# Отчёт по теме “Система автоматизированного управления и регулирования системы подачи химических элементов”

# Компонентная база

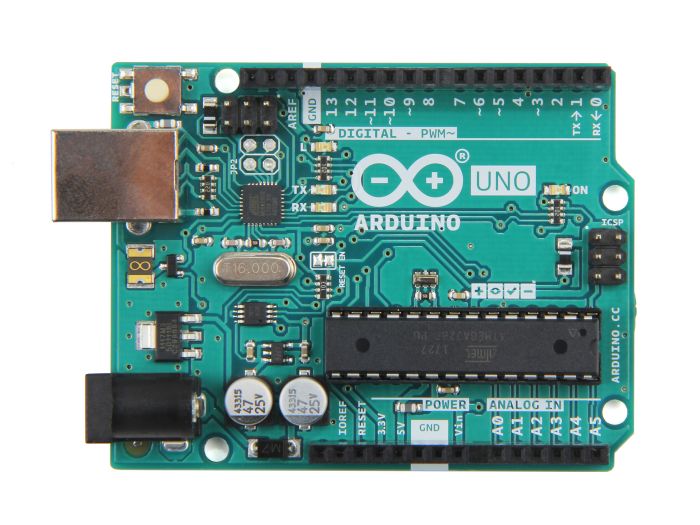
## Отладочная плата Arduino Uno Rev3

Arduino – аппаратная вычислительная платформа, состоящая из двух основных компонентов: платы ввода–вывода и среды разработки на языке Processing/Wiring.

Плата Arduino состоит из микроконтроллера и элементов обвязки для программирования и интеграции с другими схемами. На многих платах также имеется линейный стабилизатор напряжения. Выводы платформы Arduino могут работать как входы или как выходы.

Arduino Uno Rev3 — это плата, основанная на микроконтроллере ATmega328P. Платформа имеет 14 цифровых пинов входа/выхода, 6 из которых могут использоваться как выходы ШИМ, 6 аналоговых входов, кварцевый генератор 16 МГц, разъем USB, силовой разъем, разъем ICSP и кнопку перезагрузки. Для работы необходимо подключить платформу к компьютеру посредством кабеля USB, либо подать питание при помощи адаптера AC/DC или батареи.

В отличие от всех предыдущих плат Ардуино, Uno в качестве преобразователя интерфейсов USB-UART использует микроконтроллер ATmega16U2 (ATmega8U2 до версии R2) вместо микросхемы FTDI. На китайских вариантах используется преобразователь интерфейсов USB-UART CH340G.

Рисунок (Arduino Uno Rev3)

**Технические характеристики**

* Напряжение питания 5В.
* Входное питание 7-12В (рекомендованное).
* Количество цифровых пинов – 14.
* 6 аналоговых входов.
* ШИМ (PWM) пины – 6
* Постоянный ток через вход/выход – 40 мА.
* Максимальный выходной ток вывода 3.3V – 50 мА.
* Флэш- память 32 КБ (ATmega328P) из которых 0.5 КБ используются загрузчиком.
* SRAM – 2 КБ (ATmega328P).
* ОЗУ 1 Кб или 2 Кб, в зависимости от чипа.
* EEPROM 1 КБ (ATmega328P).
* Тактовая частота 16 МГц.
* Размеры 68.6 х 53.4 мм.
* Вес 25 г.

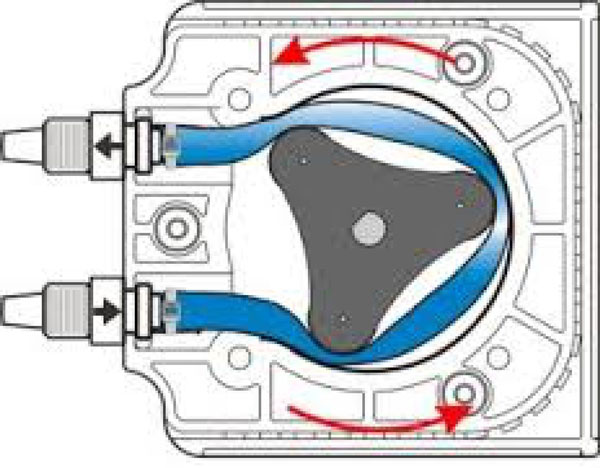
## Цифровой перистальтический насос

Цифровой перистальтический насос - это новый продукт, разработанный DFRobot. Это привод для жидкостной системы (Рисунок).

Рисунок

Перистальтический насос DFRobot включает в себя драйвер двигателя PPM на насосе, это драйвер двигателя с гравитационным интерфейсом (PH2.0-3P). Управляющий сигнал совместим со стандартным серводвигателем, вы можете использовать Arduino, Raspberry Pi, micro: bit и другие виды микроконтроллеров для управления его направлением и скоростью потока. Трубка в головке насоса изготовлена из материала BPT, который является антикислотным и имеет длительный срок службы.

Перистальтические насосы - это своего рода поршневой насос, используемый для подачи небольшого количества жидкости. Жидкость содержится в гибкой трубе, установленной внутри корпуса круглого насоса. Когда ротор вращается непрерывно, жидкость в трубе будет сдавливаться и доставляться на другую сторону. Таким образом, скорость потока жидкости может легко контролироваться с точностью (Рисунок).



Рисунок

В качестве разновидности дозирующего насоса для количественной доставки удобно использовать перистальтический насос. Он имеет широкий спектр сценариев применения, таких как научные эксперименты, гидропоника, капельное орошение, разведение рыб, контроль окружающей среды в креветочном цилиндре и т. д. С помощью этого перистальтического насоса можно автоматизировать большое количество экспериментов по титрованию, таких как потенциометрическое титрование, индикатор колориметрическое титрование, рН-основанное титрование и так далее. Это очень полезно для повышения эффективности работы

С другой стороны, перистальтические насосы также имеют широкий спектр применения в пищевой, медицинской и других отраслях промышленности.

**Технические особенности:**

* Тихая работа,
* Бюджетный,
* Простая структура,
* Легко поддерживать,
* Три ротора с умеренной пульсацией,
* Не токсичен и не загрязняет окружающую среду

**Технические характеристики:**

Драйвер:

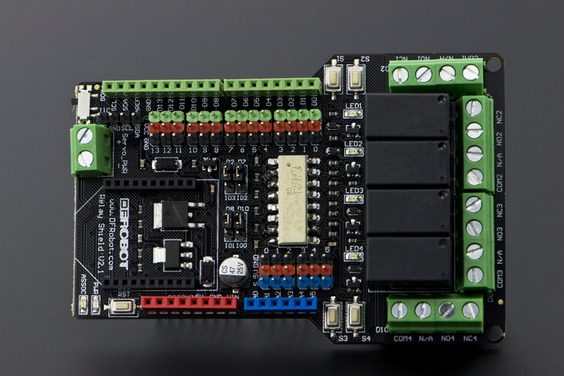
* Входное напряжение: 5V-6V
* Максимальный непрерывный рабочий ток: 1,8А
* Пиковый ток: 2.5А
* Ток покоя: <1 мА (без входного сигнала PPM)
* Разрешение сигнала PPM: 1us
* Ширина положительного импульса сигнала PPM: 500us-2500us
* Диапазон ширины импульса прямого хода: 500–1400 мс (500 мс: максимальная скорость движения вперед)
* Диапазон ширины импульса остановки: 1400us-1600us
* Диапазон ширины инвертированного импульса: 1600 - 2500 мкс (2500 мс: максимальная инвертированная скорость)
* Частота PPM: 50 Гц (сигнал сервоуправления)
* Разъем: Gravity PH2.0-3P
* Размер: 27,4 х 28,7 мм / 1,08 х 1,13 дюйма

Перистальтический насос:

* Мотор: двигатель постоянного тока
* Номинальное напряжение: 6 В
* Номинальная мощность: 5 Вт
* Материал трубки: BPT
* Технические характеристики трубки: внутренний диаметр 2,5 мм, наружный диаметр 4,5 мм
* Материал головки насоса: инженерный пластик
* Пульсация: три ролика, малая пульсация
* Расход: ≥45 мл / мин (почти 1 мл / с)
* Условия работы: температура окружающей среды 0 ~ 40 ℃, относительная влажность <80%

## 4 Channel Relay Shield for Arduino

Этот Relay Shield для Arduino действует как переключатель. Он обычно используется для управления чем-либо более 5 В, например, лампами, водяными насосами и гаражными воротами (Рисунок).

Рисунок

Возможно, один из самых интересных проектов – это возможность управлять устройствами большой мощности, такими как лампы, водяные насосы, гаражные ворота и так далее. Но сопряжение с высокой мощностью и переменным током является довольно опасной задачей при создании схемы. Реле позволяют подключать внешнюю электрическую цепь, например, переключатель. С помощью реле вы можете теперь подключать внешние цепи или устройства непосредственно к вашему Arduino.

Этот Arduino Shield – совершенно по-новому спроектированный, способный контролировать 4 реле одновременно. Максимальная коммутируемая мощность DC 90 Вт или AC 360 ВА обеспечивает прямое управление контроллерами Arduino Uno / DFRduino через цифровые входы / выходы с внешним источником питания 9 В. Со встроенным разъемом Xbee, он может быть беспроводным дистанционным управлением через Xbee / Bluetooth / WPM.

Этот Arduino Shield является идеальным решением для домашней автоматизации и робототехники.

**Технические особенности:**

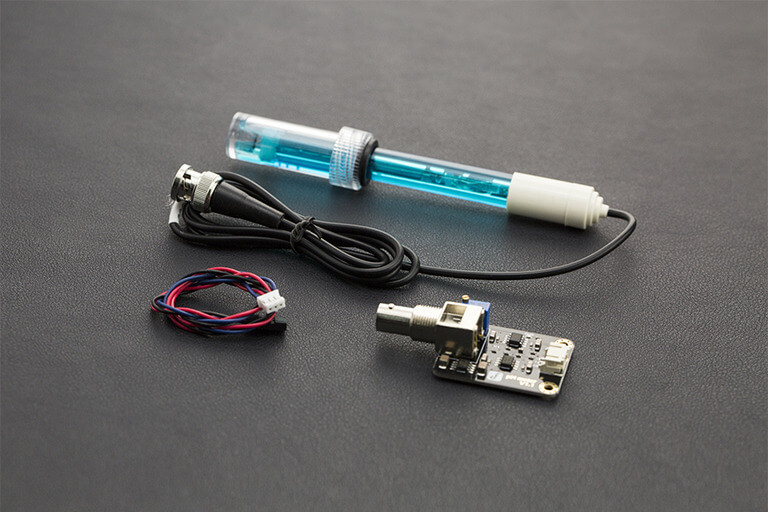
* Совместим с Arduino UNO Rev3
* У каждого реле есть своя кнопка для проверки соединения
* Стандартный DFRobot 3 Pin
* XBee I/O прямое управление

**Технические характеристики:**

* Совместим с Arduino Rev3
* 4 кнопки для тестирования модуля
* Светодиодный индикатор состояния реле
* Поддержка Xbee IO с прямым контролем
* Гнездо Xbee для беспроводной связи
* Выбираемый цифровой вывод ввода-вывода для управления (по умолчанию цифровой 2,7,8,10)
* 6 каналов аналогового ввода-вывода и 13 каналов цифрового ввода-вывода
* До 4 реле с фото-схемой
* Рейтинг контактов 3A AC 120V / DC 24V
* Максимальное коммутируемое напряжение переменного тока 240 В / 60 В постоянного тока (рекомендуется менее 38 В)
* Максимальный Ток Переключения 5А
* Максимальная коммутируемая мощность AC 360VA / DC 90 Вт
* Электрическая жизнь (мин) 100 000 операций
* Механическая жизнь (мин) 10 000 000 операций
* Стандарт безопасности (реле) UL CUL TUV CQC
* Рабочее напряжение катушки 9VDC
* Рабочая температура от -30 + до + 85 ℃
* Рабочая влажность 40% - 85%
* Размер: 95x65mm

## Аналоговый датчик pH / измерительный комплект для Arduino

Аналоговый рН-метр DFRobot, специально разработанный для контроллеров Arduino, имеет удобный и практичный разъем "Gravity" и множество функций. Мгновенное соединение с вашим Arduino для измерения pH при ± 0.1pH (25 ℃). Для большинства любителей этот большой диапазон точности и низкая стоимость делают его отличным инструментом для биоробототехники и других проектов! Он имеет светодиод, который работает как индикатор питания, разъем BNC и интерфейс датчика PH2.0. Чтобы использовать его, просто подключите датчик pH к разъему BND и подключите интерфейс PH2.0 к порту аналогового входа любого контроллера Arduino (Рисунок).

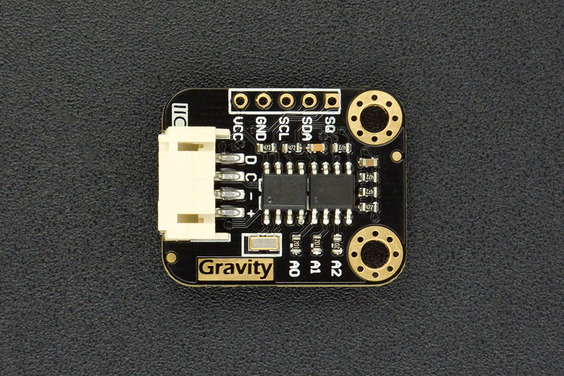
Рисунок

Это лабораторный датчик, он не может быть погружен в жидкость слишком долго.

**Технические характеристики:**

* Модуль питания: 5,00 В
* Размер модуля: 43 х 32 мм (1,69 х 1,26 ")
* Диапазон измерения: 0 - 14PH
* Температура измерения: 0 - 60 ℃
* Точность: ± 0,1pH (25 ℃)
* Время отклика: ≤ 1мин
* Датчик pH с разъемом BNC
* pH2.0 интерфейс (патч 3 фута)
* Потенциометр регулировки усиления
* Светодиодный индикатор питания

## DS1307 RTC Module

Рисунок (DS1307 RTC Module)

DS1307 это небольшой модуль, предназначенный для подсчета времени. Собранный на базе микросхемы DS1307ZN с реализацией питания от литиевой батарейки (LIR2032), что позволяет работать автономно в течение длительного времени. Также на модуле, установлена энергонезависимая память EEPROM объемом 32 Кбайт (AT24C32). Микросхема AT24C32 и DS1307ZN связаны обшей шиной интерфейсом I2C.

## Аналоговый датчик / измеритель электропроводности V2 (K = 1)

Рисунок (Аналоговый датчик / измеритель электропроводности V2 (K = 1))

Аналоговый измеритель электропроводности V2 специально используется для измерения электропроводности водного раствора, а затем для оценки качества воды, которая часто используется в водной культуре, аквакультуре, обнаружении воды в окружающей среде и других областях. Вы также можете ознакомиться с Руководством по выбору датчика жидкости, чтобы лучше ознакомиться с серией датчиков жидкости.

Этот продукт, как модернизированная версия измерителя электропроводности V1, значительно повышает удобство использования и точность данных. Он поддерживает вход напряжения 3 ~ 5 В и совместим с основной платой управления 5 В и 3,3 В; Выходной сигнал, отфильтрованный аппаратными средствами, имеет низкий уровень дрожания; Источник возбуждения принимает сигнал переменного тока, который эффективно уменьшает эффект поляризации, повышает точность и продлевает срок службы зонда; Библиотека программного обеспечения использует метод двухточечной калибровки и может автоматически определять стандартный буферный раствор, настолько простой и удобный.

С этим продуктом, главной платой управления (например, Arduino) и библиотекой программного обеспечения вы можете быстро построить измеритель электрической проводимости. DFRobot предлагает разнообразные продукты для датчиков качества воды, с одинаковым размером и интерфейсом, которые не только отвечают требованиям различных испытаний качества воды, но также подходят для изготовления многопараметрического тестера качества воды.

Проводимость является обратной величиной сопротивления, которое связано со способностью материала переносить ток. В жидкости обратная величина сопротивления, проводимости, является мерой ее способности проводить электричество. Проводимость является важным параметром качества воды. Это может отражать степень присутствия электролитов в воде.

СПЕЦИФИКАЦИЯ

Плата преобразования сигналов (передатчик) V2:

* Напряжение питания: 3,0 ~ 5,0 В
* Выходное напряжение: 0 ~ 3,4 В
* Разъем датчика: BNC
* Сигнальный разъем: PH2.0-3Pin
* Точность измерения: ± 5% F.S.
* Размер платы: 42 мм \* 32 мм / 1,65 дюйма \* 1,26 дюйма

Электропроводный зонд:

* Тип зонда: лабораторный класс
* Ячейка Константа: 1,0
* Диапазон обнаружения поддержки: 0 ~ 20 мс / см
* Рекомендуемый диапазон обнаружения: 1 ~ 15 мс / см
* Диапазон температур: 0 ~ 40 ° C
* Срок службы зонда:> 0,5 года (в зависимости от частоты использования)
* Длина кабеля: 100 см

## Датчик уровня жидкости (угловой):



Внутри полого цилиндра находится герконовое реле, реагирующее на магнитное поле. Поплавок содержит крошечный магнит. При изменении уровня воды, положение поплавка меняется, заставляя реле замыкаться и размыкаться.

Датчик подключается двумя проводами: на «минус» и на сигнальный вывод Arduino или Iskra. При замыкании реле на сигнальном выводе появляется логический ноль, при размыкании — единица.

**Характеристики:**

* Габаритные размеры (Д×Ш×В): 54,5 × 24,5 × 35 мм
* Ход поплавка: 7 мм
* Диаметр отверстия под крепление: 11,5 мм

# Практическая часть

На текущий момент используются следующие компоненты:

* Arduino Uno (1 шт.)
* 4 Relay Shield (1 шт.)
* Перистальтический насос (3 шт.)
* Датчик уровня EC (1 шт.)
* Датчик уровня pH (1 шт.)
* Датчик уровня жидкости (3 шт.)
* Часы реального времени DS1307 (1 шт.)

Все компоненты монтируются на 3-пиновые порты, расположенные на плате расширения Shield + часы реального времени имеют собственный 4-пиновый порт.

Распиновка компонентов:

2 – Насос №1;

3 – Насос №2;

4 – Насос №3;

A0 – Датчик уровня pH;

A1 – Датчик уровня EC;

A2 – Датчик уровня для жидкости регулятора pH;

A3 – Датчик уровня для жидкости №1;

A4 – Датчик уровня для жидкости №2;

Компоненты взаимодействуют по следующему алгоритму:

Датчики pH и EC постоянно замеряют показатели жидкости в основной ёмкости. Если уровень концентрации достигает нижней допустимой границы, то насосы добавляют жидкости №1 и №2 в основную ёмкость. Если уровень pH достигает допустимой границы, то насос добавляет регулятор pH в основную ёмкость.

Код программы представлен в Приложении.

Была необходимость удлинить кабели для датчиков pH и EC. Был использован 3-жильный медный кабель длиной около 1,3 метра и перемычки “папа”, “мама” для плат Arduino.

Были сделаны замеры показателей датчиков до удлинения и после. В результате разницы в показателях обнаружено не было. Следовательно удлинение кабелей датчиков на погрешность не влияет.

# ПРИЛОЖЕНИЕ

Код программы:

#include <Wire.h>

#include "RTClib.h"

#include <Servo.h>

#include "DFRobot\_PH.h"

#include "DFRobot\_EC.h"

#include <EEPROM.h>

#define PUMPPIN1 2

#define PUMPPIN2 3

#define PUMPPIN3 4

#define PH\_PIN A0

#define EC\_PIN A1

#define LowLevelPH A2

#define LowLevelPump1 A3

#define LowLevelPump2 A4

Servo PumpPH;

Servo Pump1;

Servo Pump2;

RTC\_DS1307 rtc;

DFRobot\_PH ph;

DFRobot\_EC ec;

char daysOfTheWeek[7][12] = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"};

int ml = 1333;

int PHVolume;

int Pump1Volume;

int Pump2Volume;

int TimingPH;

int TimingPump1;

int TimingPump2;

float voltage1,phValue,temperature1 = 25;

float voltage2,ecValue,temperature2 = 25;

bool LowLevelPHState;

bool LowLevelPump1State;

bool LowLevelPump2State;

void setup () {

PumpPH.attach(PUMPPIN1);

Pump1.attach(PUMPPIN2);

Pump2.attach(PUMPPIN3);

while (!Serial); // for Leonardo/Micro/Zero

Serial.begin(115200);

pinMode(LowLevelPH, INPUT\_PULLUP);

pinMode(LowLevelPump1, INPUT\_PULLUP);

pinMode(LowLevelPump2, INPUT\_PULLUP);

ph.begin();

if (! rtc.begin()) {

Serial.println("Couldn't find RTC");

while (1);

}

if (! rtc.isrunning()) {

Serial.println("RTC is NOT running!");

// following line sets the RTC to the date & time this sketch was compiled

// rtc.adjust(DateTime(F(\_\_DATE\_\_), F(\_\_TIME\_\_)));

// This line sets the RTC with an explicit date & time, for example to set

//January 21, 2014 at 3am you would call:

//rtc.adjust(DateTime(2019, 7, 8, 13, 4, 0)); //(year,mnth,day,hr,m,s)

}

}

void loop () {

TimingPH = ml \* PHVolume;

TimingPump1 = ml \* Pump1Volume;

TimingPump2 = ml \* Pump2Volume;

LowLevelPHState = !digitalRead(LowLevelPH);

LowLevelPump1State = !digitalRead(LowLevelPump1);

LowLevelPump2State = !digitalRead(LowLevelPump2);

static unsigned long timepoint1 = millis();

if(millis()-timepoint1>1000U){ //time interval: 1s

timepoint1 = millis();

//temperature1 = readTemperature1(); // read your temperature sensor to execute temperature compensation

voltage1 = analogRead(PH\_PIN)/1024.0\*5000; // read the voltage

phValue = ph.readPH(voltage1,temperature1); // convert voltage to pH with temperature compensation

if (phValue<8 && !LowLevelPHState) {

PumpPH.write(0);

delay(TimingPH);

PumpPH.write(90);

}

}

ph.calibration(voltage1,temperature1); // calibration process by Serail CMD

static unsigned long timepoint2 = millis();

if(millis()-timepoint2>1000U) //time interval: 1s

{

timepoint2 = millis();

voltage2 = analogRead(EC\_PIN)/1024.0\*5000; // read the voltage

//temperature2 = readTemperature2(); // read your temperature sensor to execute temperature compensation

ecValue = ec.readEC(voltage2,temperature2); // convert voltage to EC with temperature compensation

if (ecValue<1000 && !LowLevelPump1 && !LowLevelPump2){

Pump1.write(0);

delay(TimingPump1);

Pump1.write(90);

Pump2.write(0);

delay(TimingPump2);

Pump2.write(90);

}

}

ec.calibration(voltage2,temperature2); // calibration process by Serail CMD

}