Тема 5 · Урок 5.1

Ультразвук: видим груз

Робот измеряет расстояние звуком и останавливается перед грузом, не путая «нет эха» с «груз рядом».

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

РЕЗУЛЬТАТ УРОКА

Робот едет по линии и останавливается на заданном расстоянии от груза. Если эхо не вернулось, программа не принимает это за «груз рядом».

Ты научишься

- Объяснить, как ультразвуковой датчик измеряет расстояние.
- Вывести расстояние в монитор порта и проверить его линейкой.
- Отличить отсутствие эха от маленького расстояния.
- Остановить робота перед грузом по измерению, а не по времени.

Нужно знать

- 3.2 П-регулятор: плавно и быстро

Понадобится

Ультразвуковой датчик расстояния | Датчик линии (аналоговый, Trema) × 2

Зачем этот урок

Груз стоит в конце ответвления. Робот должен подъехать к ней и остановиться ровно так, чтобы захват мог её взять. Если ехать «по времени», робот то не доедет, то врежется. Сегодня он научится видеть груз ультразвуком — так же, как летучая мышь видит препятствия.

РАЗМИНКА · ВСПОМНИ

1. Как робот останавливается на перекрёстке в программе из урока 4.1?
2. Что делает `break` в цикле? (Если не помнишь — разберём сегодня.)
3. Какие две команды нужны, чтобы робот ехал по линии с П-регулятором?

1

ЗАНЯТИЕ 1 · 45 МИН

Измеряем расстояние

1 Как работает ультразвук

8 мин

Что должен делать робот. Курьеру нужно обнаружить предмет впереди, оценить расстояние до него и остановиться перед грузом. Датчики линии смотрят вниз и эту задачу не решают. Добавим ультразвуковой датчик: он поможет измерить расстояние, но не отличит груз от стены.

Что используем и как это работает. Передатчик датчика излучает короткую серию ультразвуковых колебаний, которые человек не слышит. Звук отражается от предмета, а приёмник улавливает эхо. Arduino запускает измерение импульсом на TRIG и измеряет длительность сигнала ECHO. Это электрический сигнал о времени измерения, а не сам звук. Путь звука двойной: до предмета и обратно. Поэтому расстояние равно скорости звука, умноженной на время и делённой на два.

Как связать измерение с действием. В учебной программе время в микросекундах делится на 58, чтобы получить приблизительные сантиметры: около 29 мкс на сантиметр в одну сторону и столько же обратно. Например, 580 мкс соответствует примерно 10 см. Целочисленный результат отбрасывает дробную часть; температура и отражающая поверхность тоже влияют на измерение. Сравниваем его с линейкой от передней поверхности датчика.

Если измерение не получилось. Наклонная поверхность может отразить звук в сторону, мягкая — поглотить, а слишком близкий или далёкий предмет — не дать пригодного эха. Поэтому «нет эха» означает «расстояние неизвестно», а не ноль и не свободный путь. В коде для этого есть отдельное значение −1. Программа принимает решение об остановке только после проверки измерения. Базовый пример при −1 продолжает движение — это ограничение учебного алгоритма, которое разберём в задании на потерю эха.

Датчик посылает короткий щелчок ультразвука и слушает эхо. Чем дальше предмет, тем дольше звук летит туда и обратно.

ПРЕДСКАЖИ

Звук пролетает 1 см примерно за 29 микросекунд. Сколько времени пройдёт до возвращения эха, если предмет в 10 см? А в 30 см?

1. Запиши расчёт в журнал.
2. Объясни, почему время нужно умножать на два.

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

Расчёт для 10 и 30 см записан с объяснением про «туда и обратно».

2 Расстояние в мониторе порта

20 мин

1. Посмотри номера контактов TRIG и ECHO на карточке у робота. Карточки нет — проследи провода от выводов TRIG и ECHO датчика до платы и прочитай номера у гнёзд. Запиши в журнал.
2. Создай скетч `L51_Distance` по шаблону и впиши номера контактов вместо `-1`.
3. Напиши функцию `readDistanceCm()` сам. Печать в мониторе порта уже готова. Пока функция не написана, монитор печатает «нет эха».

Команда	Что делает
`digitalWrite(TRIG_PIN, HIGH);` … `LOW`	Щелчок: TRIG в HIGH на 10 мкс, перед этим коротко LOW
`delayMicroseconds(мкс);`	Ждёт указанное число микросекунд
`pulseIn(ECHO_PIN, HIGH, 25000)`	Время эха в микросекундах; 0 — эха нет за 25 000 мкс

Шаблон: расстояние L51_Distance_Gaps.ino

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

// Контакты ультразвукового датчика: впиши номера из таблицы проводки своего
const int TRIG_PIN = -1;
const int ECHO_PIN = -1;

// ===== Твоя часть: напиши сам =====

int readDistanceCm() {
  // Щёлкнуть датчиком и вернуть расстояние до груза в сантиметрах.
  // Если эхо не вернулось за 25 000 мкс, вернуть -1.
  return -1;
}

// ===== Готовое: настройка датчика и печать — не меняй =====

void setupSonar() {
  // Без контактов датчика программа дальше не пойдёт.
  if (TRIG_PIN < 0 || ECHO_PIN < 0) {
    while (true) {
    }
  }
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
}

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

void loop() {
  int distance = readDistanceCm();
  if (distance < 0) {
    Serial.println("нет эха");
  } else {
    Serial.print(distance);
    Serial.println(" см");
  }
  delay(200);
```

```
41 }
```

ПРЕДСКАЖИ

Эхо вернулось через 1160 мкс. Сколько сантиметров до груза?

4. Загрузи, открой монитор порта. Ставь груз на 5, 10, 20 и 30 см по линейке. Запиши показания.

По линейке	Показание датчика
5 см	
10 см	
20 см	
30 см	

Подсказка: На что делить

Звук летит туда и обратно: 29 мкс на сантиметр в одну сторону и 29 — в другую. Сколько микросекунд приходится на один сантиметр расстояния до груза?

Подсказка: Время эха — большое число

Время может быть больше, чем помещается в `int`. Для него берут `unsigned long`.

ДОЛЖНО ПОЛУЧИТЬСЯ

Показания близки к линейке, разница — 1–2 см. Прогноз для 1160 мкс сошёлся.

ЕСЛИ НЕ ПОЛУЧИЛОСЬ

Что видишь	Что сделать
Программа ничего не печатает	Вписаны ли номера TRIG и ECHO вместо −1?
Всегда «нет эха»	Есть ли щелчок перед ожиданием эха? Не перепутаны ли TRIG и ECHO?
Числа прыгают	Груз должен стоять прямо напротив датчика

3 «Нет эха» — не «рядом»

15 мин

1. Проверь датчик в трёх положениях и запиши, что печатает монитор:

Положение	Что печатает
В пустоту	
На мягкую ткань под углом	
Ладонь ближе 2 см	

НАЙДИ ОШИБКУ · РОБОТ ЗАКРЫЛ ЗАХВАТ В ПУСТОТЕ

Соседи написали остановку так: `if (readDistanceCm() <= 5) { stopMotors(); }`. Робот иногда останавливается посреди пустой дороги. Почему? Что вернёт функция, если эха нет?

2. Запиши правильное условие остановки.

ДОЛЖНО ПОЛУЧИТЬСЯ

Таблица заполнена, ошибка соседей объяснена, правильное условие записано.

2

ЗАНЯТИЕ 2 · 45 МИН

Стоп перед грузом

4 Едем к грузу и останавливаемся

17 мин

1. Создай скетч `L51_Stop_At_Can` по шаблону. Перенеси измерения, `kp`, `baseSpeed`, контакты датчика и замени `readDistanceCm` своей функцией.
2. Напиши движение в `setup()` сам: робот едет по линии и останавливается перед грузом. Помни шаг «Нет эха — не рядом».

Шаблон: стоп перед грузом L51_Stop_At_Can_Gaps.ino

```
// Шаблон среднего уровня. Напиши код в разделе «Твоя часть» по целям в комме
// Функцию readDistanceCm ниже замени своей из L51_Distance.

const int LEFT_SENSOR_PIN = A4;
const int RIGHT_SENSOR_PIN = A5;
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;
const int BACKWARD = (FORWARD == HIGH) ? LOW : HIGH;

// Измерения и настройки из журнала (уроки 3.1 и 3.2).
const int LEFT_WHITE = 0;
const int LEFT_BLACK = 0;
const int RIGHT_WHITE = 0;
const int RIGHT_BLACK = 0;
float kp = 0.5;
int baseSpeed = 100;
int leftLevel = 100;
int rightLevel = 100;

// Контакты ультразвукового датчика: впиши номера из таблицы проводки своего
const int TRIG_PIN = -1;
const int ECHO_PIN = -1;
const int STOP_CM = 5; // на каком расстоянии от груза остановиться

// ===== Твоя часть: напиши сам =====

void setup() {
  setupSonar();
  setupRobot();

  // Ехать по линии и остановиться, когда груз не дальше STOP_CM.
  // «Нет эха» — не «груз рядом»: тогда робот едет дальше.
}

// ===== Готовое из уроков 3.2 и 5.1 — не меняй =====

int level(int value, int white, int black) {
  long result = (long)(value - black) * 100 / (white - black);
```

```
  return constrain(result, 0, 100);
}

void readSensors() {
  leftLevel = level(analogRead(LEFT_SENSOR_PIN), LEFT_WHITE, LEFT_BLACK);
  rightLevel = level(analogRead(RIGHT_SENSOR_PIN), RIGHT_WHITE, RIGHT_BLACK);
}

void drive(int leftDir, int leftSpeed, int rightDir, int rightSpeed) {
  digitalWrite(LEFT_DIR_PIN, leftDir);
  digitalWrite(RIGHT_DIR_PIN, rightDir);
  analogWrite(LEFT_SPEED_PIN, leftSpeed);
  analogWrite(RIGHT_SPEED_PIN, rightSpeed);
}

void stopMotors() {
  analogWrite(LEFT_SPEED_PIN, 0);
  analogWrite(RIGHT_SPEED_PIN, 0);
}

void followLineStep() {
  int error = leftLevel - rightLevel;
  int correction = kp * error;
  drive(FORWARD, constrain(baseSpeed + correction, 0, 255),
        FORWARD, constrain(baseSpeed - correction, 0, 255));
}

void setupRobot() {
  pinMode(LEFT_DIR_PIN, OUTPUT);
  pinMode(LEFT_SPEED_PIN, OUTPUT);
  pinMode(RIGHT_DIR_PIN, OUTPUT);
  pinMode(RIGHT_SPEED_PIN, OUTPUT);
  stopMotors();
  // Без измерений робот не поедет.
  if (LEFT_WHITE == LEFT_BLACK || RIGHT_WHITE == RIGHT_BLACK) {
    while (true) {
    }
  }
  delay(2000);
}

// Расстояние в сантиметрах или -1, если эхо не вернулось.
int readDistanceCm() {
```

```
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN, LOW);
  unsigned long echoUs = pulseIn(ECHO_PIN, HIGH, 25000);
  if (echoUs == 0) {
    return -1;
  }
  return echoUs / 58;
}

void setupSonar() {
  // Без контактов датчика программа дальше не пойдёт.
  if (TRIG_PIN < 0 || ECHO_PIN < 0) {
    while (true) {
    }
  }
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
}

void loop() {
}
```

3. Поставь груз в конце линии. Запусти робота с другого конца.

Подсказка: Ехать, пока…

Как в `followLineToCrossing()` урока 4.1: в цикле измерить расстояние, решить, пора ли останавливаться, иначе сделать шаг по линии.

Подсказка: Две проверки

Остановиться можно, только если измерение есть и груз близко. Какое число вернёт твоя функция, если эха нет?

Как это устроено: Как выйти из цикла

`break` немедленно выходит из цикла `while`. Программа продолжается после цикла.

ДОЛЖНО ПОЛУЧИТЬСЯ

Робот едет по линии и останавливается перед грузом, не касаясь его, и не останавливается посреди пустой дороги.

5 Точная остановка

15 мин

Подбери `STOP_CM` и `baseSpeed` так, чтобы робот останавливался там, где удобно захвату. Меняй одно число за раз.

1. Реши, на каком расстоянии от груза роботу нужно стоять. Если захват уже стоит, измерь у него; если нет — выбери расстояние, при котором робот не касается груза, и уточни его на уроке 5.2. Запиши.
2. Запусти, измерь линейкой, где робот остановился. Запиши в таблицу и реши, что менять.

`STOP_CM`	`baseSpeed`	Где остановился, см	Что поменять и почему

3. Когда получилось, повтори три раза.

Подсказка: Робот останавливается ближе, чем STOP_CM

Робот не останавливается мгновенно: он ещё немного катится. Чем выше скорость, тем дальше.

ДОЛЖНО ПОЛУЧИТЬСЯ

Робот три раза подряд останавливается на нужном расстоянии. В журнале — таблица попыток с объяснением.

6 Задания по уровням

13 мин

★ **СРЕДНИЙ УРОВЕНЬ · ПОВТОР ПРИ ОТСУТСТВИИ ЭХА**

Если эха нет, измерь ещё раз. Если эха нет пять раз подряд — останови робота: что-то не так с датчиком. Так робот не уедет с поля, потеряв груз.

ПРОВЕРЬ СЕБЯ

- [ ] Я могу объяснить, как датчик измеряет расстояние.
- [ ] Я знаю, почему −1 нельзя принимать за «груз рядом».
- [ ] Мой робот останавливается перед грузом по измерению.

Проверь себя

★ Средний уровень

1. Эха нет пять раз подряд. Что делает робот по заданию ★?

- ○ Останавливается: возможно, с датчиком что-то не так
- ○ Едет дальше на полной скорости
- ○ Считает, что груз в 5 см

2. Измерение вернуло «нет эха». Что программа делает сначала?

- ○ Измеряет ещё раз
- ○ Закрывает захват
- ○ Поворачивает

3. **Эхо вернулось через 1160 мкс. Сколько сантиметров до груза?**

- ○ 40
- ○ 58
- ○ 20

4. **Почему время эха делят на 58, а не на 29?**

- ○ Звук летит туда и обратно: по 29 мкс на сантиметр в каждую сторону
- ○ Датчик удваивает время
- ○ 58 — скорость звука в сантиметрах

5. **На высокой скорости робот останавливается ближе к грузу, чем на низкой. Почему?**

- ○ STOP_CM зависит от скорости
- ○ После команды стоп робот ещё катится, и тем дальше, чем выше скорость
- ○ На скорости датчик хуже видит груз

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 видит груз и останавливается перед ним по измерению. Осталось научиться его брать — в следующем уроке.

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

ультразвук	Звук такой высоты, что человек его не слышит. Датчик посылает щелчок и ждёт эхо.
эхо	Звук, отразившийся от предмета и вернувшийся к датчику.
pulseIn	Команда, которая измеряет, сколько микросекунд длился сигнал на входе.
мкс	Микросекунда — миллионная доля секунды.
медиана	Среднее по величине из нескольких чисел: из 4, 25 и 5 это 5.