



ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
УПРАВЛЕНИЕ ЦИФРОВЫХ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЙ  
Кафедра «Автоматизация производственных процессов»

**Методические указания**  
«Датчики контроля и измерения уровня  
жидкости»  
по дисциплине

**«Промышленные датчики и  
КИП»**

Авторы  
Губанова А. А.,  
Золотарев В. В.,  
Котковец С. А.

Ростов-на-Дону, 2019

## Аннотация

Методические указания «Датчики контроля и измерения уровня жидкости» предназначены для студентов очной формы обучения направления 15.04.04 Автоматизация технологических процессов и производств и 28.03.02 Наноинженерия.

## Авторы

К.т.н., доц. кафедры АПП  
Губанова А.А.

Магистрант  
Золотарев В.В.

Магистрант  
Котковец С.А.



## Оглавление

<b>ЦЕЛЬ РАБОТЫ .....</b>	<b>4</b>
<b>КРАТКАЯ ТЕОРИЯ .....</b>	<b>4</b>
Классификация датчиков уровня жидкости .....	4
Контроллер Arduino UNO R3 .....	4
Учебный стенд .....	5
Управляющая программа .....	8
Листинг программы управления (для сигнализаторов) .....	9
Листинг программы управления (для уровнемера) .....	10
<b>ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ.....</b>	<b>12</b>
<b>СОДЕРЖАНИЕ ОТЧЁТА .....</b>	<b>13</b>
<b>КОНТРОЛЬНЫЕ ВОПРОСЫ ДЛЯ САМОПОДГОТОВКИ .....</b>	<b>14</b>
<b>СПИСОК ЛИТЕРАТУРЫ .....</b>	<b>15</b>

## ЦЕЛЬ РАБОТЫ

Изучение принципов работы датчиков уровня жидкости, освоение среды разработки Arduino IDE и программирования контроллера Arduino UNO R3.

## КРАТКАЯ ТЕОРИЯ

### Классификация датчиков уровня жидкости

Датчики – это устройства, которые преобразовывают какую-либо физическую величину (температуру, силу, уровень жидкости и т.д.) в форму, понятную человеку или машине. С их помощью программа микроконтроллера получает информацию об окружающей среде и на основе её принимает решения. Датчики бывают разных типов. В данной практической работе будут рассмотрены датчики контроля уровня жидкости.

Данные датчики предназначены для измерения и контроля уровня жидкости в ёмкостях и трубопроводах. Функционально они подразделяются на уровнемеры и сигнализаторы. Первые необходимы для непрерывного контроля уровня жидкости, вторые, соответственно, предназначены для определения заданного положения уровня (заполнения или опустошения).

Основное назначение уровнемеров базируется на преобразовании значения уровня жидкости в пропорциональный аналоговый или цифровой сигнал. Сигнализаторы – имеют дискретный (релейный или транзисторный) выходной сигнал и срабатывают при блокировании или освобождении чувствительного элемента жидкостью.

По типу взаимодействия с измеряемой жидкостью рассматриваемые датчики бывают контактными и бесконтактными (применяются для измерения уровня вязких и агрессивных жидкостей).

Помимо функционального назначения и типа взаимодействия, датчики уровня жидкости различаются по принципу действия. Они бывают: ёмкостными, гидростатическими, ультразвуковыми, микроволновыми, оптическими, поплавковыми, кондуктометрическими и т.д.

### Контроллер Arduino UNO R3

В данной практической работе для сбора данных с датчиков

используется контроллер Arduino UNO R3. Данное устройство управления построено на основе чипа ATmega328с тактовой частотой 16 МГц, обладает памятью 32кБ и имеет 20 контролируемых контактов ввода/вывода для взаимодействия с внешними устройствами.

Контроллер может работать при наличии напряжения от 6 до 20 В. Однако при напряжении менее 7В работа может быть неустойчивой, а напряжение более 12В может привести к повреждению. Поэтому рекомендуемый диапазон: 7–12 В.



Рисунок 1 – Контроллер Arduino UNO R3

Программирование данной платы осуществляется в среде разработки Arduino IDE, в которой осуществляется как написание программы управления, так и загрузка данных на контроллер и осуществление мониторинга параметров в реальном времени с помощью пункта «Монитор порта».

### Учебный стенд

Для изучения принципа работы датчиков уровня жидкости был собран макет системы, осуществляющей в автоматическом режиме процесс очистки воды.

Макет системы автоматического управления процессом очистки воды состоит из: устройства управления (УУ), четырех объектов управления (ОУ), сплиттера, силового ключа и блока питания (БП). Также макет имеет стойку и два бака для жидкости

с неочищенной и очищенной водой.

В роли УУ выступает контроллер Arduino UNO R3. У контроллера задействовано три дискретных вывода, а также три выходы GND.

ОУ представляют из себя два датчика уровня жидкости поплавкового типа, фильтр и насос для подачи воды. Один из датчиков отвечает за нижний уровень отфильтрованной воды, второй – за верхний. При достижении жидкости нижнего уровня, датчик фиксирует, что бак опустел и передает сигнал на насос, который, в свою очередь подаёт воду с первого бака на фильтр, в котором осуществляется процесс фильтрации. В результате, очищенная вода поступает во второй бак. В процессе наполнения бака, достигается верхний уровень, и второй датчик подаёт сигнал на выключение насоса.

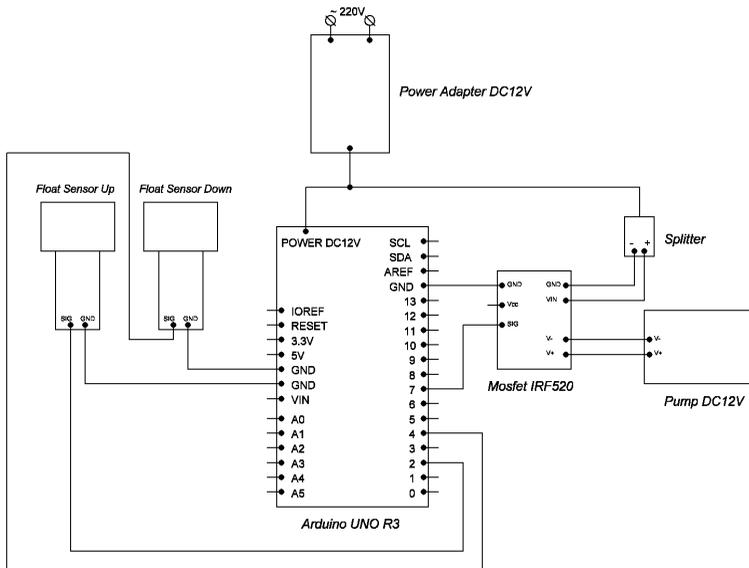


Рисунок 2 – Схема электрических соединений (с поплавковыми датчиками)

Для реализации этого функционала датчики и насос используют дискретные выводы УУ. Датчик верхнего уровня использует второй дискретный вывод, конфигурируемый как вход (для приёма сигнала). Датчик нижнего уровня, аналогично верхнему, использует вывод в качестве входа, но подключен уже к

четвертому пину.

Насос подключен к седьмому дискретному выводу, обозначенному как выход. Фильтр соединён с насосом с помощью трубки, по которой осуществляется поток фильтруемой воды. Каждый из ОУ, кроме фильтра, имеет свой вывод GND.

Питание системы осуществляется от одного БП с выходным напряжением DC12V и силой тока 4 А. Ток распараллеливается с помощью сплиттера, одна линия идёт на контроллер Arduino UNO R3, другая, для подачи питания на насос, схема электрических соединений системы представлена на рис. 2. Для подачи управляющего сигнала на насос, в системе имеется силовой ключ.

Помимо датчиков поплавкового типа (являющихся сигнализаторами) в системе также могут использоваться уровнемеры. Для реализации работы учебного стенда достаточно лишь одного датчика данного типа, который будет считывать текущий уровень жидкости и передавать на контроллер, где будет происходить сравнение с граничными уровнями. Схема электрических соединений с подключенным уровнемером представлена на рис. 3.

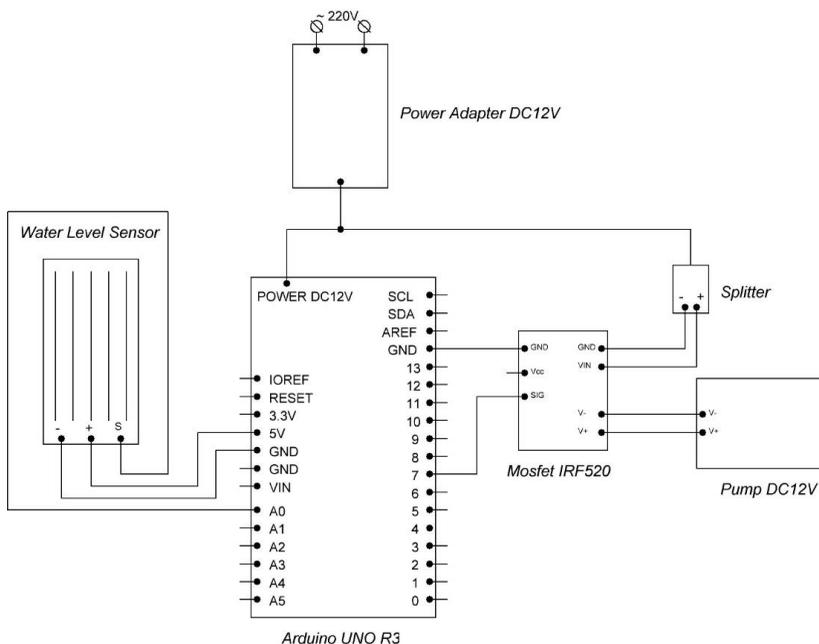


Рисунок 3 – Схема электрических соединений (с уровнемером)

## Управляющая программа

Создание ПО для учебного стенда было осуществлено в интегрированной среде разработки ArduinoIDE. Программа управления состоит из трех основных частей – подпрограмм.

В первой подпрограмме осуществляется объявление переменных, содержащих номера портов подключенных элементов АСУ. Для поплавкового сенсора верхней границы присвоен второй дискретный вывод (переменная `up_sensor_pin`), для сенсора нижней границы – четвертый дискретный вывод (переменная `down_sensor_pin`). Управление насосом осуществляется через седьмой дискретный вывод (переменная `pinp_r_pin`).

В опыте с уровнемером присваивается А0 аналоговый вывод (`water_level_sensor_pin`).

Вторая подпрограмма – `Setup`, осуществляет инициализацию используемых портов и передачи данных. Порт, отвечающий за датчик верхней границы жидкости, инициализирован как вход, причём через внутренний подтягивающий резистор. Это сделано для того, чтобы не было неопределённости в состоянии датчика в реальном времени (0 или 1), так как резистор компенсирует малое напряжение на датчике. Порт, отвечающий за датчик нижней границы, аналогично «верхнему», сконфигурирован как вход с использованием подтягивающего резистора. Вывод насоса для подачи воды инициализирован как выход.

В схеме с подключенным уровнемером – его порт инициализирован как аналоговый вход.

Третья подпрограмма – `Loop`, является основной. Она отвечает за принцип работы разрабатываемой системы. Также, в ней осуществляется вывод данных в монитор порта. Данная подпрограмма имеет ветвистую структуру, зависящую от выполнения определённых условий. Если на входах обоих датчиков зафиксирована логическая единица (т.е. датчики погружены в воду), то в монитор порты выводится надпись «БАК НАПОЛНЕН», и, в случае если работает насос, его питание прекращается. Обратно этому, когда на выходах обоих датчиков зафиксирован логический ноль – датчики не находятся в воде, в монитор порта поступает надпись «БАК ПУСТ», и подаётся питание на насос, для выполнения им подачи вода на фильтр. Если не одно из этих условий не зафиксировано, то подпрограмма `Loop` продолжает свою работу циклично, не изменяя стационарного состояния системы, до тех пор, пока не будет выполнено одно из двух описанных условий.

Для уровнемера условия аналогичны, но зависят от текуще-

го уровня жидкости (изменения сопротивления на чувствительном элементе сенсора), на который погружен датчик уровня. Сопротивление менее 250 – подаётся питание на насос, более 250 насос отключается.

Листинги программ управления для сигнализаторов и уровнемера представлены ниже.

### Листинг программы управления (для сигнализаторов)

```
// Объявление переменных, содержащих номера портов подключенных
элементов АСУ
int up_sensor_pin = 2; // верхний датчик, подключенный к 2-му
выводу
int down_sensor_pin = 4; // нижний датчик, подключенный к 4-му
выводу
int pump_pin = 7; // насос, подключенный к 7-му выводу
// Объявление используемых портов (процедура setup запускается
один раз)
void setup() {
    Serial.begin(9600); // Иницируем передачу данных по последо-
вательному порту на скорости 9600 бит/сек
    pinMode(up_sensor_pin, INPUT_PULLUP); // конфигурируем вывод
up_sensor как вход, подтягивая его до уровня логического «0» че-
рез внутренний подтягивающий резистор
    pinMode(down_sensor_pin, INPUT_PULLUP); // конфигурируем вывод
down_sensor как вход, подтягивая его до уровня логического «0»
через внутренний подтягивающий резистор
    pinMode(pump_pin, OUTPUT); // конфигурируем вывод насоса как
выход
}
// Основная программа процесса очистки воды (процедура loop за-
пускается циклично)
void loop() {
    int value_up_sensor = digitalRead(up_sensor_pin); // присвае-
ваем переменной value_up_sensor текущее значение up_sensor_pin
    int value_down_sensor = digitalRead(down_sensor_pin); // при-
сваеваем переменной value_down_sensor текущее значение
```

```

down_sensor_pin
    Serial.println(value_up_sensor);
    Serial.println(value_down_sensor);
    if ((value_down_sensor == 1) and (value_up_sensor == 1)) //
если на входе down_sensor и up_sensor уровень логической «1», то
...
    {
        Serial.println("БАК НАПОЛНЕН"); // выводим сообщение о том,
что БАК НАПОЛНЕН
        digitalWrite(pump_pin, LOW); // выключаем насос
    }
    else if ((value_down_sensor == 0) and (value_up_sensor == 0))
// если на входе down_sensor и up_sensor уровень логического
«0», то ...
    {
        Serial.println("БАК ПУСТ"); // выводим сообщение о том что
БАК ПУСТ
        digitalWrite(pump_pin, HIGH); // включаем насос
    }
// Последующее выполнение основной программы через 500 мс
delay(500);
}
    
```

## Листинг программы управления (для уровнемера)

```

// Объявление переменных, содержащих номера портов подключенных
элементов АСУ
int water_level_sensor_pin = A0; // датчик уровня (уровнемер)
жидкости, подключенный к А0 аналоговому выводу
int pump_pin = 7; // насос, подключенный к 7-му выводу
// Объявление используемых портов (процедура setup запускается
один раз)
void setup() {
    Serial.begin(9600); // Иницилируем передачу данных по последо-
вательному порту на скорости 9600 бит/сек
    pinMode(A0, INPUT); // конфигурируем вывод уровнемера как
    
```

## Промышленные датчики и КИП

```
ВХОД
    pinMode(pump_pin, OUTPUT); // конфигурируем вывод насоса как
ВЫХОД
}
// Основная программа процесса очистки воды (процедура loop за-
пускается циклично)
void loop() {
    int value_water_level_sensor_pin =
analogRead(water_level_sensor_pin); // присваиваем переменной
value_water_level_sensor_pin текущее значение wa-
ter_level_sensor_pin
    Serial.println(value_water_level_sensor_pin);
    if (value_water_level_sensor_pin > 250) // если на входе wa-
ter_level_sensor_pin значение более 250, то ...
    {
        Serial.println("БАК НАПОЛНЕН"); // выводим сообщение о том
что БАК НАПОЛНЕН
        digitalWrite(pump_pin, LOW); // выключаем насос
    }
    else if (value_water_level_sensor_pin < 250) // если на входе
water_level_sensor_pin значение менее 250, то ...
    {
        Serial.println("БАК ПУСТ"); // выводим сообщение о том что
БАК ПУСТ
        digitalWrite(pump_pin, HIGH); // включаем насос
    }
// Последующее выполнение основной программы через 500 мс
delay(500);
}
```

## ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Ознакомиться с изложенным выше теоретическим материалом.
2. Переписать исходный код программы управления (для сигнализаторов) в среду разработку Arduino IDE и загрузить в контроллер.
3. Произвести подключение основных компонентов системы согласно схеме электрических соединений (рис. 2).
4. Подключить макет (с заранее наполненными водой ёмкостями) в сеть.
5. Путём манипуляций поплавковыми датчиками верхнего и нижнего уровня эмулировать нехватку воды во втором баке, а также его переполнение. Параллельно наблюдая за данными в мониторе порта и за работой стенда. После произведённого эксперимента – отключить питание системы.
6. Произвести те же действия, только без функции PULLUP в программе управления.
7. Переписать исходный код программы управления (для уровнемера) в среду разработку Arduino IDE и загрузить в контроллер.
8. Аналогично пунктам 3-5.

## **СОДЕРЖАНИЕ ОТЧЁТА**

1. Название и цель работы
2. Рисунки схем электрических соединений
3. Описание опытов и данных, полученных в результате их проведения в мониторе порта.
4. Основные выводы по выполненной практической работе.

## КОНТРОЛЬНЫЕ ВОПРОСЫ ДЛЯ САМОПОДГОТОВКИ

1. Опишите конструктивно составляющие поплавкового датчика. Что определяет уровень поверхности жидкой среды? Что передает данные о степени наполнения резервуара в блок управления или управляет механической системой его наполнения?
2. На каком законе основана подъемная сила поплавок?
3. Какие факторы влияют на точность работы уровнемера, описанного в данной работе?
4. Какие типы датчиков можно рассмотреть в качестве датчиков уровня применительно для рассматриваемой работы? Какие недостатки имеют поплавковые датчики?
5. Перечислите факторы, влияющие на выбор датчика уровня воды в резервуаре.
6. Что в качестве коммутирующего элемента используется в датчиках уровня жидкости поплавкового типа?
7. Чем отличается электрический датчик уровня жидкости от механического?

## СПИСОК ЛИТЕРАТУРЫ

1. **Arduino, датчики и сети для связи устройств:** Пособие / Иго Т., Таранушенко С., - 2-е изд. - СПб:БХВ-Петербург, 2015. - 544 с.
2. **Основы измерений. Датчики и электронные приборы:** Учебное пособие / Клаассен К.Б., Воронов Е.В., Ларин А.Л., - 4-е изд. - Долгопрудный:Интеллект, 2012. - 352 с.:
3. **Электроника. Логические микросхемы, усилители и датчики. Для начинающих:** Пособие / Платт Ч. - СПб:БХВ-Петербург, 2015. - 448 с.