Тема 3 · Урок 3.1

Датчики линии и первое следование

Робот «смотрит» на пол, отличает чёрное от белого и сам держится линии.

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

РЕЗУЛЬТАТ УРОКА

Робот проезжает по чёрной линии с плавным изгибом, поворачивая по сигналам двух датчиков. В бортовом журнале записаны показания датчиков на белом и чёрном.

Ты научишься

- Прочитать показания датчика командой analogRead и вывести их в монитор порта.
- Различать строки Serial Monitor и каналы телеметрии с временем и единицами; сравнивать сохранённые показания модели и настоящего робота.
- Измерить показания на белом и чёрном и объяснить, почему у каждого датчика они свои.
- Записать правило «показание ближе к чёрному — значит, под датчиком линия».
- Составить два правила поворота и проверить их сначала на подставке, затем на дорожке.

Нужно знать

- 1.1 Собираем робота и оживляем плату
- 2.1 Моторы: едем и останавливаемся

Понадобится

Датчик линии (аналоговый, Trema) × 2

USB-кабель для Arduino

Зачем этот урок

До сих пор робот делал только то, что ты заранее записал: ехать столько-то, повернуть так-то. Сегодня он начнёт сам принимать решения. Два датчика будут смотреть на пол, и робот будет поворачивать, когда линия уходит в сторону.

РАЗМИНКА · ВСПОМНИ

1. Какая команда задаёт скорость мотора и какие числа она принимает?
2. Что делает `delay(500)`?
3. Какое колесо надо остановить, чтобы робот повернул налево?

ПРЕДСКАЖИ

Как робот может «увидеть» чёрную линию, если у него нет камеры? Обсудите с напарником и запишите догадку в бортовой журнал. В конце урока проверите.

БЕЗОПАСНОСТЬ

Пока меняешь провода, USB-кабель отключён, питание моторов выключено. Опыты с моторами — сначала на подставке: колёса не касаются стола.

ЗАНЯТИЕ 1 · 45 МИН

1 Датчики и их показания

1 Подготовь дорожку и робота

5 мин

Что должен делать робот. Чтобы ехать по дорожке, роботу нужно замечать, когда линия оказалась под левым или правым датчиком. Для этого используем два направленных вниз датчика отражения; они смотрят на пол, а не ищут предметы впереди.

Что используем и как это работает. В датчике есть излучатель и приёмник отражённого света. Поверхности отражают свет по-разному, а приёмник превращает отражённый свет в электрический сигнал. analogRead считывает его как число. Это не готовый ответ «чёрное»: сначала измеряем белое и чёрное для каждого датчика при рабочей высоте.

Как связать измерение с действием. Программа сравнивает чтение с измеренным порогом и выбирает действие колёс. Над линией слева и справа должны получиться разные команды коррекции. Проверим четыре сочетания карточек на подставке, затем движение по линии. Одинаковые показания обоих датчиков ещё не доказывают, что робот стоит правильно: оба могут оказаться и на белом, и на чёрном.

Датчику нужна понятная картинка: светлый фон и тёмная линия. Для заданий среднего уровня на дорожке понадобится и перекрёсток.

1. Возьми светлое матовое поле с чёрной линией или наклей чёрную матовую изоленту на белый лист: прямая, плавный изгиб и один перекрёсток.
2. Подготовь две карточки: белую и чёрную.
3. Найди на роботе левый и правый датчики. Поставь робота на линию и проверь: линия проходит между датчиками и не касается ни одного из них?

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

Дорожка с перекрёстком и карточки готовы. Линия помещается между датчиками.

2 Посмотри, что видит датчик

12 мин

Датчик светит на пол и меряет, сколько света вернулось. Плата превращает это в число, а чтобы увидеть его, программа отправляет число на компьютер.

Команда	Что делает
`analogRead(контакт)`	Показание датчика: число от 0 до 1023
`Serial.begin(9600);`	Включает связь с компьютером, один раз
`Serial.print(x);`	Печатает в той же строке
`Serial.println(x);`	Печатает и переходит на новую строку

1. Создай скетч `L31_Read_Sensors` по шаблону и напиши обе части сам. Левый датчик подключён к `A4`, правый — к `A5`.

Шаблон: показания двух датчиков L31_Read_Sensors_Gaps.ino

```
// Шаблон среднего уровня. Напиши код в разделе «Твоя часть» по целям в комме

const int LEFT_SENSOR_PIN = A4;
const int RIGHT_SENSOR_PIN = A5;

// ===== Твоя часть: напиши сам =====

void setup() {
  // Включи связь с компьютером на той же скорости, что в мониторе порта: 960
}

void loop() {
  // Пять раз в секунду печатай одну строку: L=число левого датчика  R=число
}
```

2. Загрузи программу и открой монитор порта со скоростью **9600**.
3. Поднеси под левый датчик белую карточку, затем чёрную. Держи карточку на той же высоте, на которой датчик стоит над полом.

ПРЕДСКАЖИ

Что напечатает монитор, если имя переменной в `Serial.print` взять в кавычки?

Подсказка: Текст и число в одной строке

Текст в кавычках печатается как есть, а имя переменной без кавычек заменяется её числом.

ДОЛЖНО ПОЛУЧИТЬСЯ

В мониторе бегут строки вида `L=… R=…` . Когда меняешь карточку под датчиком, его число заметно меняется.

ЕСЛИ НЕ ПОЛУЧИЛОСЬ

Что видишь	Что сделать
Вместо чисел непонятные символы	Скорость в мониторе порта и в программе должна совпадать: 9600.
Все числа в одной длинной строке	Нет перехода на новую строку.
Меняется не тот датчик	Провода левого и правого датчиков перепутаны.

3 Проверь оба датчика

8 мин

ПРЕДСКАЖИ

Будут ли числа левого и правого датчиков одинаковыми над одним и тем же белым полем?

1. Заполни таблицу для четырёх положений робота:

Положение	L	R
Оба над белым		
Чёрное только под левым		
Чёрное только под правым		
Линия точно между датчиками		

2. Почему в последней строке оба числа «белые», хотя робот стоит на линии? Запиши.

ДОЛЖНО ПОЛУЧИТЬСЯ

Таблица заполнена, прогноз проверен, ответ про положение «между датчиками» записан.

4 Измерь белое и чёрное

12 мин

Датчики не одинаковые: даже над одним полем их числа разные. Поэтому каждому датчику нужны свои измерения — **калибровка**.

1. Для каждого датчика запиши по три показания на белом и три на чёрном, в разных местах дорожки.
2. Выбери типичное число для каждой клетки и запиши, почему именно его.

Датчик	Белое: 3 числа	Белое	Чёрное: 3 числа	Чёрное
Левый				
Правый				

Подсказка: Числа в разных местах разные

Освещение и поле неодинаковые. Берут число, близкое к остальным, а не самое удачное.

ДОЛЖНО ПОЛУЧИТЬСЯ

В журнале таблица с повторами и четыре выбранных числа. Белое и чёрное заметно отличаются.

2

ЗАНЯТИЕ 2 · 45 МИН

Робот едет по линии

5 Научи программу замечать линию

10 мин

Программе нужно правило: когда считать, что под датчиком линия? Сравни новое показание с двумя измерениями: **если оно ближе к чёрному, чем к белому, — под датчиком линия**.

ПРЕДСКАЖИ

Белое — 100, чёрное — 800, пришло число 650. Линия под датчиком или нет? Посчитай до запуска: сколько от 650 до чёрного и сколько до белого?

1. Создай скетч `L31_Line_Detect` по шаблону и впиши свои четыре числа вместо нулей.
2. Напиши `loop()` сам: решение «линия или нет» для каждого датчика и печать.

Шаблон: программа замечает линию L31_Line_Detect_Gaps.ino

```
// Шаблон среднего уровня. Напиши код в разделе «Твоя часть» по целям в комме

const int LEFT_SENSOR_PIN = A4;
const int RIGHT_SENSOR_PIN = A5;

// Впиши свои измерения из бортового журнала.
const int LEFT_WHITE = 0;
const int LEFT_BLACK = 0;
const int RIGHT_WHITE = 0;
const int RIGHT_BLACK = 0;

// ===== Твоя часть: напиши сам =====

void loop() {
  int leftValue = analogRead(LEFT_SENSOR_PIN);
  int rightValue = analogRead(RIGHT_SENSOR_PIN);

  // Для каждого датчика реши, линия ли под ним (правило «ближе к чёрному»):
  // Печатай строку L=… R=…, а если под датчиком линия — допиши «слева линия»
  // Пять строк в секунду.
}

// ===== Готовое из прошлого шага — не меняй =====

void setup() {
  Serial.begin(9600);
}
```

3. Загрузи программу, открой монитор порта и подводи чёрную карточку под датчики по очереди.
4. Ответь в журнале: что начнёт печатать программа, если в ней перепутать белое и чёрное местами? Почему?

Подсказка: Как измерить расстояние между числами

От 650 до 800 — это `800 - 650`. Но программа заранее не знает, какое число больше. Команда `abs(…)` убирает знак: `abs(650 - 800)` равно 150.

Подсказка: Да или нет

Ответ «да или нет» хранит тип `bool`. Команды в `if (…) { … }` выполняются, только если ответ «да».

ДОЛЖНО ПОЛУЧИТЬСЯ

«слева линия» и «справа линия» появляются, только когда под датчиком чёрное. Прогноз про 650 сошёлся, ответ про перепутанные числа записан.

НАЙДИ ОШИБКУ · «СЛЕВА ЛИНИЯ» НА БЕЛОМ ПОЛЕ

Команда соседей написала так:

```
bool leftOnLine = abs(leftValue - LEFT_BLACK) < abs(leftValue - RIGHT_WHITE
```

Иногда «слева линия» печатается, когда под левым датчиком белое поле. Найди ошибку и объясни, почему она проявляется не всегда.

6 Придумай, как поворачивать

7 мин

Робот едет вперёд, и линия должна оставаться между датчиками. Если линия оказалась под одним датчиком, робот уехал в сторону. Колесо пока умеет только ехать или стоять. Как вернуть робота к линии?

1. Поставь выключенного робота на линию. Сдвинь его так, чтобы линия оказалась под левым датчиком. Куда робот уехал? Куда ему повернуть? Какое колесо для этого остановить?
2. То же — для правого датчика.
3. Заполни таблицу в журнале:

Что видят датчики	Левое колесо	Правое колесо	Куда едет робот
Оба белое	вперёд	вперёд	прямо
Левый — линия	?	?	?
Правый — линия	?	?	?
Оба — линия	?	?	?

4. Объясни в журнале одной фразой, почему остановка одного колеса поворачивает робота.

Подсказка: Какое колесо останавливать

Вспомни урок 2.2: левое колесо стоит, правое едет. Вокруг какого колеса повернётся робот и в какую сторону?

ДОЛЖНО ПОЛУЧИТЬСЯ

Таблица заполнена, объяснение записано. Ты показал на выключенном роботе оба поворота.

7 Запиши правила в программу

10 мин

Соберём всё вместе: датчики, правило «ближе к чёрному» и моторы из урока 2.1. Колёса будут слушаться правил из твоей таблицы.

1. Создай скетч `L31_First_Line` по шаблону. Впиши свои четыре измерения. Настройка моторов и защита уже готовы.
2. Напиши `loop()` сам: перенеси свои условия из `L31_Line_Detect`, задай колёсам скорости по своей таблице правил и отправь их на моторы.

Шаблон: первое движение по линии L31_First_Line_Gaps.ino

```ino
// Шаблон среднего уровня. Напиши код в разделе «Твоя часть» по целям в комме

const int LEFT_SENSOR_PIN = A4;
const int RIGHT_SENSOR_PIN = A5;

// Впиши свои измерения из бортового журнала.
const int LEFT_WHITE = 0;
const int LEFT_BLACK = 0;
const int RIGHT_WHITE = 0;
const int RIGHT_BLACK = 0;
const int LEFT_DIR_PIN = 7;
const int LEFT_SPEED_PIN = 6;
const int RIGHT_DIR_PIN = 4;
const int RIGHT_SPEED_PIN = 5;
const int FORWARD = HIGH;
int speed = 100; // начальная скорость, подбираем на подставке

// ===== Твоя часть: напиши сам =====

void loop() {
  int leftValue = analogRead(LEFT_SENSOR_PIN);
  int rightValue = analogRead(RIGHT_SENSOR_PIN);

  // Реши, линия ли под каждым датчиком, — перенеси свои условия из L31_Line_
  // Колёса выполняют твою таблицу правил: линия под датчиком — его колесо ст
  // Колесо, которое правило остановило, снова едет, когда его датчик видит б
}

// ===== Готовое: настройка и защита — не меняй =====

void setup() {
  pinMode(LEFT_DIR_PIN, OUTPUT);
  pinMode(LEFT_SPEED_PIN, OUTPUT);
  pinMode(RIGHT_DIR_PIN, OUTPUT);
  pinMode(RIGHT_SPEED_PIN, OUTPUT);

  analogWrite(LEFT_SPEED_PIN, 0);
  analogWrite(RIGHT_SPEED_PIN, 0);
  digitalWrite(LEFT_DIR_PIN, FORWARD);
  digitalWrite(RIGHT_DIR_PIN, FORWARD);

```

```
  // Без измерений робот не поедет: программа остановится здесь.
  if (LEFT_WHITE == LEFT_BLACK || RIGHT_WHITE == RIGHT_BLACK) {
    while (true) {
    }
  }

  delay(2000); // время убрать руки от робота
}
```

ПРЕДСКАЖИ

Что случится, если задать скорость колёс один раз в `setup()`, а не в начале каждого повтора `loop()`?

3. Нажми **Проверить**.

Подсказка: Колесо остановилось и больше не едет

Когда правило снова разрешает колесу ехать? Откуда в каждом повторе берётся его скорость?

Подсказка: Правило колеса

`if (проверка) { … }` выполняет команды в скобках, только если проверка верна. Сверься со своей таблицей: какую переменную проверить и какую скорость поменять?

ДОЛЖНО ПОЛУЧИТЬСЯ

`loop()` написан, **Проверить** проходит без ошибок, ответ на прогноз записан в журнал.

Как это устроено: Зачем строка while (true)

Если ты забыл вписать измерения, белое и чёрное равны нулю, и программа не может отличить линию. Тогда она «застревает» в пустом бесконечном цикле, и робот не поедет. Это защита от случайного старта.

8 Проверь на подставке

6 мин

> ⚠ **БЕЗОПАСНОСТЬ**
> Загружай программу при выключенном питании моторов. Колёса не касаются стола.

1. Поставь робота на подставку, загрузи программу, включи питание моторов.
2. Проверь все четыре строки своей таблицы правил: подставляй карточки так, чтобы датчики видели нужное. Записывай, что делают колёса.

Что видят датчики	Ждал	Увидел
Оба белое		
Левый — линия		
Правый — линия		
Оба — линия		

3. Проверь свой прогноз про скорость, заданную в `setup()`. Выключи питание моторов.

> ✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**
> Таблица заполнена, все строки совпали с правилами. Прогноз проверен.

ЕСЛИ НЕ ПОЛУЧИЛОСЬ

Что видишь	Что сделать
Колёса не крутятся	Написан ли `loop()`? Вписаны ли измерения? Включено ли питание моторов?
Останавливается не то колесо	Перепутаны датчики или моторы. Проверь по монитору порта и по меткам Л/П.

9 Первый заезд по линии

8 мин

1. Выключи питание моторов, отключи USB-кабель.
2. Поставь робота на прямой участок: линия между датчиками. Включи питание. Будь готов выключить его, если робот уходит с дорожки.
3. Пройди дорожку три раза при `speed = 100`, потом три раза при 80 и при 130. Записывай, сколько раз робот прошёл изгиб.

speed	Прошёл из 3	Что видно
80		
100		
130		

4. Почему робот едет рывками? Запиши одной фразой.

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

Таблица заездов при трёх скоростях и объяснение рывков.

★ **СРЕДНИЙ УРОВЕНЬ · ЕДЕМ ЧЕРЕЗ ПЕРЕКРЁСТОК**

Сейчас, если оба датчика видят чёрное, робот останавливается. На среднем поле через перекрёстки нужно проезжать прямо. Измени программу так, чтобы при чёрном под обоими датчиками оба колеса ехали вперёд. Проверь на дорожке с одним перекрёстком.

Проверь себя

★ Средний уровень

1. **На среднем поле оба датчика видят чёрное. Что должен делать робот по заданию ★?**

- ○ Ехать прямо: оба колеса вперёд
- ○ Остановиться
- ○ Развернуться

2. **Какое условие означает перекрёсток?**

- ○ leftOnLine && rightOnLine
- ○ leftOnLine || rightOnLine
- ○ !leftOnLine && !rightOnLine

3. **У датчика белое — 900, чёрное — 120. Он показывает 300. Что решит условие «ближе к чёрному, чем к белому»?**

- ○ Решить нельзя: у датчика чёрное меньше белого
- ○ Под датчиком линия: до чёрного 180, до белого 600
- ○ Под датчиком белое поле: 300 — маленькое число

4. В программе перепутали LEFT_WHITE и LEFT_BLACK местами. Что будет печатать программа?

- ○ То же, что раньше: abs уберёт разницу
- ○ Программа не загрузится
- ○ «слева линия» над белым полем и ничего над чёрным

5. Почему после остановки левого колеса робот поворачивает налево?

- ○ Правое колесо едет и поворачивает робота вокруг стоящего левого
- ○ Левое колесо начинает крутиться назад
- ○ Робот разворачивается по инерции

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

Робот научился сам выбирать, какое колесо остановить, по показаниям двух датчиков. Мы не угадывали числа, а измерили их на своём поле. Рывки при движении — цена простого правила «едет или стоит». В уроке 3.2 робот научится поворачивать плавно: тем сильнее, чем дальше линия ушла в сторону.

Вернись к своей догадке из начала урока. Подтвердилась ли она?

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

датчик линии	Датчик, который светит на пол и измеряет, сколько света отразилось. Белое отражает много, чёрное — мало.
показание	Число, которое программа получает от датчика. У нашей платы это число от 0 до 1023.
монитор порта	Панель Serial Monitor в MyRobotica, где видны сообщения, которые плата отправляет на компьютер.
калибровка	Настройка программы по измерениям своего робота и своего поля.
переменная	Именованное место в памяти, где программа хранит число.
условие	Проверка if: команды в фигурных скобках выполняются, только если проверка верна.
телеметрия	Передача именованных измерений с временем и единицами, чтобы строить графики и сохранять наблюдения.
АЦП	Аналого-цифровое преобразование: напряжение на входе платы превращается в число от 0 до 1023.