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. Вспомни программу из урока 2.1, которая едет вперёд по времени. Запусти её дважды: на полной скорости и на скорости вдвое меньше. Время одинаковое.
2. Отметь, где робот остановился оба раза. Измерь линейкой оба пути и запиши в журнал.

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

Ты видишь: при одном и том же времени робот проезжает разное расстояние.

2 Как устроен энкодер

12 мин

Энкодер — это диск с прорезями на оси колеса и датчик-«вилка». Внутри вилки с одной стороны светит светодиод, с другой — приёмник. Когда мимо проходит прорезь, свет попадает на приёмник, и датчик посылает **импульс**. Один оборот колеса — столько импульсов, сколько прорезей на диске.

1. Посчитай прорези на диске энкодера. Запиши в журнал.
2. Найди в таблице проводки своего робота, к каким контактам подключены энкодеры. Запиши номера.

⚠ **БЕЗОПАСНОСТЬ**

Не подключай энкодеры при включённом питании. Если в таблице проводки контакты энкодеров не указаны — подключение делает преподаватель.

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

В журнале — число прорезей и номера контактов левого и правого энкодеров.

ⓘ **Как это устроено: Что такое прерывание**

Импульсы приходят быстро — десятки в секунду. Если программа занята чем-то ещё, она может их пропустить. Поэтому счёт делает **прерывание**: на каждый импульс Arduino сама откладывает основную программу, прибавляет 1 к счётчику и возвращается обратно. На Arduino Uno так умеют только контакты 2 и 3.

3 Импульсов на один оборот

22 мин

1. Создай скетч `LF1_Count`. Впиши номера контактов энкодеров.

Счёт импульсов LF1_Count.ino

```
// 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;

// Arduino вызывает её сама на каждый импульс левого энкодера.
void countLeft() {
  leftTicks++;
}

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

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();
}

void loop() {
  // Читаем оба счётчика, пока прерывания выключены: число не изменится посер
  noInterrupts();
  long left = leftTicks;
```

```
  long right = rightTicks;
  interrupts();

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

2. Загрузи и открой монитор порта. Поверни колесо рукой: число должно расти.
3. Сделай мелом или маркером метку на колесе. Прокрути колесо рукой **ровно 10 оборотов** по метке. Раздели число импульсов на 10 — это импульсов на оборот.
4. Повтори для второго колеса. Запиши оба числа в журнал.

✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**

В журнале — импульсов на оборот для каждого колеса. Обычно это число прорезей на диске.

ЕСЛИ НЕ ПОЛУЧИЛОСЬ

Что видишь	Что сделать
«Впиши контакты энкодеров»	Впиши номера контактов вместо −1
Число не растёт	Проверь провода и что диск проходит через щель датчика
Число растёт, когда колесо стоит	Колесо остановилось на краю прорези: сдвинь его чуть-чуть

2

ЗАНЯТИЕ 2 · 45 МИН

Точное расстояние

4 Миллиметры в импульсы

15 мин

За один оборот колесо проезжает длину окружности: `π × диаметр`. Диаметр колеса — 45 мм. Чтобы проехать 500 мм, нужно `500 / (π × 45)` оборотов, а импульсов — в `TICKS_PER_REV` раз больше.

1. Посчитай на бумаге: сколько импульсов на 500 мм, если на оборот 20 импульсов?
2. Создай скетч `LF1_Distance`, впиши контакты и своё число импульсов на оборот.

Точное расстояние LF1_Distance.ino

```
// 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;

void countLeft() {
  leftTicks++;
}

void countRight() {
  rightTicks++;
}

// Сколько импульсов нужно, чтобы проехать mm миллиметров.
long ticksFor(int mm, int ticksPerRev, int diameterMm) {
  float revolutions = mm / (PI * diameterMm);
  return (long)(revolutions * ticksPerRev + 0.5);
}

// Сколько импульсов в среднем насчитали оба колеса.
long averageTicks() {
  noInterrupts();
  long sum = leftTicks + rightTicks;
  interrupts();
  return sum / 2;
}

```

```
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
```

```
85
86    Serial.print("Цель, импульсов: ");
87    Serial.println(target);
88    Serial.print("Насчитано после остановки: ");
89    Serial.println(averageTicks());
90  }
91
92  void loop() {
93  }
```

3. Поставь робота на пол у метки старта, включи питание моторов. Робот проедет и остановится. Измерь путь линейкой от метки до той же точки корпуса.

> ✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**
> Ты посчитал: около 71 импульса. Робот проехал примерно 50 см, путь измерен.

5 Почему робот проезжает дальше

15 мин

После команды «стоп» робот ещё немного катится — как велосипед, если перестать крутить педали.

1. Сделай три заезда на 500 мм. Запиши пути в журнал.
2. Уменьши `speed` до 70 и сделай ещё три заезда. Стало точнее?

`speed`	Заезд 1	Заезд 2	Заезд 3
100			
70			

> ✓ **ДОЛЖНО ПОЛУЧИТЬСЯ**
> Таблица заполнена. Ты видишь, на какой скорости робот точнее.

6 Задания по уровням

12 мин

ОСНОВНОЕ · РОВНО 50 СМ

Проехать 500 мм с ошибкой не больше 2 см в трёх заездах из трёх.

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

Основное

1. Что считает энкодер?

- ○ Импульсы: сколько прорезей диска прошло мимо датчика
- ○ Расстояние до препятствия
- ○ Заряд батареи

2. **На диске 20 прорезей, на каждую — один импульс. Сколько импульсов за 3 оборота колеса?**

- ○ 60
- ○ 23
- ○ 20

3. **Какой путь проходит колесо диаметром 45 мм за один оборот?**

- ○ Около 141 мм: π × 45
- ○ 45 мм
- ○ 90 мм

4. **Почему езда «по времени» неточна?**

- ○ Скорость зависит от заряда батарей и пола, а время — нет
- ○ Arduino неправильно считает время
- ○ Моторы всегда крутятся одинаково

5. **Почему после остановки робот проезжает чуть больше цели?**

- ○ Он ещё немного катится по инерции
- ○ Энкодер считает лишние прорези
- ○ Колесо становится больше

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.