

Автополив — дополнение к набору «Йодо»

Электронная версия брошюры из набора Автополив. Здесь собраны все исходные коды экспериментов, подсказки и хаки по прохождению набора.



Необходимые модули из набора «Йодо»

Для проектов Автополива тебе потребуются платы и модули из базового набора «Йодо»:

- Iskra JS
- Потенциометр (Тройка-модуль)
- Кнопка (Тройка-модуль)
- Зуммер (Тройка-модуль)
- Датчик освещённости (Тройка-модуль)
- Светодиод (Тройка-модуль)
- Микросервопривод FS90

- Ультразвуковой дальномер HC-SR04

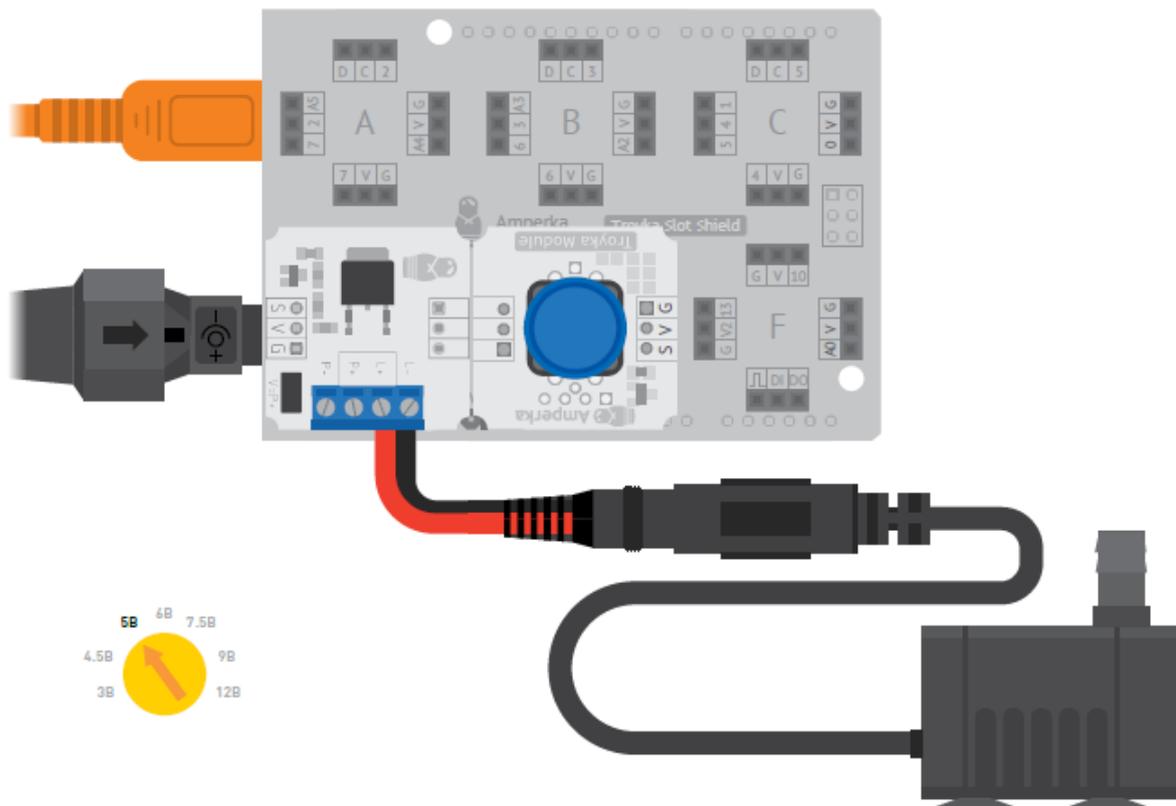
Если ты уже используешь эти модули и платы в своих крутых проектах, а разбирать их не хочется — закажи недостающие!

Эксперименты

2. Живой фонтан

Фонтан силе своей подчини. Через силовой ключ подключи и кнопку управления добавь.

Схема сборки устройства



Код эксперимента

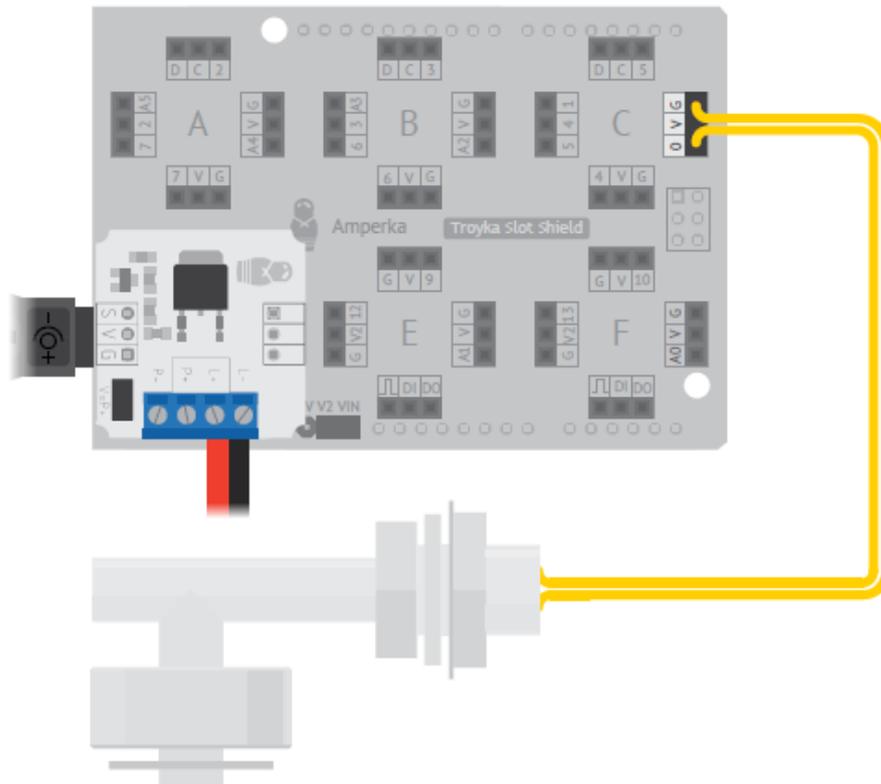
```
var button = require('@amperka/button').connect(A1);  
var pump = require('@amperka/power-control').connect(P11);  
button.on('press', function() {  
  pump.toggle();  
});
```

На нашей вики ты сможешь узнать подробнее о функциях библиотеки [@amperka/power-control](#).

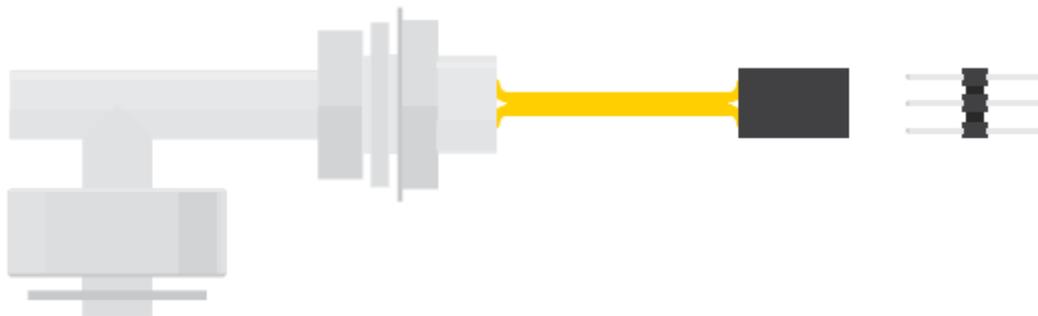
3. Дренаж

Перегреется помпа и из строя выйдет, если без воды стоять будет долго. Датчик уровня установи, чтобы силы её сберечь.

Схема сборки устройства



Используй переходники для соединения проводов.



Код эксперимента

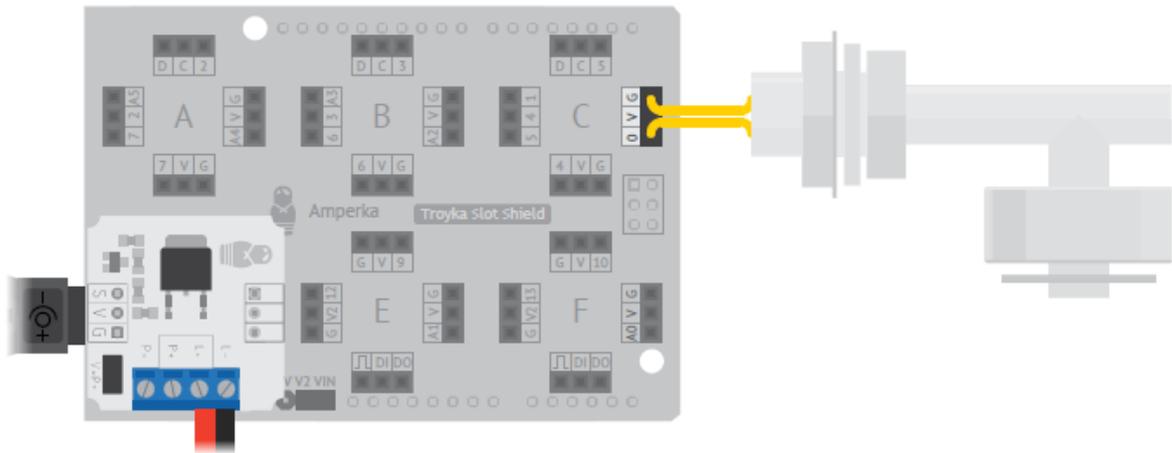
```
var pump = require('@amperka/power-control').connect(P11);  
pump.turnOn();  
  
var level = require('@amperka/water-level').connect(P0);  
  
level.on('down', function () {  
  print('Water level is low');  
  pump.turnOff();  
});
```

Библиотека [@amperka/water-level](#).

4. Автопополнение

Тонкой настройкой датчика уровня займёмся. Автопополнение воды практикой нам будет.

Схема сборки устройства

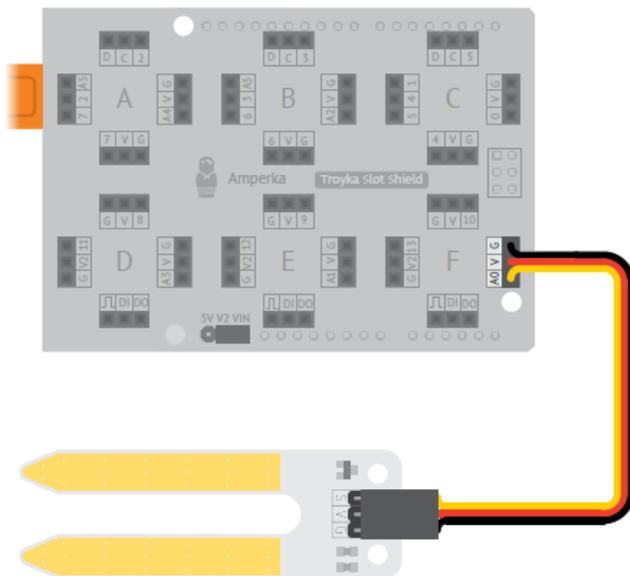


```
var pump = require('@amperka/power-control').connect(P11);  
var level = require('@amperka/water-level')  
  .connect(P0, {debounce: 0.5});  
level.on('up', function () {  
  pump.turnOff();  
});  
level.on('down', function () {  
  pump.turnOn();  
});
```

5. Калибровка

Азы датчика влажности почвы познаем и рабочий диапазон на Iskra JS увидим.

Схема сборки устройства

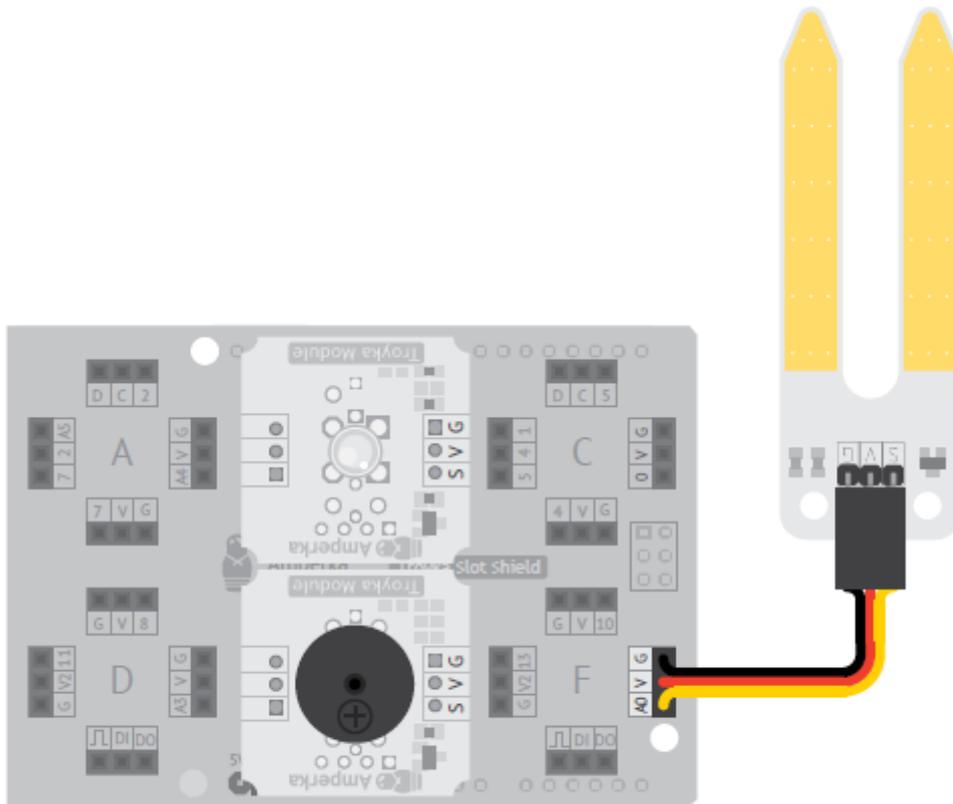


```
setInterval(function () {  
  var value = analogRead(A0);  
  print(value);  
}, 200);
```

6. Напоминальщик

В помощь джедаю звуковое уведомление сделаем. Поливать растения напоминать будет, когда почва суха.

Схема сборки устройства



```
var hyst = require('@amperka/hysteresis')
    .create({high: 0.5, highLag: 2, low: 0.4, lowLag: 2});
var led = require('@amperka/led').connect(A2);
var buzzer = require('@amperka/buzzer').connect(A1);

setInterval(function() {
  hyst.push(analogRead(A0));
}, 200);

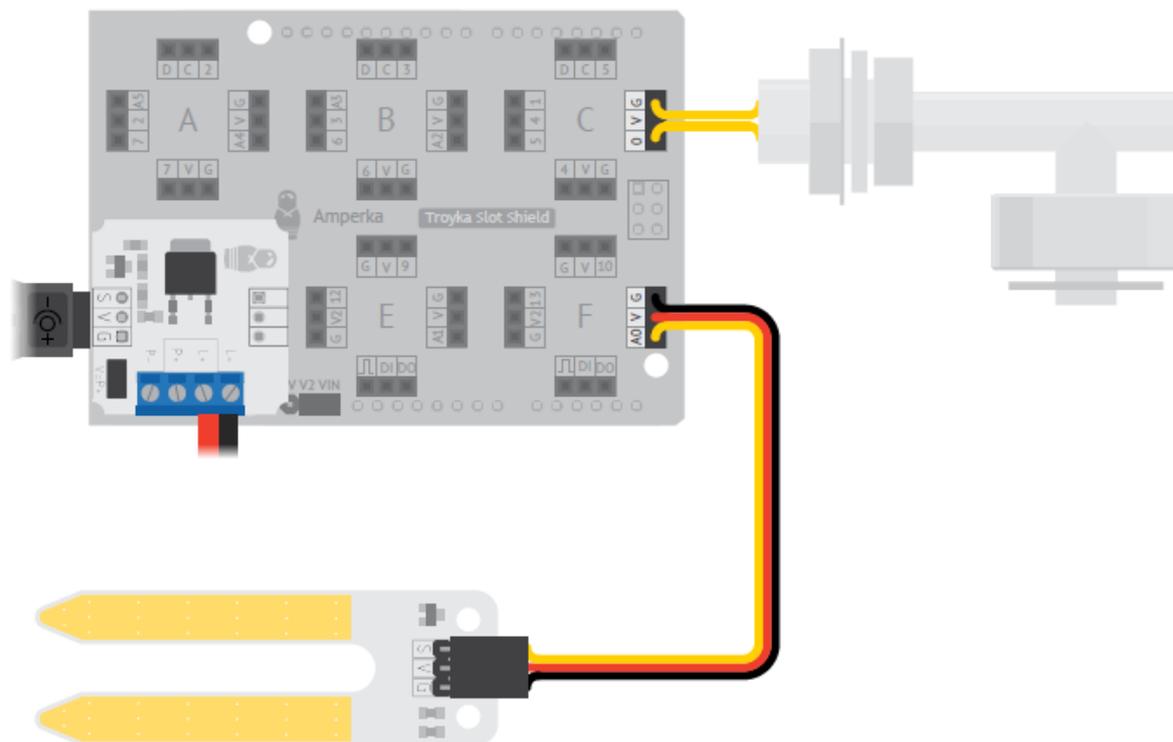
hyst.on('low', function() {
  buzzer.beep(1, 0.5);
  led.blink(1, 0.5);
});

hyst.on('high', function() {
  buzzer.turnOff();
  led.turnOff();
});
```

7. Автополив

Автополив создай. Если почва суха, пусть помпа включится и растение польёт. Датчик уровня помпе выйти из строя не даст.

Схема сборки устройства



```
var hyst = require('@amperka/hysteresis')
    .create({high: 0.5, highLag: 2, low: 0.4, lowLag: 2});
var pump = require('@amperka/power-control').connect(P11);
var level = require('@amperka/water-level').connect(P0);

setInterval(function() {
  hyst.push(analogRead(A0));
}, 200);

hyst.on('low', function() {
  pump.turnOn();
});

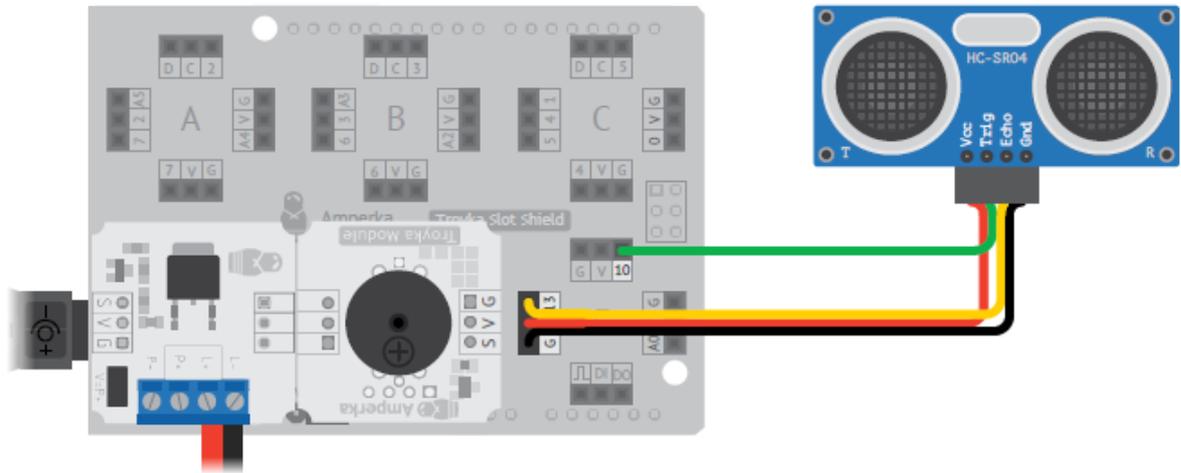
hyst.on('high', function() {
  pump.turnOff();
});

level.on('down', function() {
  pump.turnOff();
});
```

8. Пушистая оборона

Домашних животных своих отучи растительность уничтожать. Демотиватор для них сделаем.

Схема сборки устройства



```
var sonic = require('@amperka/ultrasonic')
    .connect({trigPin: P10, echoPin: P13});

var pump = require('@amperka/power-control').connect(P11);
var buzzer = require('@amperka/buzzer')
    .connect(A1)
    .frequency(440);

setInterval(function() {
    sonic.ping(function(err, value) {
        if (err) return;

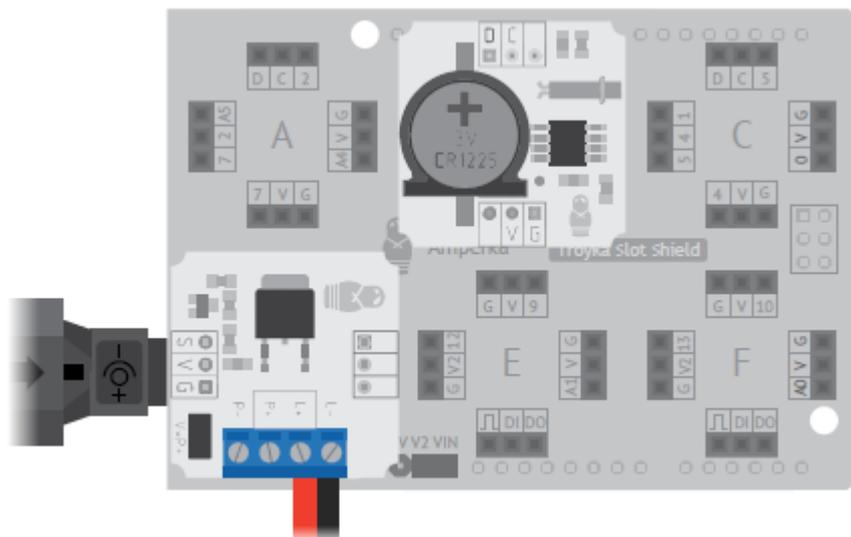
        if (value < 20) {
            buzzer.beep(0.5, 0.2);
        } else if (value < 50) {
            buzzer.beep(1, 0.5);
        } else {
            buzzer.turnOff();
        }

        if (value < 5) {
            pump.turnOn();
        } else {
            pump.turnOff();
        }
    }, 'cm');
}, 200);
```

9. Таймер

Галактическое время почувствуй, чтобы растения по расписанию поливать.

Схема сборки устройства



```
var rtc = require('@amperka/rtc').connect();
rtc.setTime();
var pump = require('@amperka/power-control').connect(P11);
var lastWatering = -1;
setInterval(function () {
  var date = rtc.getTime();
  print(date.toString());
  if (lastWatering === date.getDate()) return;
  if (date.getHours() === 13) {
    lastWatering = date.getDate();
    pump.pulse(3);
  }
}, 5000);
```

Внимание! Если прошивка твоей Iskra JS старше 1.93, значит она уже умеет работать с часовыми поясами! Просто добавь самой первой строкой в коде:

```
E.setTimeZone(+3);
```

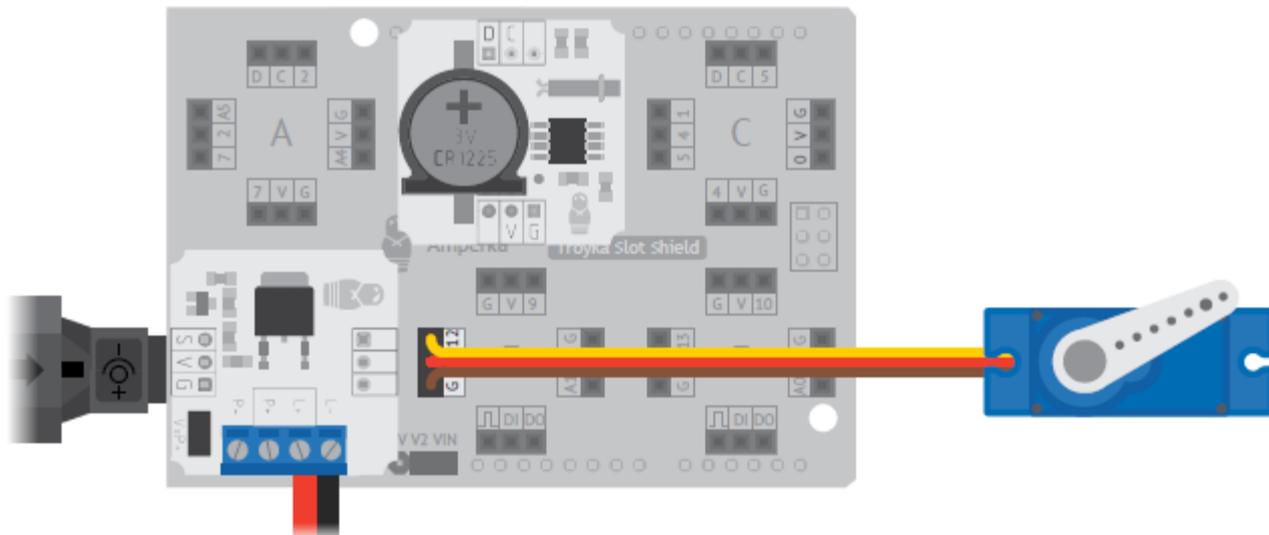
Важно! Обновить прошивку поможет раздел вики [Обновление прошивки](#).

На нашей вики ты сможешь узнать подробнее о функциях библиотеки [@amperka/rtc](#).

10. Дождеватор

О растениях на даче забывать не стоит. Дождеватор для газона сделаем. Сервопривод большую площадь охватить поможет.

Схема сборки устройства



```
var rtc = require('@amperka/rtc').connect();
rtc.setTime();

var pump = require('@amperka/power-control')
    .connect(P11);

var servo = require('@amperka/servo').connect(P12);
var swing = require('@amperka/animation').create({
  from: 30,
  to: 150,
  duration: 5,
  updateInterval: 0.02
}).queue({
  from: 150,
  to: 30,
  duration: 5
});

swing.on('update', function(val) {
  servo.write(val);
});

var lastWatering = -1;

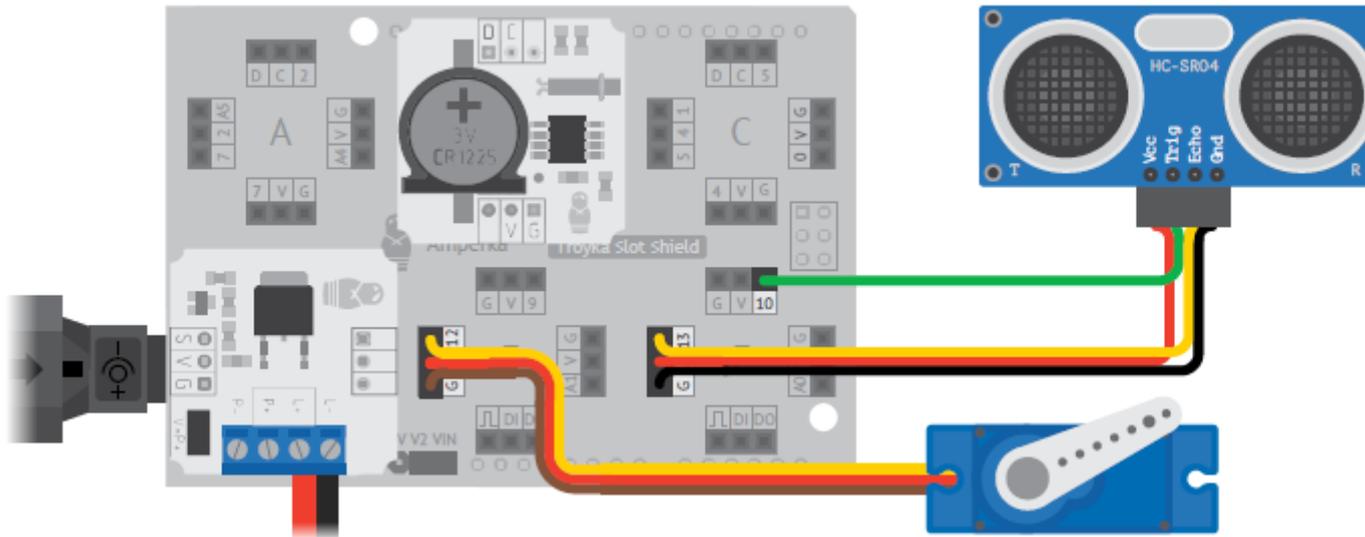
setInterval(function () {
  var date = rtc.getTime();
  if (lastWatering === date.getDate()) return;

  if (date.getHours() === 6) {
    lastWatering = date.getDate();
    swing.play();
    pump.pulse(10);
  }
}, 1000);
```

11. Дождеватор 3000

Дождеватор усовершенствуем, ультразвуковой дальномер добавив. Чтобы других джедаев водой не облить.

Схема сборки устройства



```
var rtc = require('@amperka/rtc').connect();
var pump = require('@amperka/power-control').connect(P11);
var servo = require('@amperka/servo').connect(P12);
var sonic = require('@amperka/ultrasonic')
    .connect({trigPin: P10, echoPin: P13});

var angle = 90;
var delta = 1;
var lastWatering = -1;
var portions = 50;
var wateringIntervals = portions;

setInterval(function () {
    var date = rtc.getTime();
    if (lastWatering === date.getDate()) return;

    if (date.getHours() === 6) {
        wateringIntervals = 0;
        lastWatering = date.getDate();
    }
}, 1000);

setInterval(function() {
    if (wateringIntervals > portions) {
        pump.turnOff();
        return;
    }

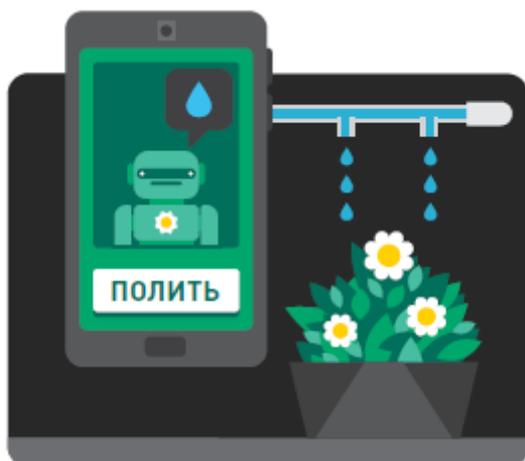
    sonic.ping(function(err, value) {
        if (value < 100 || wateringIntervals > portions) {
            pump.turnOff();
        } else {
            wateringIntervals = wateringIntervals + 1;
            servo.write(angle);
            pump.turnOn();
            if (angle === 150 || angle === 30) {
                delta = -delta;
            }
            angle = angle + delta;
        }
    }, 'cm');
}, 100);
```

Идеи проектов

На достигнутом не останавливайся, свои проекты создавай и мудрость джедая постигай.

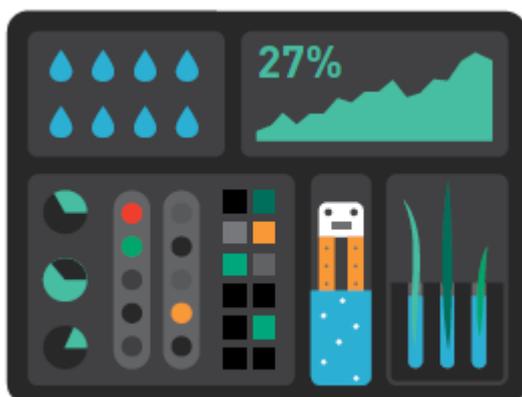
Telegram-бот и IFTTT

Автоматизируй полив и общайся со своим садом с помощью Telegram-бота. Установи программные триггеры с сервисом IFTTT. Всему этому ты научишься с эпизодом 2: «Интернет вещей» — дополнение набора «Йодо»



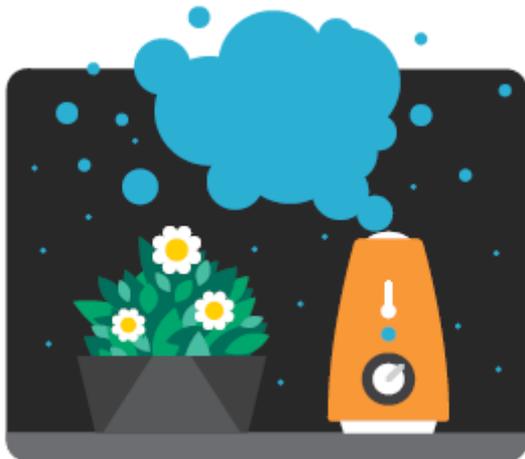
Автополив 2.0

Добавь датчик влажности воздуха для качественного ухода за растениями. Отслеживай параметры на сайте dweet.io. Для этого потребуется модуль Wi-Fi или Ethernet Shield.



Увлажнитель воздуха

Приспособь бытовой увлажнитель воздуха. Управляй им через модули Nano Switch. Используй часы реального времени для работы по расписанию.



#Структор

Устройства на Тройка Slot Shield можно удобно оформлять в корпус из #Структора с помощью набора плашек Slot Box (#Структор).

