Тема 7 · Урок Ф.1

Энкодеры: считаем обороты и точное расстояние

Робот считает обороты колёс и проезжает отмеренное расстояние, а не «сколько успеет за время».

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

РЕЗУЛЬТАТ УРОКА

Программа считает обороты колёс энкодерами и проезжает отмеренное расстояние. Число импульсов на оборот измерено, точность проверена линейкой. Не нужен для итоговой задачи.

Ты научишься

- Объяснить, как энкодер превращает оборот колеса в импульсы.
- Измерить число импульсов на оборот колеса.
- Пересчитать расстояние в импульсы через длину окружности колеса.
- Проехать отмеренное расстояние и оценить точность линейкой.

Нужно знать

- 2.1 Моторы: едем и останавливаемся
- 3.2 П-регулятор: плавно и быстро

Понадобится

Колесо × 2 | Мотор-редуктор × 2 | USB-кабель для Arduino

Зачем этот урок

В уроке 2.1 робот ехал «по времени»: моторы включались на две секунды. Но сколько он проедет за две секунды, зависит от заряда батарей, пола и веса. **Энкодер** считает обороты колеса, поэтому программа может выбирать момент остановки по вращению колёс, а не только по времени. Точность пути всё равно нужно проверить линейкой.

Для урока нужен дополнительный модуль: энкодер на каждое колесо. В основной комплект курса он не входит; без него урок не провести. Факультатив не нужен для итоговой задачи.

РАЗМИНКА · ВСПОМНИ

1. Как в уроке 2.1 робот проезжал заданный участок?
2. Почему при тех же двух секундах робот с севшими батареями проезжает меньше?
3. Что такое П-регулятор из урока 3.2 в одной фразе?

ЗАНЯТИЕ 1 · 45 МИН

1 Считаем импульсы

1 Почему время — плохая линейка

8 мин

Что должен делать робот. Роботу нужно проехать заданное расстояние точнее, чем по секундомеру. Для этого понадобятся дополнительные энкодеры: в базовом комплекте их нет. Они измеряют вращение колёс, а не расстояние до препятствия.

Что используем и как это работает. Энкодер выдаёт импульсы при вращении. Счётчик накапливает их; число импульсов за оборот связывает счёт с оборотами колеса. Длина окружности связывает обороты с предполагаемым путём. Прерывание позволяет быстро увеличивать счётчик при импульсе, даже когда основная программа занята другой работой.

Как связать измерение с действием. Измерим импульсы на оборот и используем диаметр своего колеса. На AVR чтение многобайтного счётчика прерывание может разорвать посередине, поэтому копируем его при кратко выключенных прерываниях. Проверим результат линейкой: при проскальзывании колесо вращается, а корпус проходит меньше, поэтому энкодер не даёт абсолютной координаты.

1. Запусти программу «по времени» из урока 2.1 трижды: скорость 100, 150 и 200 при одном времени. Измерь пути линейкой.
2. Ответь в журнале: что, кроме скорости в программе, меняет путь за одно и то же время? Назови не меньше трёх причин.

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

Три пути измерены, причины записаны.

2 Как устроен энкодер

12 мин

Энкодер — диск с прорезями на оси колеса и оптический датчик щели: прорезь проходит мимо датчика — датчик даёт **импульс**. Считать импульсы будет **прерывание**: Arduino сама вызывает короткую функцию на каждый импульс, что бы ни делала основная программа.

1. Посчитай прорези на диске, запиши номера контактов энкодеров из таблицы проводки.

ПРЕДСКАЖИ

Колесо диаметром 45 мм, на диске 20 прорезей. Сколько миллиметров проезжает колесо между двумя импульсами? Посчитай до запуска.

2. Ответь в журнале: почему счётчик импульсов нельзя увеличивать в `loop()` после `delay(200)`?

БЕЗОПАСНОСТЬ

Не подключай энкодеры при включённом питании. Контакты не указаны в таблице — подключает преподаватель.

Подсказка: Сколько проезжает колесо за оборот

За один оборот колесо проходит длину своей окружности: π × диаметр.

ДОЛЖНО ПОЛУЧИТЬСЯ

Число прорезей, контакты, прогноз «мм на импульс» и ответ про `delay` — в журнале.

3 Импульсов на один оборот

22 мин

1. Создай скетч `LF1_Count` по шаблону. Впиши контакты, напиши свою часть.

Шаблон: счёт импульсов LF1_Count_Gaps.ino

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

// Контакты энкодеров: впиши номера из таблицы проводки своего робота.
// На Arduino Uno счёт по прерыванию работает только на контактах 2 и 3.
const int LEFT_ENCODER_PIN = -1;
const int RIGHT_ENCODER_PIN = -1;

// Счётчики импульсов. volatile: их меняет прерывание.
volatile long leftTicks = 0;
volatile long rightTicks = 0;

// ===== Твоя часть: напиши сам =====

void countLeft() {
  // Каждый импульс левого энкодера увеличивает его счётчик на 1.
}

void countRight() {
  // То же для правого энкодера.
}

void loop() {
  long left = 0;
  long right = 0;
  // Скопируй оба счётчика в left и right так, чтобы прерывание не изменило и

  Serial.print("Левое: ");
  Serial.print(left);
  Serial.print("  правое: ");
  Serial.println(right);
  delay(200);
}

// ===== Готовое: подключение энкодеров — не меняй =====

void setupEncoders() {
  // Без контактов энкодеров программа дальше не пойдёт.
  if (LEFT_ENCODER_PIN < 0 || RIGHT_ENCODER_PIN < 0) {
    Serial.println("Впиши контакты энкодеров");
    while (true) {
    }
```

```
  }
  pinMode(LEFT_ENCODER_PIN, INPUT);
  pinMode(RIGHT_ENCODER_PIN, INPUT);
  attachInterrupt(digitalPinToInterrupt(LEFT_ENCODER_PIN), countLeft, RISING)
  attachInterrupt(digitalPinToInterrupt(RIGHT_ENCODER_PIN), countRight, RISIN
}

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

2. Прокрути каждое колесо рукой ровно 10 оборотов по метке и посчитай импульсы на оборот.
3. Сравни с числом прорезей. Совпало? Если нет — найди причину.

Подсказка: Как не испортить число при чтении

Счётчик — `long`, это 4 байта, а Arduino читает их по одному. Прерывание посередине даст смесь старого и нового числа. Есть пара команд, которая на время выключает прерывания.

ДОЛЖНО ПОЛУЧИТЬСЯ

Счёт работает, импульсы на оборот обоих колёс записаны и сравнены с числом прорезей.

2

ЗАНЯТИЕ 2 · 45 МИН

Точное расстояние

4 Миллиметры в импульсы

15 мин

1. Создай скетч `LF1_Distance` по шаблону, впиши контакты и импульсы на оборот.
2. Напиши `ticksFor` и `averageTicks` по целям.

Шаблон: точное расстояние LF1_Distance_Gaps.ino

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

// Контакты энкодеров: впиши номера из таблицы проводки своего робота.
// На Arduino Uno счёт по прерыванию работает только на контактах 2 и 3.
const int LEFT_ENCODER_PIN = -1;
const int RIGHT_ENCODER_PIN = -1;

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 TICKS_PER_REV = 0; // импульсов на оборот колеса: измерь и впиши
const int WHEEL_DIAMETER_MM = 45;
const int DISTANCE_MM = 500; // сколько проехать
int speed = 100;

volatile long leftTicks = 0;
volatile long rightTicks = 0;

// ===== Твоя часть: напиши сам =====

long ticksFor(int mm, int ticksPerRev, int diameterMm) {
  // Сколько оборотов колеса нужно на mm миллиметров: один оборот — длина окр
  // Сколько это импульсов, округлённо до целого. Число PI в Arduino уже есть
  return 0;
}

long averageTicks() {
  // Среднее двух счётчиков. Читай их так, чтобы прерывание не изменило число
  return 0;
}

// ===== Готовое: энкодеры, моторы и поездка — не меняй =====

void countLeft() {
  leftTicks++;
}

void countRight() {
```

```
  rightTicks++;
}

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 setupEncoders() {
  // Без контактов энкодеров программа дальше не пойдёт.
  if (LEFT_ENCODER_PIN < 0 || RIGHT_ENCODER_PIN < 0) {
    Serial.println("Впиши контакты энкодеров");
    while (true) {
    }
  }
  pinMode(LEFT_ENCODER_PIN, INPUT);
  pinMode(RIGHT_ENCODER_PIN, INPUT);
  attachInterrupt(digitalPinToInterrupt(LEFT_ENCODER_PIN), countLeft, RISING)
  attachInterrupt(digitalPinToInterrupt(RIGHT_ENCODER_PIN), countRight, RISIN
}

void setup() {
  Serial.begin(9600);
  pinMode(LEFT_DIR_PIN, OUTPUT);
  pinMode(LEFT_SPEED_PIN, OUTPUT);
  pinMode(RIGHT_DIR_PIN, OUTPUT);
  pinMode(RIGHT_SPEED_PIN, OUTPUT);
  drive(FORWARD, 0, FORWARD, 0);
  setupEncoders();

  // Без числа импульсов на оборот робот не поедет.
  if (TICKS_PER_REV <= 0) {
    Serial.println("Впиши число импульсов на оборот");
    while (true) {
    }
  }
  delay(2000); // время убрать руки от робота

  long target = ticksFor(DISTANCE_MM, TICKS_PER_REV, WHEEL_DIAMETER_MM);
  drive(FORWARD, speed, FORWARD, speed);
  while (averageTicks() < target) {
```

```
  }
  drive(FORWARD, 0, FORWARD, 0);
  delay(1000); // робот докатывается

  Serial.print("Цель, импульсов: ");
  Serial.println(target);
  Serial.print("Насчитано после остановки: ");
  Serial.println(averageTicks());
}

void loop() {
}
```

ПРЕДСКАЖИ

Сколько импульсов вернёт `ticksFor(500, 20, 45)` ? Посчитай до запуска.

3. Запусти поездку от метки, измерь путь линейкой. Сравни с монитором порта: насколько «насчитано после остановки» больше цели?

Подсказка: Округление до целого

Если прибавить 0.5 и отбросить дробную часть, получится округление до ближайшего целого.

ДОЛЖНО ПОЛУЧИТЬСЯ

Прогноз совпал: 71. Робот проехал около 50 см, перекат после остановки виден в мониторе порта.

5 Почему робот проезжает дальше

15 мин

1. Сделай три заезда на 500 мм. Для каждого запиши путь линейкой и «перекат» в импульсах из монитора порта.
2. Добавь в программу поправку: останавливайся на `OVERSHOOT` импульсов раньше цели, где `OVERSHOOT` — средний перекат из трёх заездов. Сделай ещё три заезда.
3. Ответь в журнале: будет ли эта поправка верна на другой скорости? Почему?

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

Шесть заездов в журнале, поправка проверена, ответ про скорость записан.

6 Задания по уровням

12 мин

★ **СРЕДНИЙ УРОВЕНЬ · ПРЯМО НА МЕТР**

Проехать 1 м, выравнивая колёса по разнице импульсов: если левое насчитало больше — замедлить левое и ускорить правое, как П-регулятор из урока 3.2. Робот уходит от прямой не больше чем на 3 см.

ПРОВЕРЬ СЕБЯ

- [ ] Я могу объяснить, как энкодер считает обороты.
- [ ] Я могу перевести миллиметры в импульсы.
- [ ] Я могу объяснить, почему робот проезжает чуть дальше цели и как это уменьшить.

Проверь себя

★ Средний уровень

1. **20 импульсов на оборот, колесо 45 мм. Сколько импульсов нужно на 500 мм?**

- ○ Около 71
- ○ Около 222
- ○ 11

2. **Зачем счётчик импульсов объявлен как volatile long?**

- ○ Его меняет прерывание, и программа должна каждый раз читать его заново
- ○ Чтобы он сохранялся после выключения питания
- ○ Чтобы счёт шёл быстрее

3. **За одно время левое колесо насчитало 300 импульсов, правое — 270. Куда уводит робота?**

- ○ Вправо: левое колесо проехало больше
- ○ Влево: левое колесо проехало больше
- ○ Робот едет прямо

4. **Как точнее измерить число импульсов на оборот?**

- ○ Прокрутить колесо 10 оборотов по метке и разделить счёт на 10
- ○ Прокрутить один оборот на глаз
- ○ Посчитать прорези и не проверять

5. Почему поправка «остановиться на 5 импульсов раньше» перестаёт работать на другой скорости?

- ○ Чем быстрее робот едет, тем дальше он катится после остановки
- ○ На другой скорости меняется диаметр колеса
- ○ На другой скорости энкодер не работает

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

Энкодер превращает оборот колеса в импульсы, а импульсы — в пройденный путь. Робот едет не «сколько успеет», а сколько нужно, при любом заряде батарей.

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

энкодер	Датчик оборотов: диск с прорезями на оси колеса и датчик, который даёт импульс на каждую прорезь.
импульс	Короткий сигнал энкодера: прорезь прошла мимо датчика.
прерывание	Arduino откладывает основную программу и сразу выполняет короткую функцию, когда на контакт пришёл импульс.
volatile	Пометка переменной, которую меняет прерывание: программа каждый раз читает её заново.
длина окружности	Путь колеса за один оборот: π × диаметр.