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.

1. Возьми светлое матовое поле с чёрной линией (или наклей чёрную матовую изоленту на белый лист). Для начала хватит прямой с одним плавным изгибом.
2. Подготовь две карточки: белую и чёрную.
3. Найди на роботе левый и правый датчики. Они смотрят вниз и стоят по обе стороны от линии, когда робот стоит на ней.

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

Дорожка и карточки готовы. Ты можешь показать левый и правый датчик.

2 Посмотри, что видит датчик

12 мин

Датчик светит на пол и меряет, сколько света вернулось. Плата превращает это в число.

Чтобы увидеть число, плата будет отправлять его на компьютер, а мы откроем **монитор порта**.

1. Создай скетч `L31_Read_Sensors` и набери программу. Левый датчик подключён к `A4`, правый — к `A5`.

Показания двух датчиков — L31_Read_Sensors.ino

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

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

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

  Serial.print("L=");
  Serial.print(leftValue);
  Serial.print("  R=");
  Serial.println(rightValue);

  delay(200);
}
```

2. Загрузи программу. Открой **Инструменты → Монитор порта** и выбери скорость **9600**.
3. Поднеси под левый датчик белую карточку, затем чёрную. Держи карточку на той же высоте, на которой датчик стоит над полом.

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

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

Как это устроено: Что значит каждая строка

- `analogRead(LEFT_SENSOR_PIN)` — прочитать показание датчика: число от 0 до 1023.
- `int leftValue = …` — сохранить число в переменную `leftValue`. `int` — целое число.
- `Serial.begin(9600)` — начать связь с компьютером. 9600 — скорость, такая же должна быть выбрана в мониторе порта.
- `Serial.print` печатает в той же строке, `Serial.println` — печатает и переходит на новую.
- Текст в кавычках печатается как есть, а имя переменной без кавычек заменяется её числом.

ЕСЛИ НЕ ПОЛУЧИЛОСЬ

Что видишь	Что сделать
Вместо чисел непонятные символы	В мониторе порта выбери 9600.
Число не меняется	Проверь провод сигнала S и номер входа в программе.
Меняется не тот датчик	Провода левого и правого датчиков перепутаны. Поменяй их местами или номера в программе.

3 Проверь оба датчика

8 мин

ПРЕДСКАЖИ

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

1. Поставь робота на белое поле. Запомни числа L и R.
2. Подведи чёрную карточку только под левый датчик, потом только под правый.
3. Сдвинь робота так, чтобы линия была точно между датчиками. Какие числа сейчас?

ДОЛЖНО ПОЛУЧИТЬСЯ

Ты знаешь, какое число у каждого датчика на белом и как оно меняется на чёрном. Когда линия между датчиками, оба показывают «белое».

4 Измерь белое и чёрное

12 мин

Датчики не одинаковые: даже над одним полем их числа разные. Поэтому каждому датчику нужны свои измерения. Такая настройка по своим измерениям называется **калибровкой**.

1. Поставь робота на белое поле. Посмотри на несколько строк и запиши типичное число левого датчика в бортовой журнал: «Левый, белое».
2. Подведи под левый датчик чёрную линию. Запиши «Левый, чёрное».
3. Повтори для правого датчика.
4. Проверь себя: повтори каждое измерение ещё раз. Числа должны быть близки к записанным.

Датчик	Белое	Чёрное
Левый		
Правый		

ДОЛЖНО ПОЛУЧИТЬСЯ

В журнале четыре числа. Белое и чёрное у каждого датчика заметно отличаются.

Подсказка: Белое и чёрное почти одинаковые

Датчик может стоять слишком высоко или линия блестит. Опусти датчики ближе к полу и проверь линию в другом месте поля, где нет бликов.

Как это устроено: Почему у одних датчиков чёрное — большое число, а у других — маленькое

Это зависит от устройства датчика. Одни выдают тем большее число, чем больше света вернулось, другие — наоборот. Поэтому мы не запоминаем «чёрное — это много», а измеряем свои числа.

2

ЗАНЯТИЕ 2 · 45 МИН

Робот едет по линии

5 Научи программу замечать линию

10 мин

Как программе решить, что под датчиком линия? Сравним новое показание с нашими двумя измерениями: **если число ближе к чёрному, чем к белому, — под датчиком линия**.

Пример: белое — 100, чёрное — 800. Пришло число 650. До чёрного 150, до белого 550. Ближе к чёрному — значит, линия.

1. Создай скетч `L31_Line_Detect`. Впиши свои четыре числа вместо нулей.

Программа замечает линию L31_Line_Detect.ino

```ino
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;

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

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

  bool leftOnLine = abs(leftValue - LEFT_BLACK) < abs(leftValue - LEFT_WHITE)
  bool rightOnLine = abs(rightValue - RIGHT_BLACK) < abs(rightValue - RIGHT_W

  Serial.print("L=");
  Serial.print(leftValue);
  Serial.print("  R=");
  Serial.print(rightValue);

  if (leftOnLine) {
    Serial.print("  слева линия");
  }
  if (rightOnLine) {
    Serial.print("  справа линия");
  }

  Serial.println();
  delay(200);
}
```

2. Загрузи программу, открой монитор порта и подводи чёрную карточку под датчики по очереди.

ДОЛЖНО ПОЛУЧИТЬСЯ

Когда под левым датчиком чёрное, в строке появляется «слева линия», под правым — «справа линия».

НАЙДИ ОШИБКУ · ПОЧЕМУ СООБЩЕНИЕ НЕ ПОЯВЛЯЕТСЯ

Команда соседей написала так:

```
bool leftOnLine = abs(leftValue - LEFT_BLACK) > abs(leftValue - LEFT_WHITE)
```

Под левым датчиком чёрная карточка, а «слева линия» не печатается. Найди ошибку и объясни, что программа проверяет сейчас.

Как это устроено: Что такое abs и bool

- `abs(число)` убирает знак: `abs(-150)` равно 150. Так мы получаем **расстояние** между двумя числами, не думая, какое из них больше.
- `bool` — переменная, в которой хранится не число, а ответ «да» или «нет» (`true` или `false`).
- `if (leftOnLine) { … }` — если ответ «да», выполнить команды в скобках.

6 Придумай, как поворачивать

7 мин

Робот едет вперёд. Линия должна оставаться между датчиками. Что делать, если линия оказалась под левым датчиком? Значит, робот уехал вправо, и ему надо повернуть налево.

Самый простой способ повернуть налево — **остановить левое колесо**, а правое пусть едет.

1. Поставь выключенного робота на линию. Сдвинь его так, чтобы линия оказалась под левым датчиком. Покажи, какое колесо нужно 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	?	?	?

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

Таблица заполнена. Ты можешь объяснить, почему остановка одного колеса поворачивает робота.

7 Запиши правила в программу

10 мин

Соберём всё вместе: датчики, наше правило «ближе к чёрному» и моторы из урока 2.1.

1. Создай скетч `L31_First_Line`. Перенеси свои четыре измерения.

Первое движение по линии L31_First_Line.ino

```ino
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

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
}

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

```

```
  bool leftOnLine = abs(leftValue - LEFT_BLACK) < abs(leftValue - LEFT_WHITE)
  bool rightOnLine = abs(rightValue - RIGHT_BLACK) < abs(rightValue - RIGHT_W

  int leftSpeed = speed;
  int rightSpeed = speed;

  if (leftOnLine) {
    leftSpeed = 0;
  }
  if (rightOnLine) {
    rightSpeed = 0;
  }

  analogWrite(LEFT_SPEED_PIN, leftSpeed);
  analogWrite(RIGHT_SPEED_PIN, rightSpeed);
}
```

2. Найди в `loop()` два правила из твоей таблицы. Объясни напарнику, что делает каждое.
3. Нажми **Проверить**.

Как это устроено: Почему скорость задаётся заново в каждом повторе

В начале `loop()` обе скорости снова становятся равны `speed`. Если бы мы этого не делали, колесо, остановленное один раз, так бы и стояло, даже когда датчик снова увидит белое.

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.
2. Включи питание моторов. Оба колеса должны крутиться вперёд.
3. Подведи чёрную карточку под левый датчик: левое колесо остановится. Убери карточку: колесо снова поедет. Проверь правый датчик.
4. Выключи питание моторов.

> ✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**
> Каждое колесо останавливается, когда под его датчиком чёрное, и снова едет на белом.

ЕСЛИ НЕ ПОЛУЧИЛОСЬ

Что видишь	Что сделать
Колёса не крутятся	Проверь, что вписаны измерения и включено питание моторов.
Колесо крутится назад	Проверь `FORWARD` и подключение мотора.
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.
3. Включи питание и наблюдай. Будь готов выключить питание, если робот уходит с дорожки.
4. Попробуй проехать изгиб. Запиши в журнал, что получилось.

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

Робот едет вдоль линии и проходит плавный изгиб. Он двигается рывками — это нормально для такого способа.

● **ОСНОВНОЕ · ПРЯМАЯ И ПЛАВНЫЙ ИЗГИБ**

Робот проходит прямую и плавный изгиб по двум правилам и не теряет линию.

Проверь себя

● Основное

1. Почему мы измеряем белое и чёрное для каждого датчика отдельно?

- ○ Так требует Arduino
- ○ Датчики не одинаковые: над одним полем их числа отличаются
- ○ Чтобы программа работала быстрее

2. Белое — 100, чёрное — 800. Датчик показывает 300. Что под датчиком?

- ○ Линия: число ближе к чёрному
- ○ Белое поле: число ближе к белому
- ○ Нельзя сказать

3. Робот едет вперёд, линия оказалась под правым датчиком. Что сделает наша программа?

- ○ Остановит левое колесо, робот повернёт налево
- ○ Остановит правое колесо, робот повернёт направо
- ○ Остановит оба колеса

4. Что напечатает этот код, если leftValue равно 512?

```
1 Serial.print("L=");
2 Serial.println(leftValue);
```

- ○ L=leftValue
- ○ L=512
- ○ "L="512

5. Зачем в начале loop() снова писать leftSpeed = speed и rightSpeed = speed?

- ○ Чтобы колесо поехало снова, когда датчик опять увидит белое
- ○ Чтобы робот ехал быстрее
- ○ Без этого программа не загрузится

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.