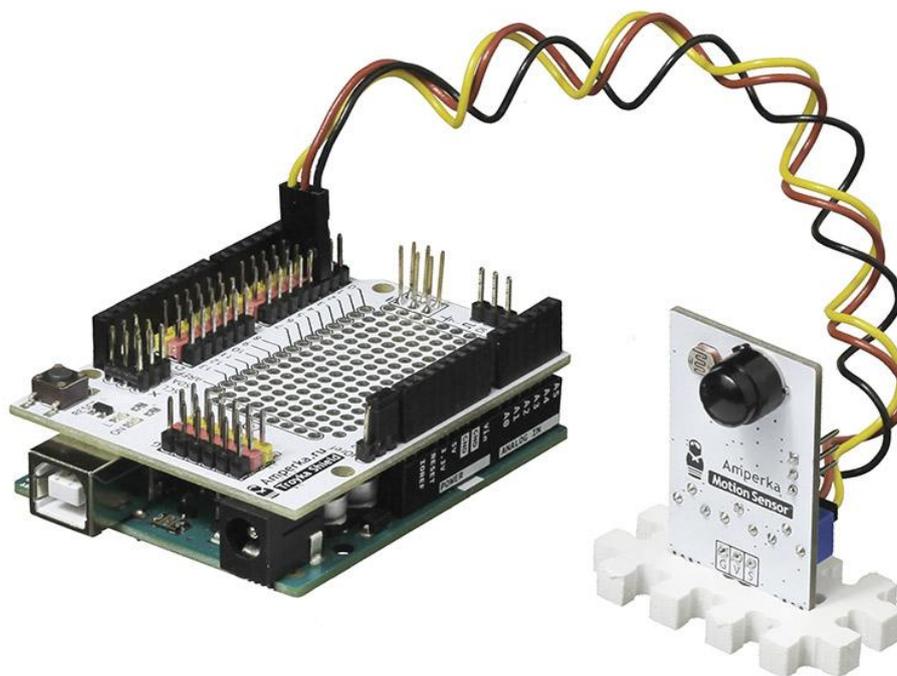


Инфракрасный датчик движения (Zelo-модуль)

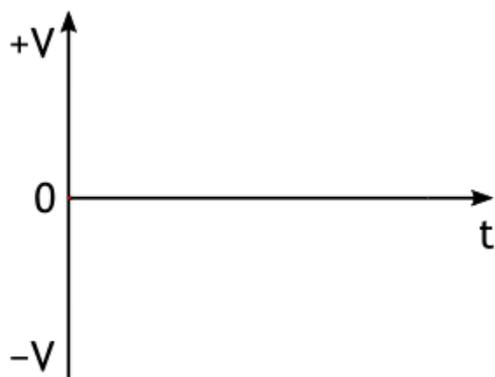
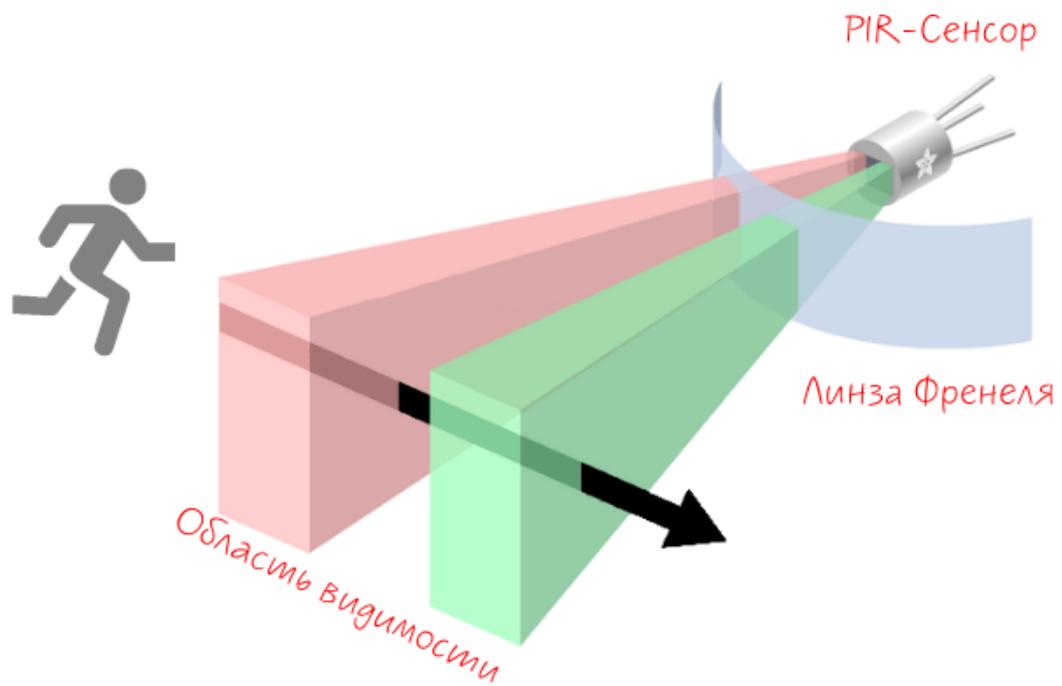
[Инфракрасный датчик движения](#) зафиксирует любое перемещения тёплых объектов: людей, животных и даже лунных рыб.



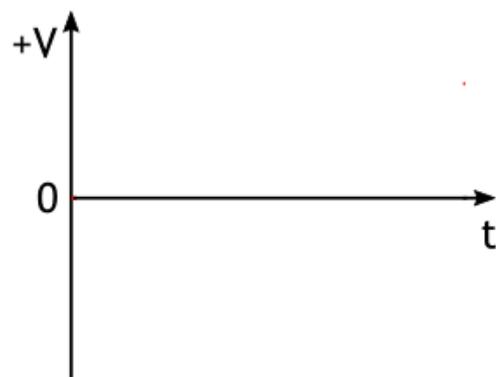
Принцип работы

Каждый теплокровный объект является источником теплового излучения. Длина волны теплового излучения зависит от температуры и находится в инфракрасной части спектра. ИК излучение невидимо для глаза, но улавливается пироэлектрическими датчиками.

- В радиусе видимости датчика полная тишина. Каждый чувствительный элемент PIR-сенсора получает постоянную дозу излучения. Следовательно выдаваемое напряжение равноценно.
- В области видимости появляется человек. Персонаж первым делом попадает в зону обзора первого элемента, на котором появляется положительный электрический импульс.
- Человек движется и пересекает второй элемент, который генерирует отрицательный импульс.
- Разнонаправленные импульсы регистрируются электронной схемой модуля, которая фиксирует перемещение объекта. В результате на выходе модуля генерируется положительный импульс.



Выходной сигнал
PIR-сенсора



Выходной сигнал
модуля

Примеры работы

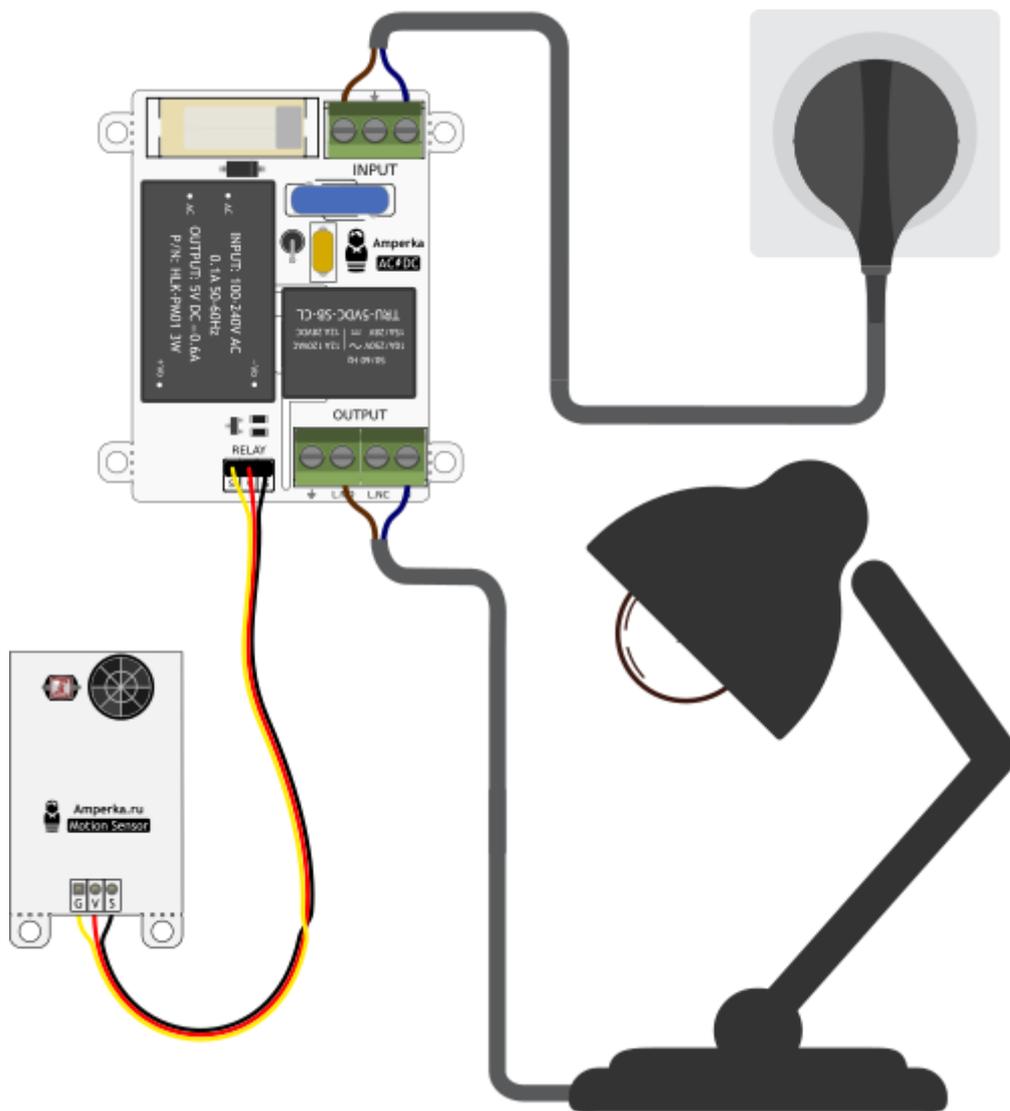
Простой датчик движения

Инфракрасный датчик может работать даже без микроконтроллера. Соберите простой детектор движения объекта.

Что вам понадобится?

- [Датчик движения](#)
- [Реле \(Тройка-модуль\)](#)
- [AC/DC \(Zero-модуль\)](#)
- Лампа освещения на 220 вольт

Схема устройства



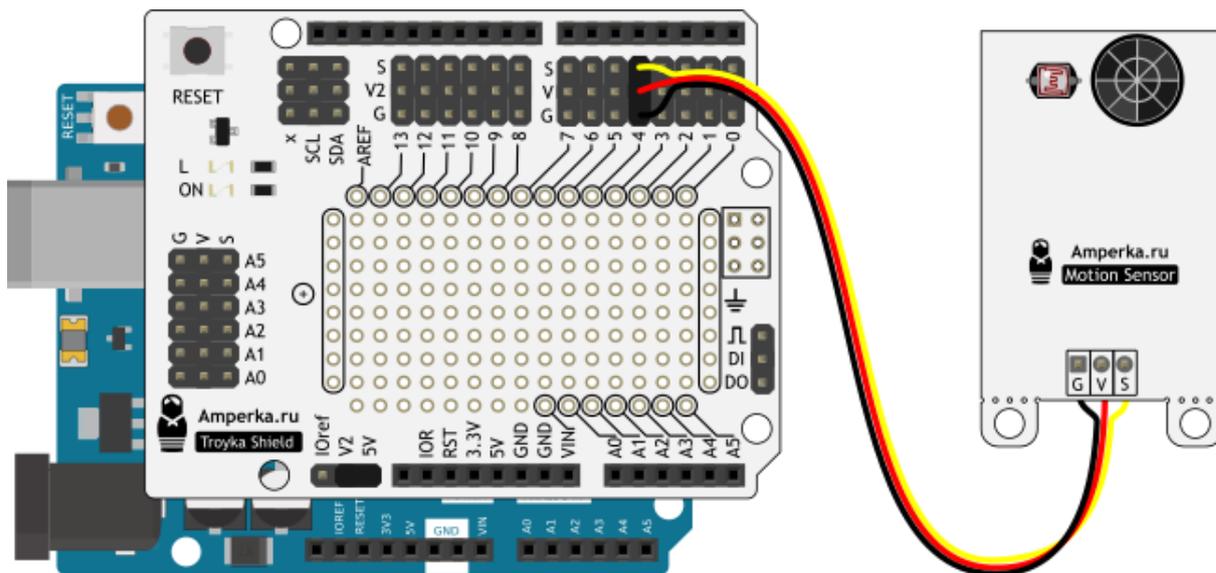
При появлении объекта в зоне видимости датчика, лампочка загорится.

Используйте инфракрасный датчик движения как одно из зёрен в своём умном доме. Тут уже не обойтись без [Arduino](#), [Raspberry Pi](#) или [Iskra JS](#).

Пример для Arduino

Подключим датчик движения к [Arduino Uno](#) через [Troyka Shield](#) к 4 цифровому пину.

Схема устройства



Код программы

Выведем в Serial-порт текущее состояние датчика с обновлением каждые 100 миллисекунд.

[motionState.ino](#)

```
// пин инфракрасного датчика движения
#define MOTION_PIN 4

void setup()
{
  // открываем монитор Serial-порта
  Serial.begin(9600);
  // настраиваем пин в режим входа
  pinMode(MOTION_PIN, INPUT);
}

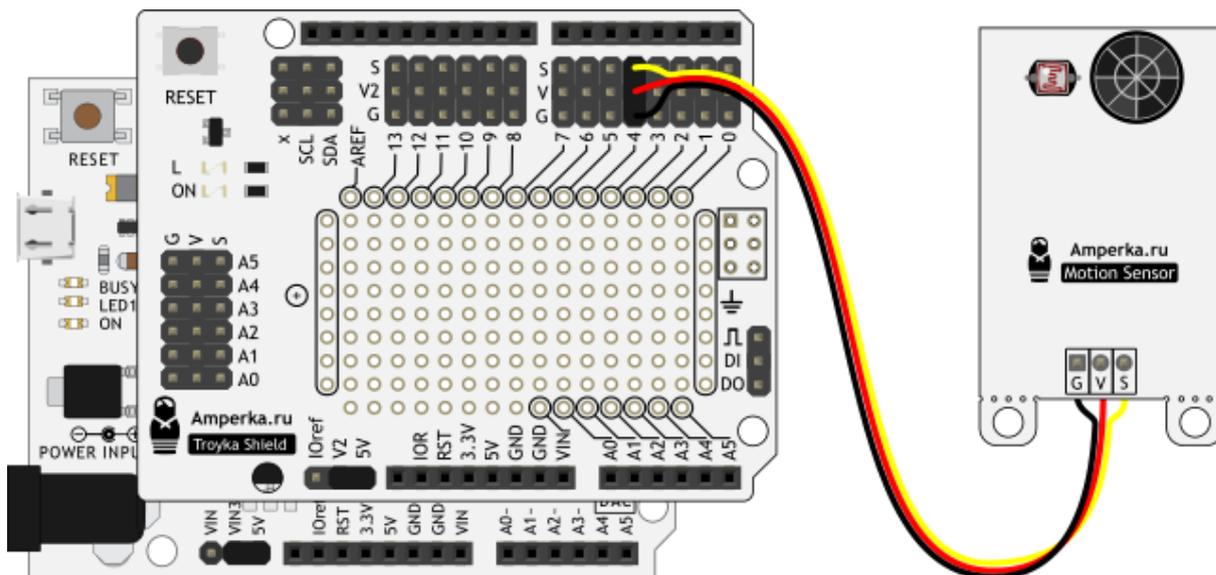
void loop()
{
  // считываем состояние пина
  int motionState = digitalRead(MOTION_PIN);
  // выводим в Serial-порт
  Serial.println(motionState);
  delay(100);
}
```

После прошивки платы, вы увидите бегущие нули. А как только появится живой объект на горизонте — нули сменятся на единицы.

Пример для Iskra JS

Скоммутируем PIR-сенсор к [Iskra JS](#) через [Troyka Shield](#) к 4 цифровому пину.

Схема устройства



Код программы

Зафиксируем движение объекта с помощью Espruino и языка JavaScript.

[motionDetect.js](#)

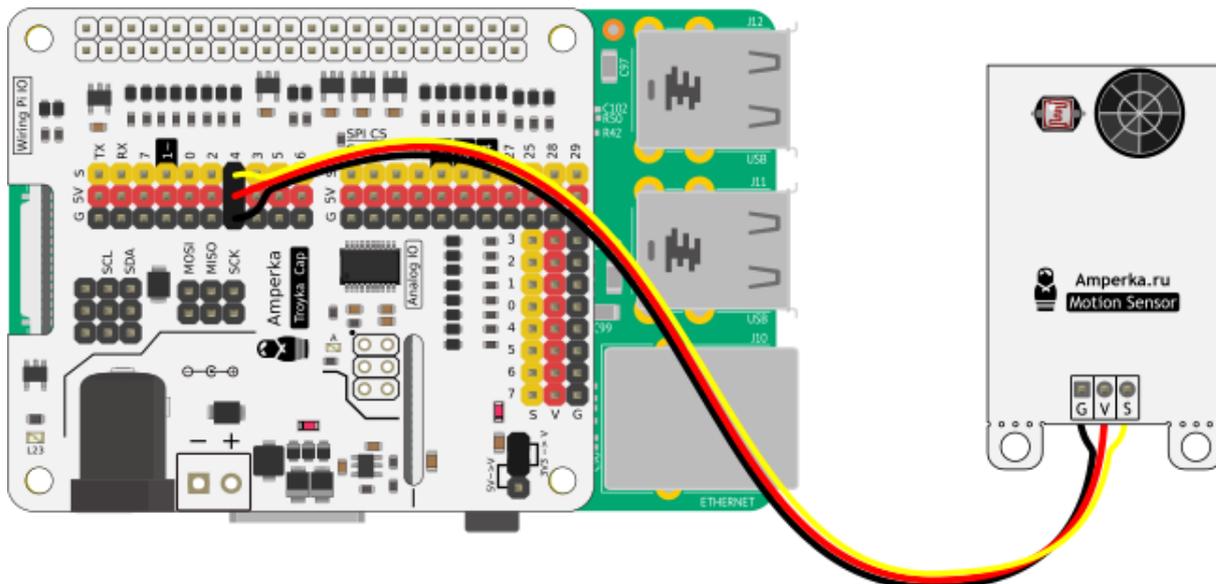
```
// наблюдаем за датчиком движения
setWatch(function() {
  // если датчик зафиксировал движение
  // печатаем об этом в консоль
  print("Movement detected");
}, P4, {
  // функция вызывается многократно
  repeat: true,
  // фиксация восходящего фронта
  edge: "rising"
});
```

В результате вы увидите сообщение в консоле, при обнаружении живого объекта в зоне видимости сенсора.

Пример для Raspberry Pi

Поймаем живой объект одноплатником [Raspberry Pi](#). Подключите сенсор движения к 4 пину Raspberyy. Для избежания макеток и проводов используйте плату расширения [Troyka Cap](#).

Схема устройства



Код программы

[motionState.py](#)

```
# библиотека для работы с методами языка Wiring (Arduino)
import wiringpi as wp
# инициализация WiringPi
wp.wiringPiSetup()
# пин 4 в режим входа
wp.pinMode(4, 0)

while (True):
    # считываем состояние с датчика движения
    motionState = wp.digitalRead(4)
    # печатаем результат в консоль
    print(motionState);
    # ждём 100 мс
    wp.delay(100)
```

После запуска скрипта вы увидите текущие показатели сенсора. Пока движения нет — в консоле выводятся нули, при обнаружении живого объекта — единицы.

Элементы платы



Пирозлектрический сенсор с линзой Френеля

Микросхема управления

Мозгом сенсора является микросхема [BISS0001](#). Чип считывает и обрабатывает сигналы с PIR-сенсора. В итоге на выходе модуля бинарный цифровой. Есть движение — единица, нет — ноль.

Выбор режима работы

Режим работы модуля задается переключкой. Есть два режима — режим H и режим L. На фото выше в модуле установлен режим H.

Режим H — в этом режиме при срабатывании датчика несколько раз подряд на его выходе (на OUT) остается высокий логический уровень.

Режим L — в этом режиме на выходе при каждом срабатывании датчика появляется отдельный импульс.

Регулировка режимов работы

На модуле расположено три потенциометра отвечающие за подстройку режима работы:

- T.on — регулировка длительности сигнала при обнаружении движения объекта. Время на которое сенсор будет выдавать гарантированно высокий уровень при детектировании объекта. Диапазон длительности: от одной секунды до пяти минут.
- T.off — регулировка длительности игнорирования движения при повтором срабатывании датчика. Время на которое сенсор не будет реагировать на

движущий объект при циклическом срабатывании датчика. Временной диапазон: от нуля до пяти секунд.

- SENS — регулировка чувствительности сенсора.

Световой индикатор

Индикаторный светодиод дублирующий выходной сигнал с датчика движения. При высоком уровне сигнала с модуля — светодиод горит, при низком — не горит.

Датчик освещённости

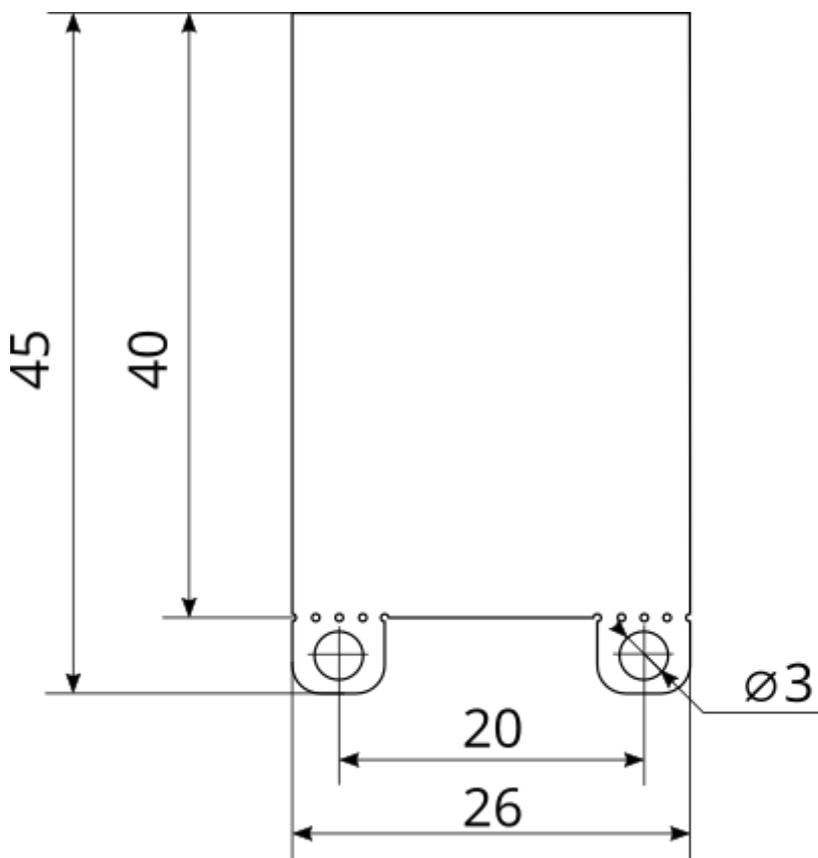
Датчик освещённости на фоторезисторе [GL5528](#), подкорректирует чувствительность модуля на солнечный свет. Это удобно при необходимости отключение работы сенсора в дневное время суток.

Тройка-контакты

На модуле выведена группа Тройка-контактов:

- Сигнальный (S) — цифровой выход сенсора. Используется для передачи текущего состояния модуля. Подключите к цифровому пину микроконтроллера.
- Питание (V) — соедините с рабочим напряжением микроконтроллера.
- Земля (G) — соедините с землёй микроконтроллера.

Габаритный чертёж



Характеристики

- Напряжение питания: 3.3–5 В
- Расстояние наблюдения: 7 м
- Угол обзора: 110°
- Длительность сигнала при обнаружении движения (Ton): от 1 секунды до 5 минут
- Длительность игнорирования движения при повтором срабатывании (Toff): до 5 секунд