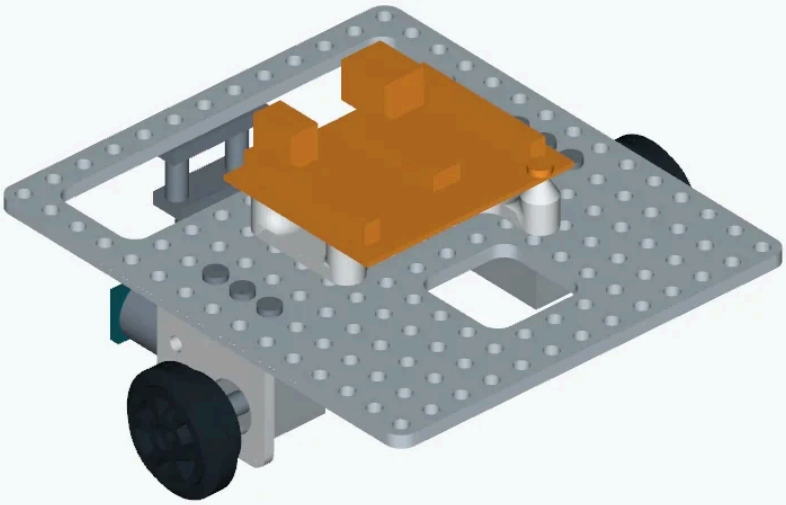
ЗАНЯТИЕ 2 · 45 МИН

2 Электроника и первая программа

9 Установи Arduino

6 мин

Arduino — «мозг» робота. Плата лежит на держателе, чтобы её контакты снизу не касались винтов.



Держатель и плата в центре основы.

1. Закрепи держатель Arduino в центре основы стойками вверх.
2. Положи плату на стойки так, чтобы разъём USB был у края и до него можно было дотянуться.
3. Закрепи плату винтами. Затягивай их по очереди, понемногу.

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

Плата лежит ровно и не касается винтов основы. Разъём USB свободен.

10 Подготовь компьютер

5 мин

1. Подключи USB-кабель к Arduino и к компьютеру. Батарейный отсек моторов остаётся отключённым.
2. В MyRobotica открой курс **«Робот-курьер»**, урок **1.1** и его учебный проект. В правой части кнопки **Загрузить на робота** выбери своего настоящего робота. Через **Подробнее о связи** проверь плату **Arduino Uno** и найди её порт.
3. Сохрани программу с пустыми `setup()` и `loop()`, нажми **Проверить код**, а потом **Загрузить на робота**.
4. Запиши в журнал: плату, порт, версию MyRobotica и чем **Проверить код** отличается от **Загрузить на робота** — что делает каждая кнопка и нужна ли для неё плата.

ДОЛЖНО ПОЛУЧИТЬСЯ

Пустой скетч проверен и загружен. В журнале — плата, порт, версия MyRobotica и разница двух кнопок.

Подсказка: Проверить код и Загрузить на робота

Отключи кабель и нажми обе кнопки по очереди. Какая сработала без платы?

11 Проверь загрузку программы

3 мин

Загрузи пустой скетч и прочитай сообщение об успешной загрузке.

Проверка связи с платой — L11_Empty.ino

```
1 void setup() {
2 }
3
4 void loop() {
5 }
```

Это проверка платы, USB-связи и загрузки. Она ещё ничего не говорит о моторах, шилде, датчиках и качестве механической сборки. Работоспособность проверим после подключения всех узлов.

12 Надень шилд и подключи моторы

7 мин

Выводы Arduino не могут дать моторам достаточно тока. Поэтому моторы подключаются к **шилду** — плате, которая надевается сверху на Arduino и включает моторы по её командам.

> ⚠ **БЕЗОПАСНОСТЬ**
> Сначала отключи USB-кабель. Ничего не подключай и не отключай, пока плата под питанием.

1. Совмести ножки шилда с гнёздами Arduino и аккуратно надави сверху. Ни одна ножка не должна остаться снаружи или согнуться.
2. Подключи провода моторов к клеммам шилда:

Что подключаем	Куда
Левый мотор Пометь провода буквой «Л».	Клеммы канала `M2` на шилде
Правый мотор Пометь провода буквой «П».	Клеммы канала `M1` на шилде
Батарейный отсек (через выключатель) ВЗРОСЛЫЙ Подключает взрослый.	Клеммы питания моторов на шилде

3. Батарейный отсек с выключателем уже подключён взрослым. Проверь, что выключатель выключен. Если провода отсека не подключены — сам их не подключай, позови взрослого.

> ✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**
> Шилд сидит на Arduino ровно. Провода моторов закреплены в клеммах и подписаны.

13 Установи датчики линии

6 мин

Датчики линии будут «смотреть» на пол и замечать чёрную линию. Ставим их спереди, поперёк робота.

1. Закрепи планку датчиков снизу у переднего края основы на стойках.
2. Закрепи на планке два датчика — левый и правый — чувствительной стороной **вниз**, к полу. Датчики на одинаковой высоте.
3. Подключи датчики к шилду по таблице. Буквы у контактов датчика: S — сигнал, V — питание, G — земля.

Что подключаем	Куда
Левый датчик · S (сигнал)	`A4`
Левый датчик · V (питание)	`5V`
Левый датчик · G (земля)	`GND`
Правый датчик · S (сигнал)	`A5`
Правый датчик · V (питание)	`5V`
Правый датчик · G (земля)	`GND`

4. Уложи провода так, чтобы они не касались колёс.

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

Два датчика смотрят вниз, их провода подключены и не свисают к колёсам.

14 Проверь все узлы робота

14 мин

Теперь проверим, что каждая часть робота работает: оба колеса и оба датчика. Программа для проверки уже открыта в редакторе справа.

> ⚠ **БЕЗОПАСНОСТЬ**
> Поставь робота на подставку так, чтобы колёса висели в воздухе и не касались стола. Пока выключатель моторов выключен, колёса крутиться не могут.

Напиши программу сам. В заготовке есть только номера контактов и выключение моторов.

Напиши свою диагностику L11_Diagnostics_Empty.ino

```
// Программа проверки для робота курса (Arduino Uno + шилд L298P).
// Робот стоит на подставке: колёса висят в воздухе.
// Колесо крутится со скоростью 120 из 255 около 0,3 секунды.
const int LEFT_DIR = 7;
const int LEFT_PWM = 6;
const int RIGHT_DIR = 4;
const int RIGHT_PWM = 5;
const int LEFT_SENSOR = A4;
const int RIGHT_SENSOR = A5;
const int FORWARD = HIGH;
const int BACKWARD = LOW;
const int TEST_PWM = 120;
const unsigned long PULSE_MS = 300;
bool running = false;
unsigned long startedAt = 0;

// Напиши диагностику по требованиям урока. При включении моторы стоят.
void setup() {
  pinMode(LEFT_PWM, OUTPUT);
  pinMode(RIGHT_PWM, OUTPUT);
  analogWrite(LEFT_PWM, 0);
  analogWrite(RIGHT_PWM, 0);
}

void loop() {
}
```

Что должна уметь программа:

- работать с монитором порта на скорости 9600;
- по букве `d` печатать подписанные числа обоих датчиков;
- по `1` и `2` крутить левое колесо вперёд и назад, по `3` и `4` — правое;
- каждый раз крутить со скоростью 120 не дольше 300 мс, потом выключать оба мотора;
- по букве `s` сразу выключать оба мотора, даже во время вращения;
- после включения и кнопки Reset моторы стоят, пока не придёт новая команда;
- лишние нажатия не копятся в очередь и не продлевают вращение.

> **Подсказка: Как обойтись без delay**
>
> Запомни время начала вращения через `millis()`. В каждом проходе `loop()` проверяй, не пришла ли `s` и не прошло ли 300 мс.

Как проверить робота

1. **Проверь провода.** Выключатель моторов выключен. Сверь провода с таблицами подключения: каждый на своём месте и не касается колёс. Колёса крутятся рукой свободно.
2. **Загрузи программу.** Нажми **Загрузить на робота**. Внизу открой **Serial Monitor** со скоростью 9600. Появится строка `READY` — плата отвечает. Нажми кнопку Reset на плате: колёса стоят.
3. **Проверь датчики.** Отправь `d` и прочитай два числа. Подложи под левый датчик белую бумагу и отправь `d`, потом чёрную — и снова `d`. Сделай так же с правым датчиком. На белом и чёрном числа должны заметно отличаться.
4. **Проверь колёса.** Включи выключатель моторов. Отправь `1` и смотри на колёса: коротко крутится только левое колесо вперёд и само останавливается. Потом по очереди отправь `2`, `3` и `4`.
5. **Проверь остановку.** Отправь `1` и сразу `s` — колесо останавливается. Выключи выключатель моторов и отправь `1` ещё раз: колёса не крутятся.
6. **Запиши результат** в журнал для каждого узла из таблицы: команду, ответ в Serial Monitor и фактическое вращение или показание датчика. Если узел ещё не проверен, так и запиши: «не проверен». Если что-то не так — выключи выключатель моторов, отключи USB, открой «Если не получилось», исправь и проверь этот узел ещё раз.

Что проверяем	Что сделать	Хорошо, если
Плата	Загрузить программу, нажать Reset	Есть `READY`, колёса стоят
Левое колесо	Отправить `1`, потом `2`	Крутится только левое: вперёд, потом назад
Правое колесо	Отправить `3`, потом `4`	Крутится только правое: вперёд, потом назад
Левый датчик	`d` на белом и на чёрном	Числа заметно разные
Правый датчик	`d` на белом и на чёрном	Числа заметно разные
Остановка	`s` и выключатель моторов	Колесо останавливается

🔧 ЕСЛИ НЕ ПОЛУЧИЛОСЬ

Что видишь	Что сделать (выключатель моторов выключен)
Крутится не то колесо	Поменяй местами провода моторов в клеммах M1 и M2
Колесо крутится назад вместо вперёд	Поменяй местами два провода этого мотора в клемме
Колесо не крутится	Проверь, что выключатель включён и провода плотно зажаты в клеммах
Число датчика не меняется	Проверь провода S, V, G и высоту датчика над полом
Нет строки `READY`	Проверь скорость 9600 и выбранный порт
Колесо крутится само после Reset	Проверь, что загружена именно эта программа
Ничего не помогает	Позови взрослого и покажи, на каком пункте остановился

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

Ты проверил каждое колесо в обе стороны, оба датчика и остановку. Результаты записаны в таблицу.

15 Запиши результат в журнал

4 мин

1. Запиши в бортовой журнал дату, имя робота и что получилось сегодня.
2. Отметь, какие узлы прошли проверку сразу, а какие ты исправлял.
3. Сохрани программу проверки — она пригодится, если робот начнёт вести себя странно.

ДОЛЖНО ПОЛУЧИТЬСЯ

Робот проверен, результаты записаны в журнал, программа проверки сохранена.

ПРОВЕРЬ СЕБЯ

- [ ] Я могу объяснить, чем `setup()` отличается от `loop()`.
- [ ] Я проверил оба колеса в обе стороны, остановку и оба датчика.
- [ ] Я записал результаты и прошёл проверку знаний.
- [ ] Я знаю, где у моего робота левый мотор и левый датчик.

Проверь себя

★★ Продвинутый уровень

1. Как обеспечить обработку s во время импульса?

- ○ Проверять время без блокирующего ожидания
- ○ Ждать весь импульс в delay
- ○ Отключить последовательный порт

2. Что делает скетч после Reset?

- ○ Ждёт новой команды с выключенными моторами
- ○ Повторяет последний импульс
- ○ Запускает все тесты автоматически

3. Что делать с накопленными командами движения?

- ○ Выполнить все подряд
- ○ Не создавать очередь запусков
- ○ Продлить текущий импульс

4. Можно ли без энкодеров автоматически подтвердить вращение?

- ○ Да, по значению PWM
- ○ Нет, нужна обратная связь или наблюдение человека
- ○ Да, по числу отправленных команд

5. Все программные проверки прошли, но колесо не крутится. Итог?

- ○ Робот исправен
- ○ Достаточно проверить светодиод
- ○ Узел не принят; нужна диагностика и повторная физическая проверка

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

Ты собрал ходовую часть робота, установил плату и датчики и написал первую программу. Ты подтвердил работу установленных узлов наблюдениями. Проверка знаний обязательна отдельно. В следующем уроке перейдёшь от отдельных проверок на подставке к управляемому движению.

Как подтвердить результат

Для каждого опыта сохраняй условия, версию программы, наблюдение и вывод. Сравнивая попытки, меняй одну причину за раз; неудачные попытки тоже остаются в журнале. Отметка результата относится к уже выполненным действиям, повторять их ради отметки не нужно. Если результат не получен, запиши расхождение и следующую проверяемую гипотезу. Модель, измерения настоящего робота и объяснение — разные виды доказательств.

Словарь урока

Arduino	Плата с микроконтроллером — маленьким компьютером, который выполняет твою программу.
шилд	Плата, которая надевается сверху на Arduino и добавляет ей возможности. Наш шилд управляет моторами.
скетч	Программа для Arduino.
функция	Именованный блок команд. setup и loop — функции.
светодиод	Маленькая лампочка на плате. Светодиод L подключён к выводу 13.
вывод	Контакт платы, которым программа может управлять или с которого может читать сигнал.
мс	Миллисекунда — тысячная доля секунды. 1000 мс = 1 секунда.