

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Тольяттинский государственный университет»

Институт машиностроения  
(наименование института полностью)

Кафедра «Промышленная электроника»  
(наименование)

11.03.04 Электроника и нанoeлектроника  
(код и наименование направления подготовки / специальности)

Промышленная электроника  
(направленность (профиль) / специализация)

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА  
(БАКАЛАВРСКАЯ РАБОТА)**

на тему      Автоматизированный стеллаж для гидропоники

---

Обучающийся

В.А. Абрамов

(Инициалы Фамилия)

\_\_\_\_\_  
(личная подпись)

Руководитель

И.И. Золотов

\_\_\_\_\_  
(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Тольятти 2025

## **Аннотация**

Тема представленной выпускной квалификационной работы «Автоматизированный стеллаж для гидропоники».

Рассматриваемая бакалаврская работа включает в себя пояснительную записку на 60 листах включающую в себя 56 рисунков, 2 приложения, списка литературы из 20 источников, в том числе 8 на иностранном языке и графическую часть на 6 листах формата А1.

В первой части работы был выполнен анализ готовых технических решений в области автоматизированных систем гидропоники, представленных на рынке в текущий момент? На основе полученных данных была определена основная концепция нашего проекта и выделен список задач, которые необходимо решить для достижения поставленной цели.

Во второй части работы были рассмотрены варианты выбора и установки аппаратного обеспечения системы управления. Было разработано и промоделировано схмотехническое решение цепей обеспечения питания системы от сети 220 В. Был произведен подбор всей необходимой компонентной базы системы управления.

В третьей части работы была рассмотрена реализация управляющей программы автоматизированной системы, специализированных разработанных функций и алгоритмов работы. Была рассмотрена разработанная система HMI и общая компоновка разработанного устройства.

Целью выполнения данной выпускной квалификационной работы является автоматизация и оптимизация процесса выращивания полезных культур с использованием гидропоники, по средствам создания конечного программно-аппаратного продукта, оснащенного всем необходимым функционалом и имеющим более приемлемую ценовую характеристику по сравнению с существующими аналогами.

## Содержание

|                                                                                                        |    |
|--------------------------------------------------------------------------------------------------------|----|
| Введение .....                                                                                         | 4  |
| 1 Состояние вопроса .....                                                                              | 5  |
| 1.1 Общие понятия о гидропонике .....                                                                  | 5  |
| 1.2 Анализ технического задания и определение концепции .....                                          | 15 |
| 2 Разработка автоматизированного стеллажа для гидропоники .....                                        | 17 |
| 2.1 Система Power supply .....                                                                         | 17 |
| 2.2 Аппаратное обеспечение системы управления .....                                                    | 26 |
| 2.3 Системы контроля и поддержания микроклимата .....                                                  | 33 |
| 2.4 Разработка системы НМІ .....                                                                       | 44 |
| 3 Разработка управляющей программы .....                                                               | 51 |
| Заключение .....                                                                                       | 58 |
| Список используемой литературы .....                                                                   | 59 |
| Приложение А Перечень элементов к схеме электрической<br>принципиальной стеллажа для гидропоники ..... | 61 |
| Приложение Б Листинг участка управляющей программы .....                                               | 63 |

## **Введение**

В современном мире автоматизация различных производственных и повседневных процессов является неотъемлемой частью жизни общества.

Активное применение автоматизированные системы сыскали и в области промышленного сельского хозяйства и частного садоводства, как одних из древнейших сфер деятельности человека. Автоматизированные системы полива и контроля микроклимата в разы упрощают и совершенствуют процесс выращивания полезных культур.

Наряду с аппаратными и программными решениями совершенствуются и технологии выращивания, таким образом, в настоящее время активно развивается такое направление сельского хозяйства, как гидропоника – выращивание культур без использования привычного грунта. Применение гидропоники значительно снижает затраты на обработку почвы, обработку посевов от вредителей, дополнительно обеспечивая возможность выращивания большого количества растений на относительно ограниченных площадях.

Автоматизация процесса выращивания полезных культур с применением гидропоники позволяет кратно увеличить эффективность и результативность работы в условиях ограниченной площади целевого помещения с минимальным вмешательством и вложениями собственника.

Целью выполнения данной выпускной квалификационной работы является автоматизация и оптимизация процесса выращивания полезных культур с использованием гидропоники, по средствам создания конечного программно-аппаратного продукта, оснащенного всем необходимым функционалом и имеющим более приемлемую ценовую характеристику по сравнению с существующими аналогами.

## 1 Состояние вопроса

### 1.1 Общие понятия о гидропонике

Гидропоника в обобщенном представлении является способом выращивания полезных культур в искусственной среде, без использования привычных методов и почвы. Питание выращиваемых культур осуществляется при помощи подачи специализированного питательного раствора. Применение гидропоники позволяет регулировать и адаптировать условия выращивания растений путем создания режима питания корневой системы, контроля содержания углекислого газа в газовой смеси, а также регулирования интенсивности освещения и температуры [1]. На рисунке 1 представлен пример выращивания полезных культур при помощи гидропоники.



Рисунок 1 – Выращивание полезных культур при помощи гидропоники

«При выращивании полезных культур гидропонным способом корни растения черпают питательные вещества не из почвы, а из влажно-воздушной среды, сильно аэрируемой водой, также распространено использование твердой, но пористой влаго-воздухоемкой среды, которая способствует дыханию корней и требует регулярного капельного полива питательным раствором. Корневая система полезных культур, выращиваемых таким способом развивается на твердых субстратах, которые сами по себе не имеют питательного значения, а весь микроклимат и поставка питательных вещества организуется искусственным методом, как показано на рисунке 2» [3].



Рисунок 2 – Искусственная среда для выращивания растений

«Одним из примеров органического субстрата, используемого в гидропонике, является кокосовое волокно. Волокно представляет из себя перемолотую кокосовую скорлупу и орех, что за счет своих свойств является идеальной средой для взращивания побегов» [6]. На рисунке 3 представлен пример использования такого органического субстрата.



Рисунок 3 – Использование кокосового субстрата

«В естественной среде кокосовое волокно служит первоначальным грунтом для корней новорожденной пальмы. Кокосовое волокно легче воды, поэтому при поливе не утапливается как почвогрунт, а разбухает, наполняясь воздухом. Каждое волокно содержит в своей толще большое количество пор и канальцев. Силой поверхностного натяжения канальцы заполняются рабочим раствором, но корневой волосок выпивает содержимое, прорастая рядом. Гладкая поверхность волокна позволяет корню свободно скользить от выпитой микропоры к следующей. Сетью микротрубочек кокосовое волокно распределяет воду и воздух по всему своему объёму. Волокно кокоса, как полностью рекультивируемый, экологически дружелюбный субстрат, используется на многих голландских гидропоник-фермах, при выращивании многолетних растений, например, роз. В Голландии промышленная гидропоника имеет обширное распространение, несмотря на климатические особенности — светолюбивым культурам солнце заменяют специальные лампы ДНаТ и Grow Led» [7].

Применение гидропоники значительно снижает затраты на обработку почвы, обработку посевов от вредителей, дополнительно обеспечивая возможность выращивания большого количества растений на относительно ограниченных площадях, а автоматизация процесса выращивания полезных культур с применением гидропоники позволяет кратно увеличить эффективность и результативность работы в условиях ограниченной площади целевого помещения с минимальным вмешательством и вложениями собственника. Помимо того, к достоинствам гидропоники можно отнести высокую производительность растений и их ускоренный рост, ввиду отсутствия многих неблагоприятных факторов, зависящих от природы [9].

Несмотря на то, что гидропоника является одной из самых популярных способов альтернативного земледелия, она может подойти не под все существующие культуры, условия выращивания и возможности человека, поэтому стоит отметить, что существуют еще такие способы и разновидности, как аэропоника, метод глубоководной культуры и фитильная система [8].

«Аэропоника - это относительно новая и инновационная система выращивания растений в помещении. При данном подходе корни растений находятся в воздушной среде, а питательный раствор наносится на них в виде тонких водяных капель. Этот метод позволяет растениям получать все необходимые питательные вещества, однако в условиях сухого воздуха. Аэропоника также позволяет достичь максимальной производительности, улучшить качество урожая и сэкономить воду. Выбор системы выращивания в помещении зависит от предпочтений фермера или садовода, а также от типа растений, которые он планирует выращивать. Грунт, гидропоника и аэропоника имеют свои достоинства и недостатки, но все они являются эффективными методами выращивания растений в закрытом помещении, позволяющими получить качественный и обильный урожай круглый год» [10].

Основные способы и разновидности альтернативного выращивания полезных культур представлены на рисунке 4.

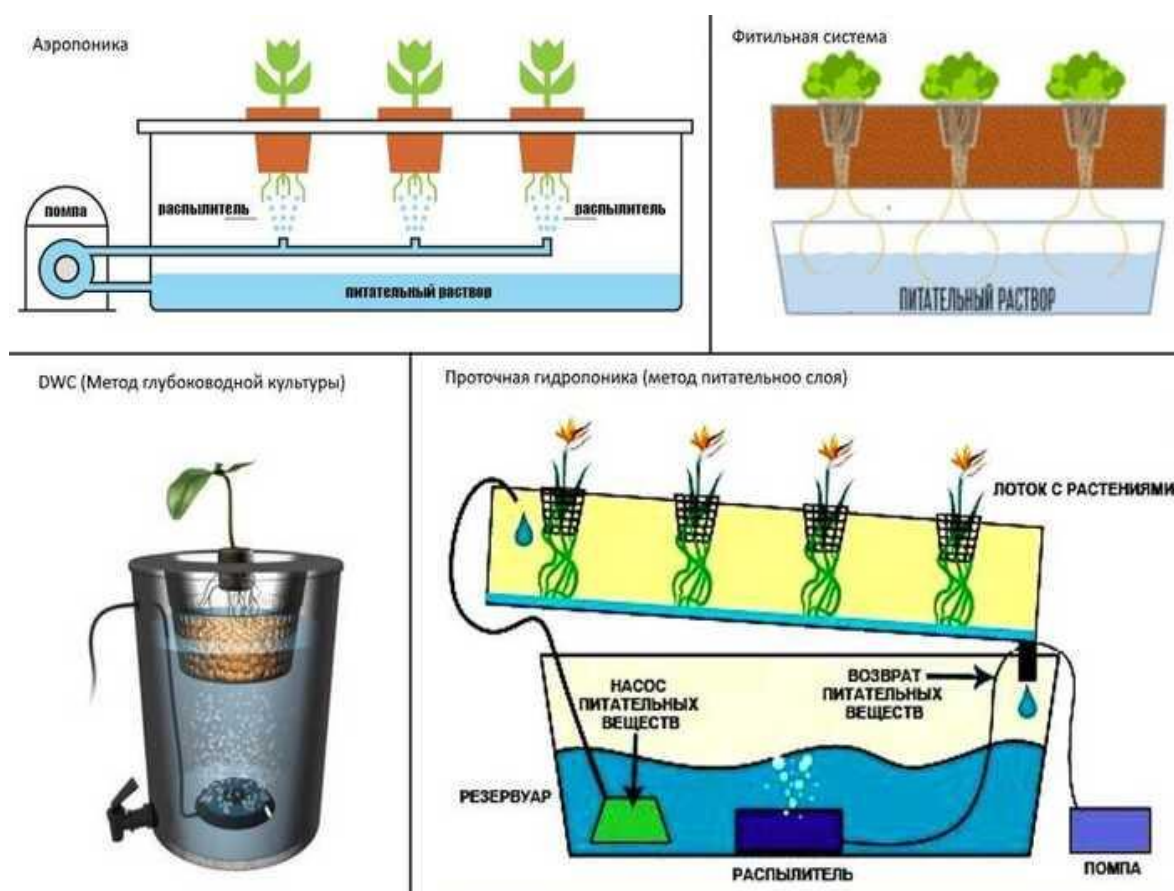


Рисунок 4 – Разновидности способов выращивания полезных культур

Обзорно рассмотрим другие представленные виды альтернативного земледелия. «К примеру, фитильная гидропоника является пассивной системой, полезная культура помещается в субстрат, а питательный раствор поставляется к корневой системе за счет действия капиллярных сил. Такой же способ зачастую используется при орошении цветочных горшков в магазинах. Данная система является хорошим примером автоматизации процесса без использования сторонних технических средств, однако эффективность и сфера применения таких систем сильно ограничена и как правило может применяться для выращивания декоративных растений, растущих медленно и не требующих особого ухода» [8].

«Методом глубоководной культуры называется вид гидропонных систем, состоящих из бака и устройства для аэрации воды. При таком методе выращивания корни растений полностью погружают в питательный раствор, а установленный компрессор осуществляет аэрацию раствора. Преимущество данной системы заключается в том, что происходит активное снабжение кислородом корневой зоны растения, а компрессор одновременно осуществляет как аэрацию, так и капельный полив» [8].

«Корни растений в DWC-системах обычно заполняют свободное пространство по максимуму и растения растут намного быстрее, чем в почве. К методу водной культуры также относятся и плавающие платформы. Это огромные прямоугольные бассейны, наполненные питательным раствором, в которых плавают пенопластовые плоты с рассадой салата. Один и тот же раствор, добавляя воду и удобрения для баланса, используют весь сезон. Регулярно на одном конце бассейна снимают урожай салата и передвигают остальные плоты, чтобы высадить новую партию рассады, это непрерывный процесс» [8].

Одним из самых простых способов реализации гидропонных систем является NTF. При использовании данного метода питательный раствор постоянно циркулирует по замкнутому контуру системы и обеспечивает большую площадь соприкосновения питательного раствора с воздушной средой, а для создания корпуса под посадку растений достаточно будет использовать простые пластиковые водосточные трубы.

В связи с достаточно активным развитием такого направления как гидропоника и роста его популярности как среди коммерческих лиц, так и среди частных на рынке существует достаточное количество готовых систем и стеллажей для гидропоники с разным функционалом, целевым назначением и ценой. На рисунке 5 представлен пример готового стеллажа для гидропоники.



Рисунок 5 – Стеллаж для гидропоники

Представленная выше установка предназначена для выращивания полезных культур методом проточной гидропоники, что хорошо подходит для выращивания рассады и разнообразных низкорослых растений. Конструкция предусматривает 5 ярусов с поддонами, на которых можно разместить емкости с субстратом или горшки. Установка может дополнительно комплектоваться простейшей системой пассивного поддержания микроклимата – включение освещения и полива по заданному времени. На рисунке 6 представлен комплект поставки, предусмотренный с данной установкой.

|                                                                             |
|-----------------------------------------------------------------------------|
| <b>Бак на 70 литров 1шт</b>                                                 |
| <b>Помпа 6000л час 1шт</b>                                                  |
| <b>Система подтопления слив-залив (фитинги, трубы, штуцеры,краны)</b>       |
| <b>Поддоны для проращивания 1250*560*45мм 5шт</b>                           |
| <b>Таймер включения/выключения системы освещения и подтопления 2шт</b>      |
| <b>Кассеты для проращивания(по запросу) до 15шт</b>                         |
| <b>Стаканчики для проращивания(по запросу) до 300шт</b>                     |
| <b>Освещение LED 20w IRUVS 10шт</b>                                         |
| <b>Система аэрирования бака компрессор двуканальный+шланг + 2 аэрокамня</b> |

Рисунок 6 – Предусмотренный комплект поставки

Стоимость такой установки составляет порядка 75 тысяч рублей без учета расходных элементов, что при условии отсутствия систем активного контроля микроклимата, полностью автоматизированного полива и освещения является большим недостатком. Включение освещения полезных культур и полив растений в данной системе опционально осуществляются за счет реле времени, что является недостаточным уровнем автоматизации при выставленной стоимости оборудования и не исключает участие человека в процессе выращивания и отладке системы.

Примером полностью автоматизированной системой гидропоники может являться продукция компании Greenfeya, представленная на рисунке 7.



Рисунок 7 – Автоматизированный стеллаж для гидропоники от компании Greenfeya

Представленные установки уже являются примером реализации полного цикла автоматизации выращивания полезных культур при помощи гидропоники. В гидропонном стеллаже автоматически функционирует система полива, освещения, аэрации питательного раствора, стерилизации воды и обдува, однако, все еще отсутствуют системы активного контроля микроклимата.

Блок управления данного устройства оснащен Wi-Fi модулем, что предоставляет пользователю возможность проводить настройку и управление системой дистанционно, при помощи специализированного приложения на телефоне или подключить стеллаж к существующей системе умного дома. Помимо того, блок управления не зависит от наличия сети и способен работать полностью автономно, по заданным ранее уставкам и выбранным режимам. На рисунке 8 представлен пример реализации шкафа управления для данного устройства.

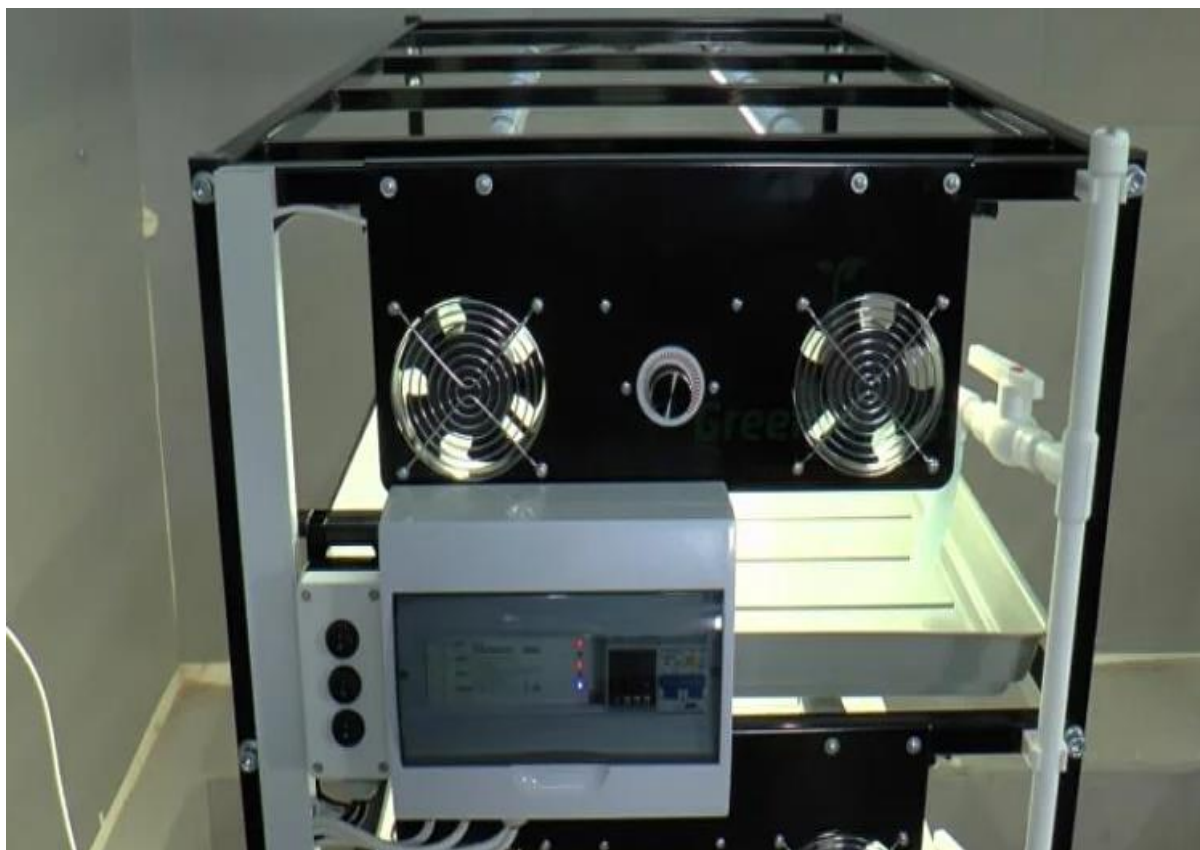


Рисунок 8 – Шкаф управления на корпусе стеллажа

К недостаткам данной системы все еще можно отнести отсутствие гибкости автоматизированной системы и отсутствию систем активного контроля в режиме реального времени, а также стоимость оборудования, варьирующуюся от 150 до 200 тысяч рублей, что значительно сужает круг потенциальных пользователей среди физических лиц.

Помимо вышеперечисленных готовых решений на рынке также существуют более бюджетные устройства для гидропоники, цена которых варьируется от 3 до 10 тысяч рублей, однако, данные системы в подавляющем большинстве представляют собой просто собранный каркас с посадочными местами под растения и опционально проложенную трассу для полива без внедрения каких-либо электронных и электрических устройств.

## **1.2 Анализ технического задания и определение концепции проекта**

Для последующего выполнения выпускной квалификационной работы была изучена информация о системах гидропоники, их разновидностях и способах реализации. Помимо того, был произведен анализ рынка готовых технических решений в области гидропонных систем. По результатам анализа были выделены преимущества и недостатки исследуемых устройств. Вся информация, полученная в ходе анализа принципа построения и работы гидропонных систем, а также при анализе готовых решений, представленных на рынке, будет использована при проектировании автоматизированного стеллажа для гидропоники в рамках выполнения данной выпускной квалификационной работы.

Согласно выданному техническому заданию, в ходе выполнения нашей выпускной квалификационной работы необходимо разработать техническое решение для физической реализации автоматизированного стеллажа для гидропоники. Напряжение питания цепей управления, а также выбранного устройства управления должна составлять 5 В. В качестве устройства управления необходимо использовать микроконтроллер.

Разработанная автоматизированная система управления и ее программно-аппаратная реализация должны отличаться гибкостью и сохранять весь необходимый для работы функционал при более низкой себестоимости, по сравнению с существующими аналогами. Все разработки, алгоритмы работы и схемотехнические решения, полученные в ходе выпускной квалификационной работы, должны представлять исчерпывающую информацию для последующей сборки и эксплуатации автоматизированного стеллажа для гидропоники.

В разрабатываемое программно-аппаратное решение было решено внедрить системы активного контроля влажности субстрата, уровня освещенности, температуры и влажности окружающей среды, автоматическую систему полива и освещения выращиваемых культур, а также удаленную и бортовую систему диспетчеризации всех необходимых данных.

После проведения всех необходимых подготовительных процедур и анализа существующих аналогов, представленных в текущий момент на рынке, а также определения основной концепции выполняемого проекта был составлен список задач, которые необходимо решить для достижения поставленной цели:

- разработка структуры устройства;
- проработка необходимых функциональных связей между элементами;
- выбор необходимой компонентной базы;
- проработка необходимых схемотехнических решений;
- проработка алгоритмов работы системы;
- разработка управляющей программы;
- разработка системы диспетчеризации и человеко-машинного интерфейса.

#### Выводы по разделу

После постановки цели, которую необходимо достичь в рамках выполнения данной выпускной квалификационной работы был проведен анализ информации, связанной с гидропонными системами, их разновидностями и принципами работ. Был выполнен анализ готовых технических решений в области автоматизированных систем гидропоники, представленных на рынке в текущий момент. На основе полученных данных была определена основная концепция нашего проекта и выделен список задач, которые необходимо решить для достижения поставленной цели и успешной реализации проекта в рамках выполнения данной выпускной квалификационной работы.

## **2 Разработка автоматизированного стеллажа для гидропоники**

### **2.1 Система Power supply**

Из технического задания на выполнение выпускной квалификационной работы нам известно, что питание всех основных цепей управления и системы управления в целом должны осуществляться от напряжения 5 В DC, что делает невозможным работу системы, к примеру, напрямую от сетевого напряжения. В связи с этим возникает необходимость разработки технического решения по обеспечению системы управления необходимым рабочим напряжением. Готовые источники напряжения 5 В DC как правило используются в лабораториях, сервисных центрах и прочих специализированных служебных и производственных помещениях. Приобретение лабораторного источника напряжения для штатного функционирования разрабатываемой системы неизбежно несет дополнительную финансовую нагрузку на конечного потребителя.

Исходя из всего вышеперечисленного, было решено использовать в качестве входного напряжения штатное сетевое напряжение 220 В AC, что делает возможным установку нашей системы возможной практически в любом бытовом, складском или производственном помещении. Для реализации такой функции на данном этапе выполнения выпускной квалификационной работы было решено разработать AC-DC преобразователь напряжения 220-5 В.

При проектировании и проверке цепи питания автоматизированного стеллажа для гидропоники использовалась программное обеспечение Micro-Cap. Данная среда проектирования применяется при необходимости проектирования, отладки, и тестирования электрических схем и составных частей различных устройств. Рассматриваемое программное обеспечение включает в себя функцию редактирования пассивных и активных элементов радиоэлементов, создания собственных и работу с базовой библиотекой готовых компонентов, помимо этого, Micro-Cap предоставляет доступ к

инструментам анализа по переменному и постоянному току, а также анализу переходных процессов и временных осциллограмм.

На первом этапе проектирования системы power supply необходимо было снизить входное сетевое напряжение до необходимого нам значения (порядка 15-30 В), что обеспечит вариативность и простоту дальнейшей разработки системы и работы с полученным напряжением.

Для реализации данной задачи идеально подходит такое электромагнитное устройство, как трансформатор, преобразующий входные величины тока и напряжения в величины, необходимые пользователю, за счет действия электромагнитной индукции. На рисунке 9 представлен обобщенное строение и принцип работы трансформатора.

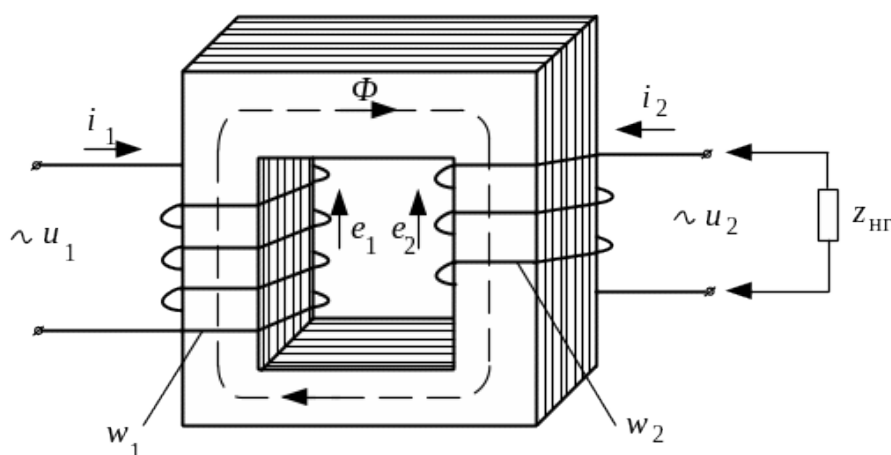


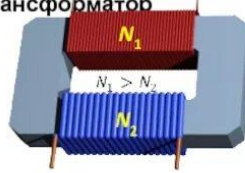
Рисунок 9 – Принцип работы трансформатора

Трансформаторы в простейшем функциональном представлении делятся на понижающие и повышающие. «Физически этот процесс связан с количеством витков на первичной и вторичной обмотках и их соотношением, которое выражается в коэффициенте трансформации трансформатора. Если коэффициент трансформации больше единицы, то трансформатор считается понижающим, а если меньше, то повышающим, как показано на рисунке как показано на рисунке 10» [6].

Коэффициент трансформации ( $k$ ) — отношение числа витков в первичной обмотке к числу витков во вторичной обмотке.

$$k = \frac{N_1}{N_2}$$

Понижающий  
трансформатор



Повышающий  
трансформатор

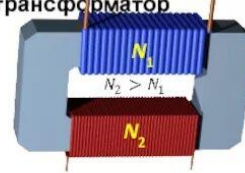


Рисунок 10 – Определение коэффициента трансформации

В рамках текущего проекта нам необходимо провести преобразование напряжения АС-АС из диапазона 220 В в рабочее напряжение порядка 30 В, как наиболее унифицированное для подбора и работы последующих функциональных модулей. Исходя из этого нам необходимо использовать понижающий трансформатор с коэффициентом трансформации порядка 7.

Для проверки рассчитанного коэффициента трансформации и работы схемы в среде Micro-Cap была собрана простейшая электрическая схема преобразователя АС-АС, состоящая из трансформатора с необходимым нам коэффициентом трансформации и источника напряжения 220 В АС 50 Гц, имитирующего сетевое напряжение, получившаяся схема представлена на рисунке 11.

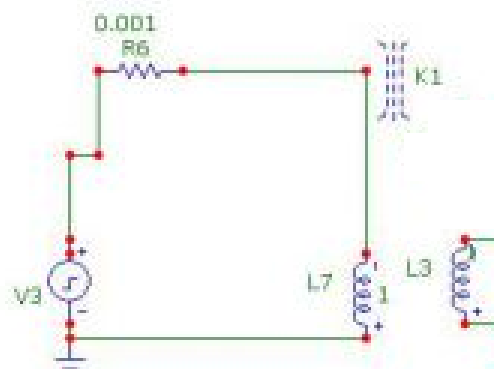


Рисунок 11 – Схема АС-АС преобразователя

На рисунке 12 представлена осциллограмма работы исследуемой схемы.

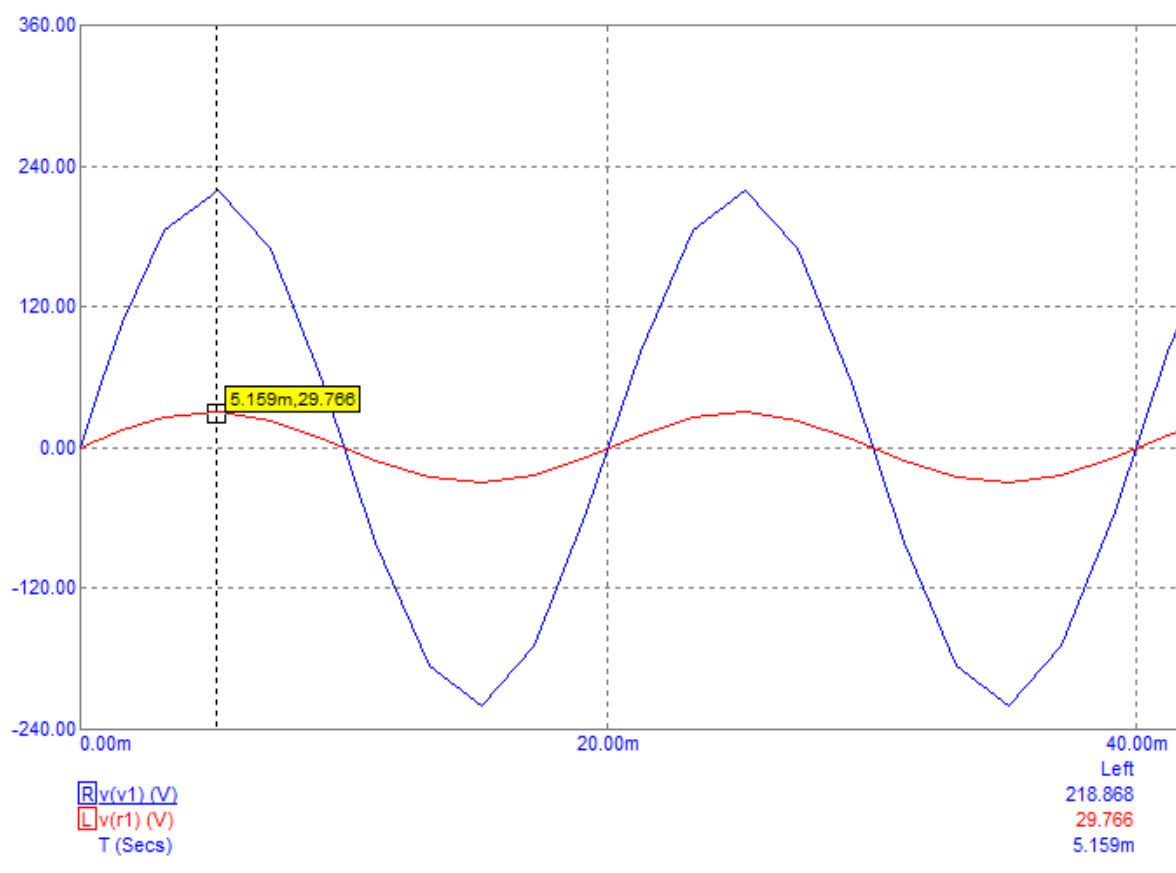


Рисунок 12 – Осциллограмма работы первичного узла

Как видно из графика, представленного выше, напряжение вторичной обмотки трансформатора ровняется 29 В, что совпадает с ранее рассчитанным значением выходного напряжения. После АС-АС преобразования и получения рабочего напряжения нам необходимо произвести АС-DC преобразование для последующей работы с другими частями схемы, в том числе и низковольтной электроникой. В качестве выпрямителя решено было использовать диодный мост, схема которого представлена на рисунке 13.

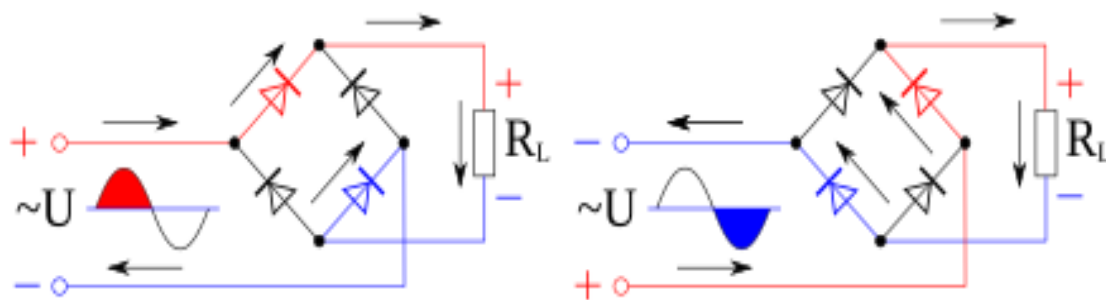


Рисунок 13 – Полный диодный мост

Для создания узла выпрямителя в моделируемую ранее схему был добавлен диодный мост, как показано на рисунке 14.

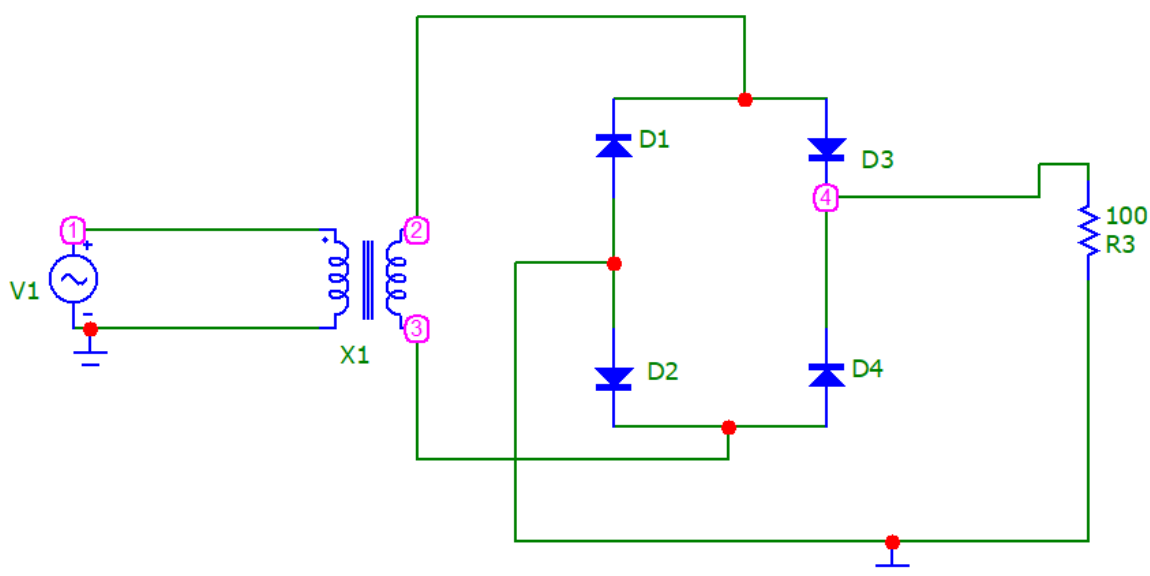


Рисунок 14 – Исследуемая схема после введения выпрямителя

Данная схема состоит из выпрямителя, основанного на диодах VD1-VD4 и резистора R3, имитирующего нагрузку. На рисунке 15 представлена осциллограмма работы полученного устройства.

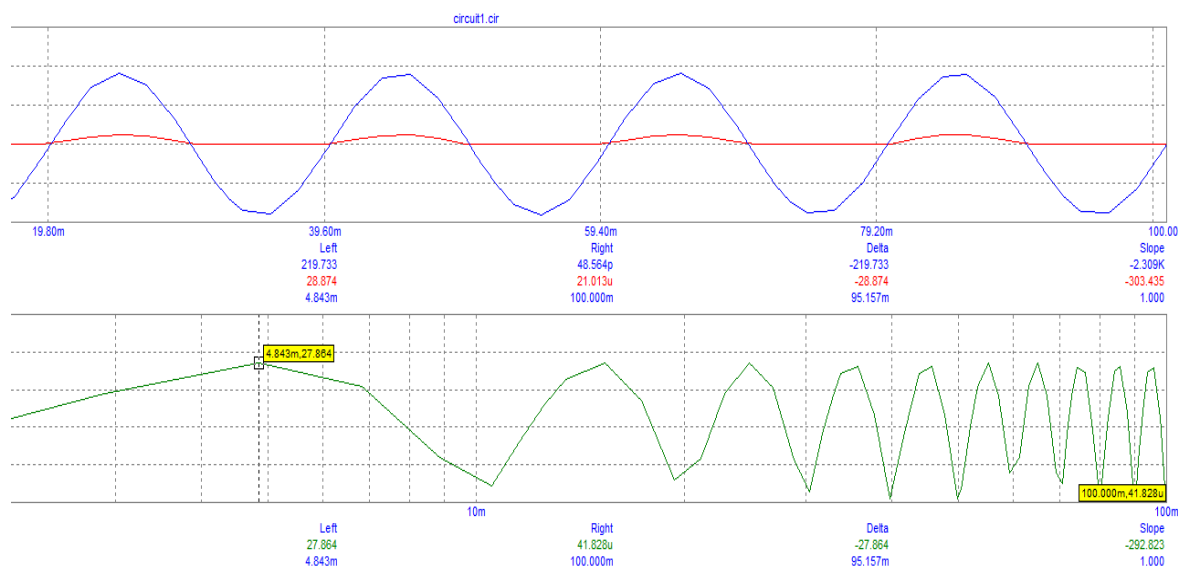


Рисунок 15 – Осциллограмма работы устройства

Из осциллограммы, представленной выше, видно наличие переходного процесса и помех в выходном сигнале системы, занимающие порядка 10 мс, а также выраженные пульсации выпрямленного напряжения от 0 до 30 В DC, что является неудовлетворительным и недопустимым результатом. Стоит отметить, что выпрямленное напряжение с диодного моста может и будет колебаться в каком-либо диапазоне, а колебания выходного напряжения в диапазоне от 0 до 30 В могут быть получены ввиду неправильного подбора библиотечных компонентов или особенности работы системы моделирования.

Для уменьшения и сглаживания выходных пульсаций напряжения после диодного моста в схему был добавлен фильтрующий конденсатор, как показано на рисунке 16.

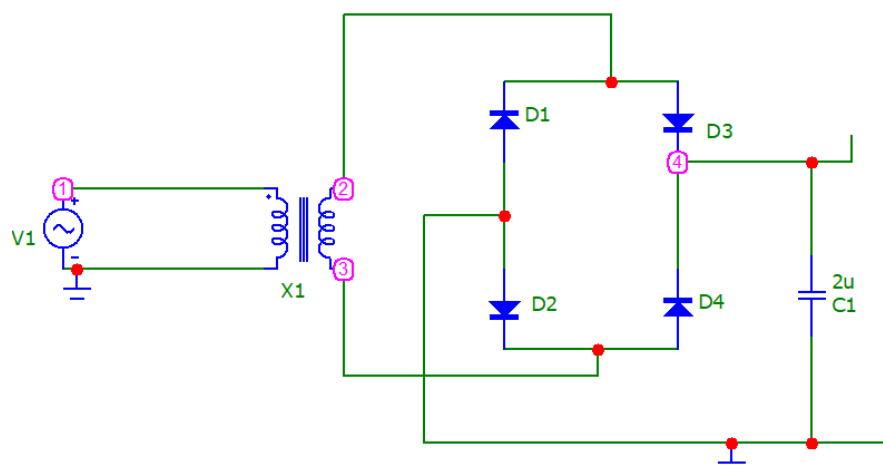


Рисунок 16 – AC-DC преобразователь с фильтром

Емкость фильтрующего конденсатора будет влиять на степень сглаживания выходного напряжения и длительность переходных процессов, подбирая его, к примеру, опытным путем можно добиться идеальных параметров выходного напряжения. На рисунке 17 представлена осциллограмма работы устройства с использованием конденсатора емкостью 100 мкФ.

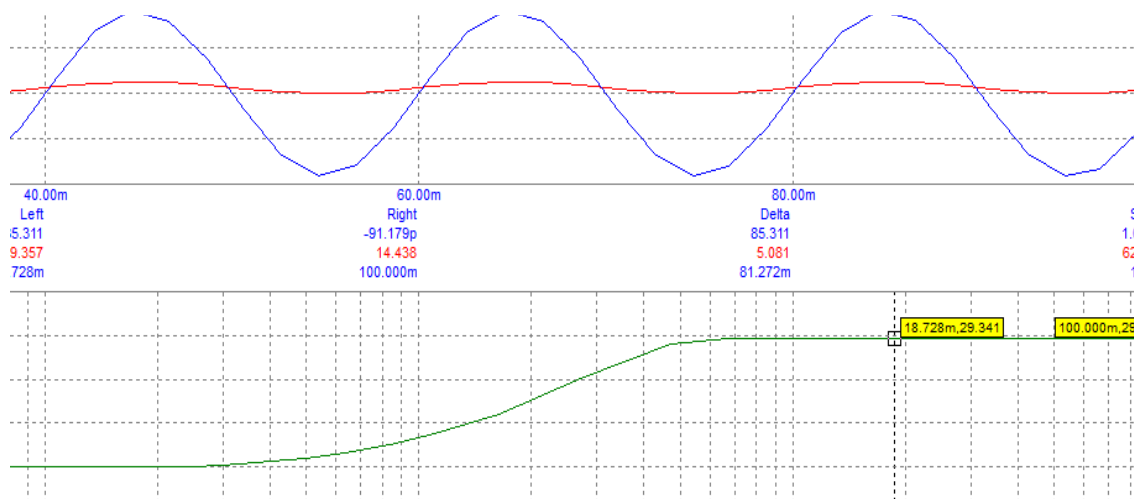


Рисунок 17 – Работа преобразователя с конденсатором 100 мкФ

Как видно из рисунка выше, после окончания переходных процессов постоянное напряжение не имеет колебаний и составляет необходимые нам 30 В. Для окончательной стабилизации, нормализации, регулирования и преобразования полученного напряжения в необходимые нам 5. Для решения данной задачи может быть использован стабилизатор напряжения, к примеру, L7805CV, представленный на рисунке 18.



Рисунок 18 – Стабилизатор напряжения L7805CV

На рисунке 19 представлены технические характеристики данного стабилизатора напряжения L7805CV, заявляемые заводом изготовителем.

#### Технические параметры

|                                                 |               |
|-------------------------------------------------|---------------|
| Полярность                                      | Положительная |
| Тип выхода                                      | Фиксированный |
| Количество выходов                              | 1             |
| Выходное напряжение, В                          | 5             |
| Максимальный ток нагрузки, А                    | 1.5           |
| Падение напряжения при I <sub>вых</sub> , В (А) | 2(1)          |
| Максимальное входное напряжение, В              | 35            |
| Рабочая температура, °С                         | 0...+125      |
| Корпус                                          | to-220        |
| Вес, г                                          | 2.5           |

Рисунок 19 – Технические характеристики L7805CV

Как видно из рабочих характеристик устройства, представленных выше, стабилизатор напряжения L7805CV поддерживает входное напряжение до 35 В, при 30, полученных нами в ходе моделирования предыдущих частей преобразователя напряжения и выдает на выходе напряжение равное 5 В, что полностью удовлетворяет нашим требованиям.

Помимо того, для реализации данной задачи можно использовать и готовую плату DC-DC преобразователя на базе LM2596, представленную на рисунке 20.



#### Технические параметры

|                              |                       |
|------------------------------|-----------------------|
| Тип компонента               | dc-dc преобразователь |
| Напряжение входное, В        | 4...40                |
| Напряжение выходное, В       | 1.5...35              |
| Максимальный выходной ток, А | 3                     |
| Вес, г                       | 11                    |

Рисунок 20 – DC-DC преобразователь на базе LM2596

Данный преобразователь напряжения способен регулировать выходное напряжение в диапазоне от 1.5 до 35 В и также может быть использован в качестве DC-DC преобразователя в рамках нашего проекта.

## 2.2 Аппаратное обеспечение системы управления

При выборе устройства управления, как одной из основных составляющих единиц аппаратной части технического решения предпочтение было отдано микроконтроллеру AVR архитектуры.

«В процессе подбора микроконтроллера учитывалась степень доступности и интегрирования среды программирования, производительность и функциональность микроконтроллера, простота закупки, достаточное количество технической документации, а также простота разработки прикладного программного обеспечения и удобство дальнейшей отладки и эксплуатации» [5].

«AVR принято называть семейство восьмибитных микроконтроллеров от компании Atmel. Микроконтроллеры данного семейства активно применяются в разнообразных встраиваемых системах и часто встречаются в реализации отладочных плат для программирования и самостоятельного конструирования электронных устройств от популярных фирм, например Arduino» [13].

Микроконтроллеры AVR имеют гарвардскую архитектуру, что подразумевает хранение пользовательской программы и данных в разных адресных пространствах и наличие систем команд, близких к идеологии RISC. Процессор AVR имеет 32 восьмибитных регистра общего назначения, объединенных в файл регистров. Все множество функциональных команд AVR микроконтроллеров можно разделить на следующие группы:

- логические операции;
- арифметические операции;
- сдвиговые операции;
- битовые операции;
- операции пересылки данных;
- команды передачи управления;
- команды системы управления.

На рисунке 21 представлено схематичная архитектура микроконтроллера AVR семейства.

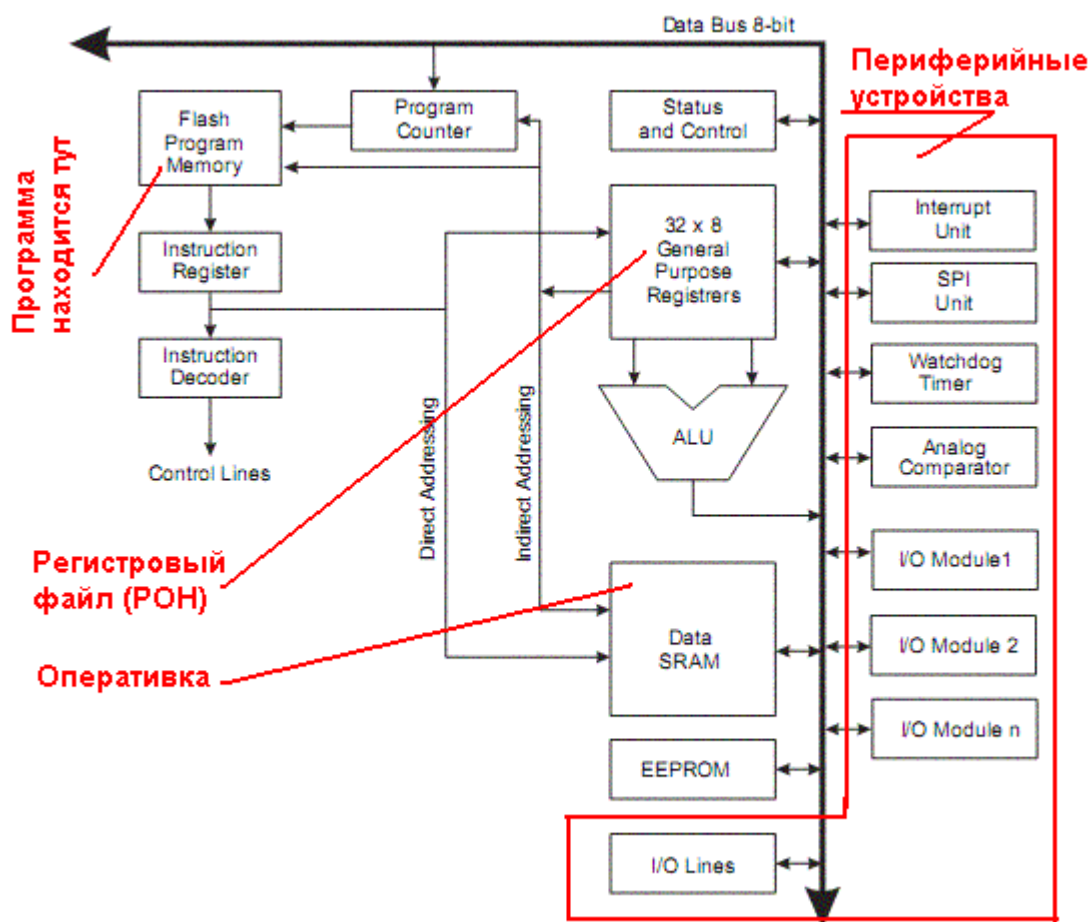


Рисунок 21 – Архитектура AVR микроконтроллера

Вычислительное ядро у подавляющего большинства AVR микроконтроллеров идентично, что обеспечивает единство архитектуры и легкость переноса программного обеспечения. В состав вычислительного ядра микроконтроллера входит память программы, арифметико-логическое устройство, блок управления и программный счетчик. На рисунке 22 представлена более подробная архитектурная схема AVR микроконтроллера, с учетом служебных, транспортных и управляющих блоков.

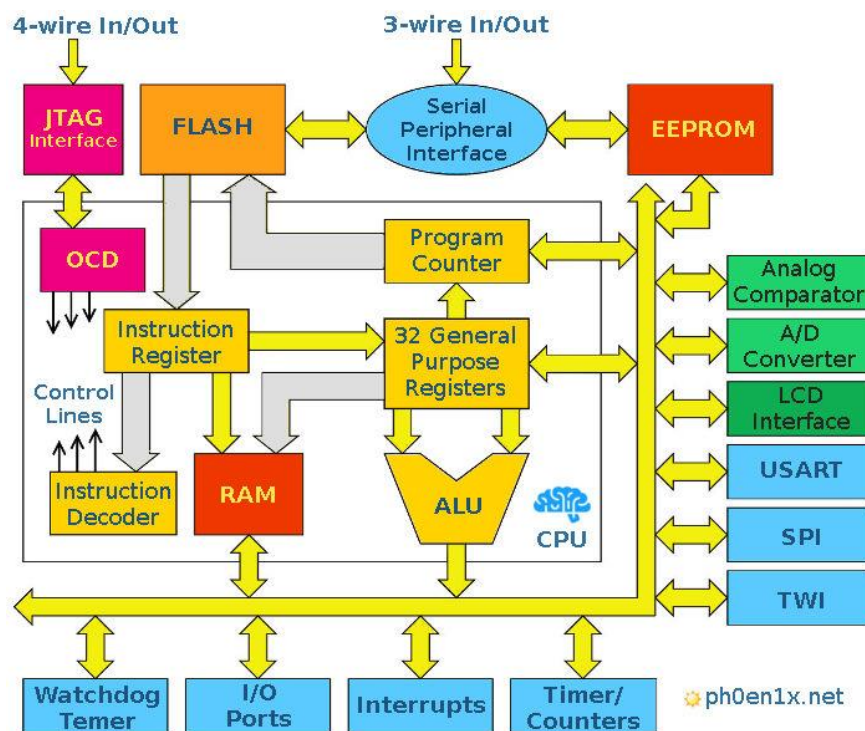


Рисунок 22 – Структурная схема AVR микроконтроллера

На рисунке 23 представлено описание основных составных элементов AVR микроконтроллера.

- **JTAG Interface** (Joint Test Action Group Interface) - интерфейс внутрисхемной отладки (4 провода);
- **FLASH** - перепрограммируемая память для сохранения программы;
- **Serial Peripheral Interface, SPI** - последовательный периферийный интерфейс (3 провода);
- **EEPROM** (Electrically Erasable Programmable Read-Only Memory) - перепрограммируемое ПЗУ, энергонезависимая память;
- **CPU** (ЦПУ) - центральный процессор управления, сердце микроконтроллера, 8-битное микропроцессорное ядро;
- **ALU** (АЛУ) - арифметико-логическое устройство, основа блока CPU;
- **RAM** (Random Access Memory) - оперативная память процессора;
- **Program Counter** - счетчик команд;
- **32 General Purpose Registers** - 32 регистра общего назначения;
- **Instruction Register** - регистр команд, инструкций;
- **Instruction Decoder** - декодер команд;
- **OCD** (On-Chip Debugger) - блок внутренней отладки;
- **Analog Comparator** - аналоговый компаратор, блок сравнения аналоговых сигналов;
- **A/D Converter** (Analog/Digital converter) - аналогово-цифровой преобразователь;
- **LCD Interface** (Liquid-Crystal Display Interface) - интерфейс для подключения жидко-кристаллического дисплея, индикатора;
- **USART** (Universal Asynchronous Receiver-Transmitter), UART - универсальный асинхронный приемопередатчик;
- **TWI** (Two-Wire serial Interface) - последовательный интерфейс с двухпроводным подключением;
- **Watchdog Timer** - сторожевой или контрольный таймер;
- **I/O Ports** - порты ввода/вывода;
- **Interrupts** - блок управления и реакции на прерывания;
- **Timers/Counters** - модули таймеров и счетчиков.

Рисунок 23 – Описание составных элементов AVR микроконтроллера

Говоря о AVR микроконтроллерах невозможно не выделить отладочные платы фирмы Arduino, спроектированные на их основе. «Отладочные платы Arduino активно используются при автоматизации несложных производственных операций, проектировании, прототипировании и сборке разнообразных электронных устройств. В текущий момент на рынке представлен большой выбор отладочных плат Arduino с разными техническими характеристиками, форм-фактором, количеством портов и функциональных возможностей, производительностью и микроконтроллером, установленным на борту» [4]. На рисунке 24 представлено разнообразие существующих плат Arduino.



Рисунок 24 – Разнообразие отладочных плат Arduino

Анализируя доступные отладочные платы, было принято решения использовать Arduino UNO для реализации нашего проекта в рамках выполнения выпускной квалификационной работы, как полностью удовлетворяющую нашим требованиям как программно, так и аппаратно [12]. На рисунке 25 представлена плата Arduino UNO с обозначением выводов.

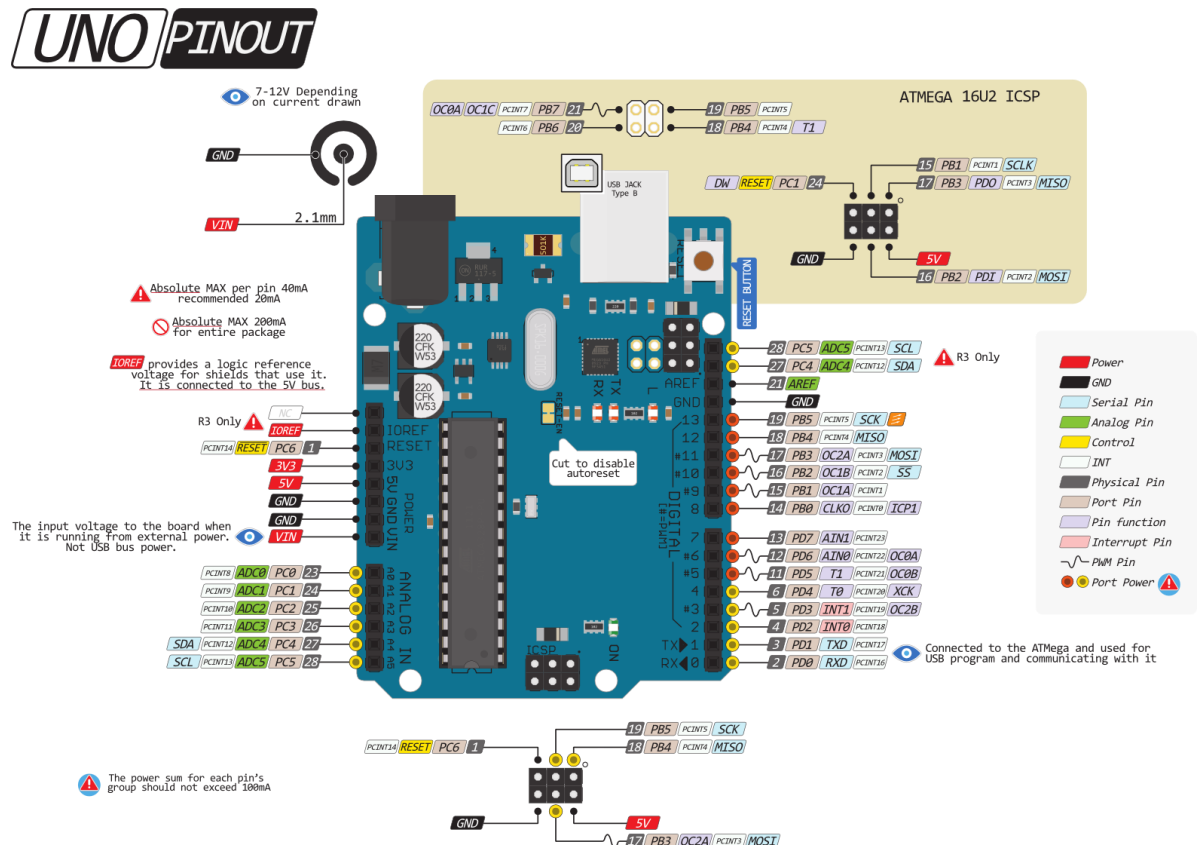
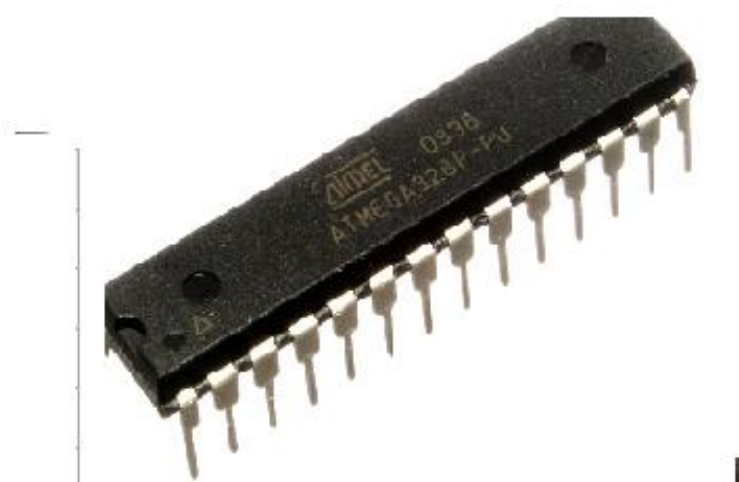


Рисунок 25 – Назначение выводов Arduino UNO

«Отладочная плата Arduino UNO сконструирована на базе микроконтроллера ATmega328. Данный чип представляет собой 8-и битный микроконтроллер AVR архитектуры и поддерживает максимальную рабочую тактовую частоту 20 МГц» [12]. На рисунке 26 представлен микроконтроллер ATmega328 с заявленными характеристиками.



| Parameter                         | Value                            |
|-----------------------------------|----------------------------------|
| CPU type                          | 8-bit AVR                        |
| Maximum CPU speed                 | 20 MHz                           |
| Performance                       | 20 MIPS at 20 MHz <sup>[2]</sup> |
| Flash memory                      | 32 KB                            |
| SRAM                              | 2 KB                             |
| EEPROM                            | 1 KB                             |
| Package pin count                 | 28 or 32                         |
| Capacitive touch sensing channels | 16                               |
| Maximum I/O pins                  | 23                               |

Рисунок 26 – Микроконтроллер ATmega328

Сама отладочная плата Arduino UNO на базе данного микроконтроллера обладает характеристиками, представленными на рисунке 27.

|                                    |                                                                        |
|------------------------------------|------------------------------------------------------------------------|
| Микроконтроллер                    | ATmega328                                                              |
| Рабочее напряжение                 | 5 В                                                                    |
| Входное напряжение (рекомендуемое) | 7-12 В                                                                 |
| Входное напряжение (предельное)    | 6-20 В                                                                 |
| Цифровые Входы/Выходы              | 14 (6 из которых могут использоваться как выходы <a href="#">ШИМ</a> ) |
| Аналоговые входы                   | 6                                                                      |
| Постоянный ток через вход/выход    | 40 мА                                                                  |
| Постоянный ток для вывода 3.3 В    | 50 мА                                                                  |
| Флеш-память                        | 32 Кб (ATmega328) из которых 0.5 Кб используются для загрузчика        |
| ОЗУ                                | 2 Кб (ATmega328)                                                       |
| EEPROM                             | 1 Кб (ATmega328)                                                       |
| Тактовая частота                   | 16 МГц                                                                 |

Рисунок 27 – Технические характеристики Arduino UNO

Из рисунка выше видно, что с выбранной платой можно связываться по UART и SPI интерфейсам, а напряжение питания подается на отладочную плату при помощи USB 5 В DC или при помощи специализированных входов питания, вплоть до 20 В DC благодаря встроенному стабилизатору напряжения на 5 В, однако при подключении внешнего напряжения все таки рекомендуется использовать не более 12 В DC, как показано на рисунке 28.

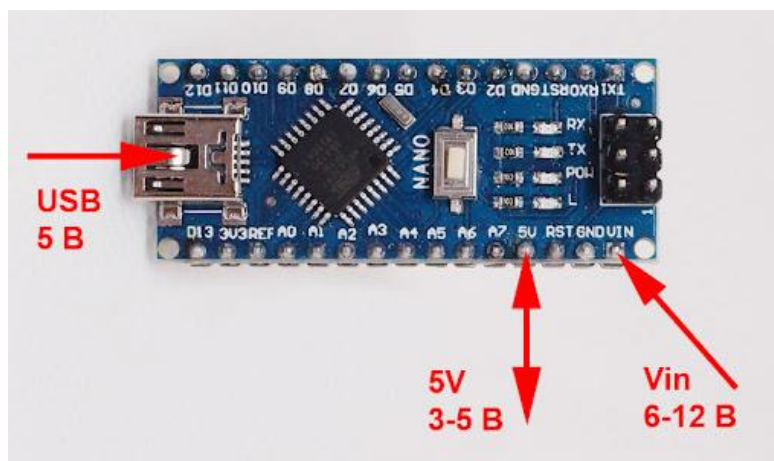


Рисунок 28 – Вариативность подключения питания к платам Arduino

### 2.3 Системы контроля и поддержания микроклимата

Для определения уровня влажности субстрата и регулирования процесса автоматического полива в состав разрабатываемой системы добавлен емкостной датчик влажности почвы, совместимый с Arduino и представленный на рисунке 29.

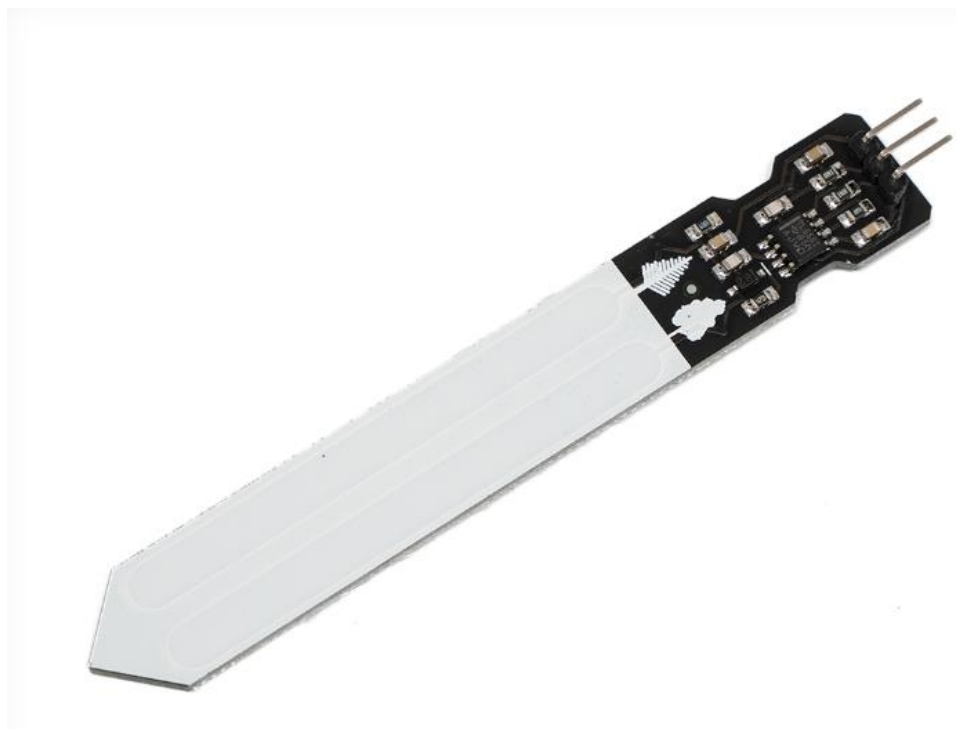


Рисунок 29 – Емкостной датчик влажности почвы

В отличие от датчиков резистивного типа, данный датчик не подвержен коррозии благодаря использованию технологии шунтирования высокой частоты, что позволяет электродам не соприкасаться с влажной средой, это является для нас дополнительным преимуществом. Датчик влажности почвы является аналоговым, а напряжение на его выходе обратно пропорционально влажности измеряемого субстрата или почвы [13]. На рисунке 30 представлены основные технические характеристики данного датчика.

- Напряжение питания  $V_{cc}$ : 5 В или 3,3 В
- Напряжение на выходе датчика при  $V_{cc}$  5В: ~3 ... 1,75 В; при  $V_{cc}$  3,3В: ~2 ... 1 В
- Выход датчика инверсный
- Максимальный потребляемый ток: < 5,3 мА, при  $V_{cc}$  = 5 В и датчик погружён в грунтовую воду.
- Глубина погружения в почву: 65 мм
- Рабочая температура: 0 ... 85 °С
- Габариты: 105х16х7 мм (с учётом колодки выводов)
- Вес: 5 г

Рисунок 30 – Основные технические характеристики датчика  
влажности почвы

На рисунке 31 представлены способы подключения выбранного датчика к отладочной плате, аналогичной Arduino UNO.

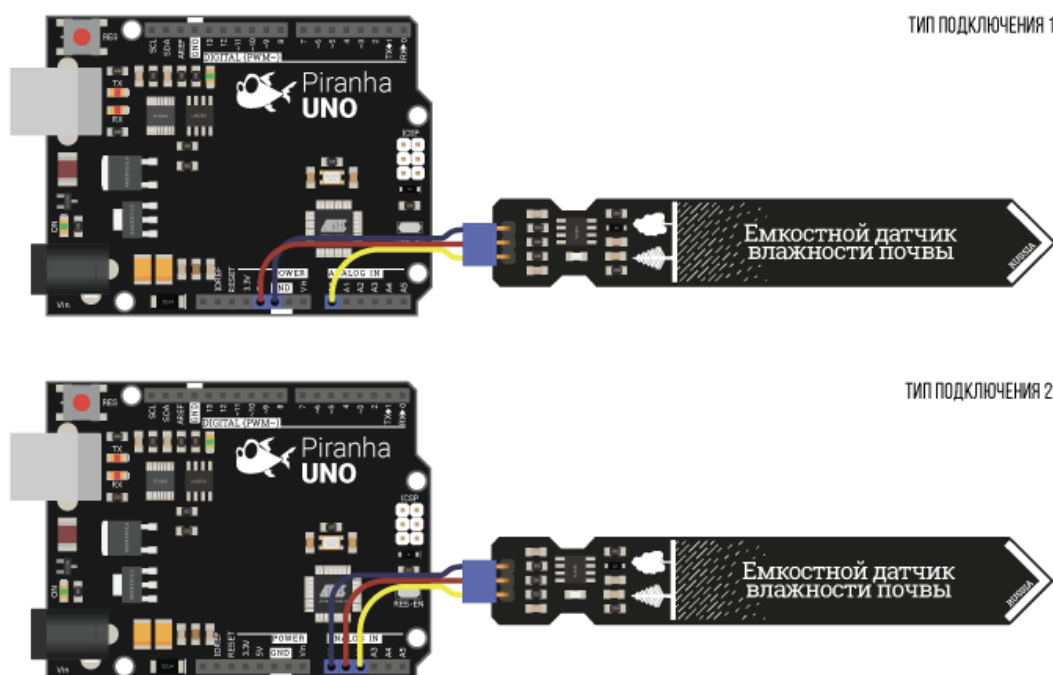


Рисунок 31 – Способы подключения датчика

В зависимости от влажности измеряемой среды, показания датчика будут варьироваться в диапазоне от 1.75 до 3 В, при условии питания от источника напряжения номиналом 5 В, при использовании питания 3.3 В, показания датчика будут варьироваться в диапазоне от 1 до 2 В, это стоит учитывать при использовании функции `analogRead()` и последующей обработке полученных значений. На рисунке 32 представлен график зависимости выходного напряжения датчика от влажности измеряемой среды.

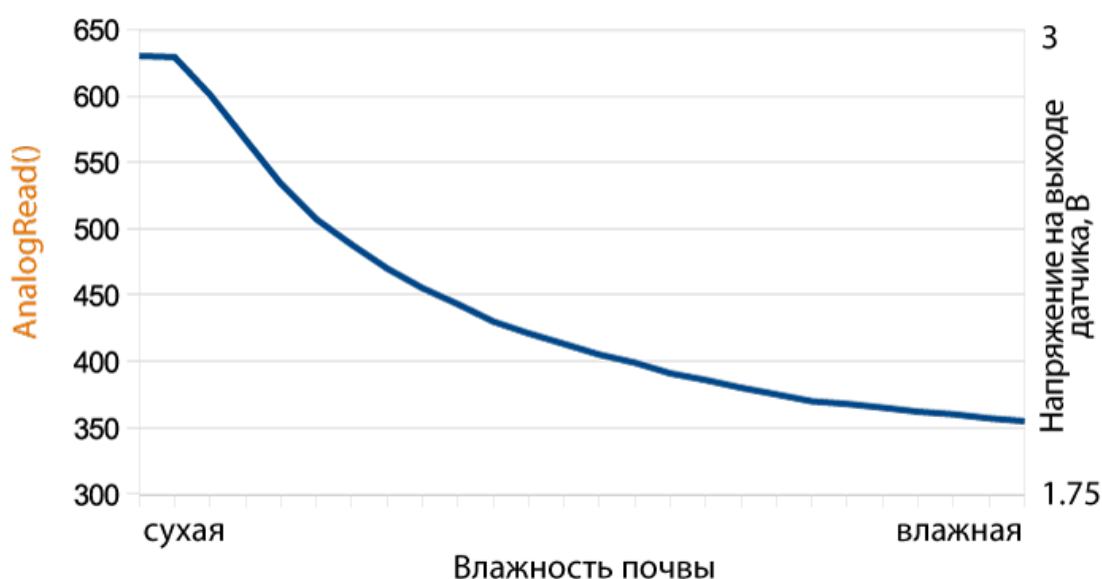


Рисунок 32 – График зависимости выходного напряжения датчика от влажности почвы при напряжении питания 5 В

Для определения температуры и влажности окружающей газовой смеси будет использоваться совместимый с системой Arduino датчик температуры и влажности DHT-11 или его аналог DHT-22, отличающийся более стабильной и точной картиной показаний. На рисунке 33 представлены климатические датчики DHT-11 и DHT-22.



Рисунок 33 – Климатические датчики DHT-11 и DHT-22

Конструктивно данный датчик включает в себя два основных элемента, это термистор и емкостной датчик влажности.

Термистор представляет из себя резистор, сопротивление которого изменяется при изменении температуры, если температура уменьшается, то возрастает сопротивление термистора, а если увеличивается, то сопротивление термистора уменьшается, что влияет на выходной сигнал датчика, который и преобразуется системой управления в конечную температуру.

Емкостной датчик представляет из себя переменный конденсатор, содержащий токопроводящие обкладки из медной фольги, расположенные на листах текстолита, а сама конструкция помещена в герметичный чехол, поверх которого установлен влагопоглощающий слой. Когда частички влаги из окружающей воздушной смеси попадают на данный слой происходит изменение его диэлектрической проницаемости, что влечет за собой изменение емкости конденсатора и соответственно изменение параметров выходного сигнала.

На рисунке 34 и 35 представлены сравнительные характеристики климатических датчиков DHT-11 и DHT-22 и пример подключения датчика к отладочной плате соответственно.

- Потребляемый ток – 2,5 мА (максимальное значение при преобразовании данных);
- Измеряет влажность в диапазоне от 20% до 80%. Погрешность может составлять до 5%;
- Применяется при измерении температуры в интервале от 0 до 50 градусов (точность – 2%)
- Габаритные размеры: 15,5 мм длина; 12 мм ширина; 5,5 мм высота;
- Питание – от 3 до 5 Вольт;
- Одно измерение в единицу времени (секунду). То есть, частота составляет 1 Гц;
- 4 коннектора. Между соседними расстояние в 0,1 ".

**DHT-11**

- Питание – от 3 до 5 Вольт;
- Максимальный ток при преобразовании – 2,5 мА;
- Способен измерять влажность в интервале от 0% до 100%. Точность измерений колеблется от 2% до 5%;
- Минимальная измеряемая температура – минус 40, максимальная – 125 градусов по Цельсию (точность измерений – 0,5);
- Устройство способно совершать одно измерение за 2 секунд. Частота – до 0,5 Гц;
- Габаритные размеры: 15,1 мм длина; 25 мм ширина; 5,5 мм высота;
- Присутствует 4 коннектора. Расстояние между соседними – 0,1 ";

**DHT-22**

Рисунок 34 – Сравнительные характеристики DHT-11 и DHT-22

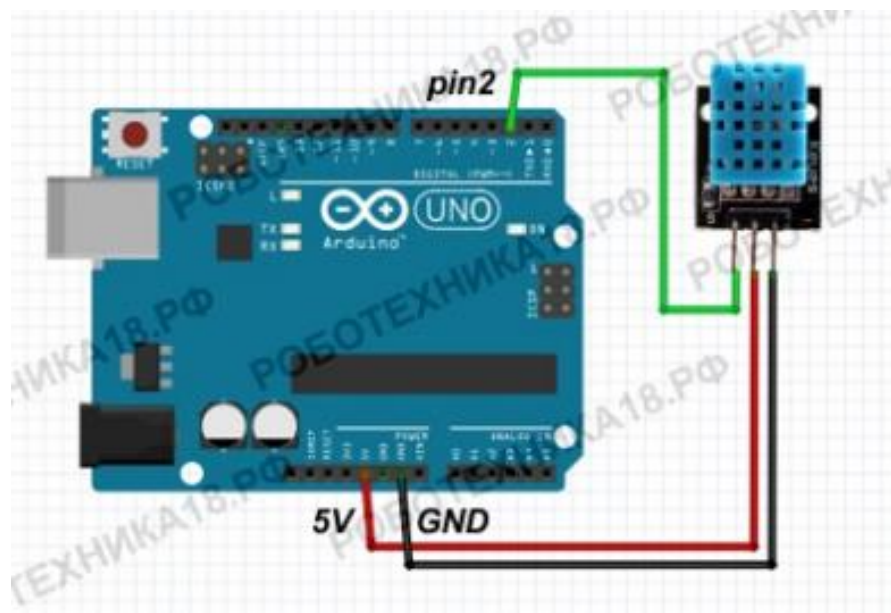


Рисунок 35 – Пример подключения датчика к отладочной плате

Для осуществления процесса аэрации питательного водного раствора решено было использовать воздушный компрессор, который можно было бы совместить с аппаратной платформой Arduino, к примеру, Air pump UNO R3, представленный на рисунке 36.



Рисунок 36 – Компрессор UNO R3

На выходной канал воздушного компрессора будет установлен обратный клапан, препятствующий попаданий воды в воздушную систему стеллажа для гидропоники. В свою очередь воздух будет забираться и подаваться в емкость с питательным водным раствором через силиконовые трубки, на конце которой, непосредственно внутри бака, будет установлен аэратор. На рисунке 37 представлена функциональная структура системы аэрации автоматизированного стеллажа для гидропоники.



Рисунок 37 – Система аэрации

Подключение компрессора к плате управления осуществляется согласно назначению выводов, представленному на рисунке 38.

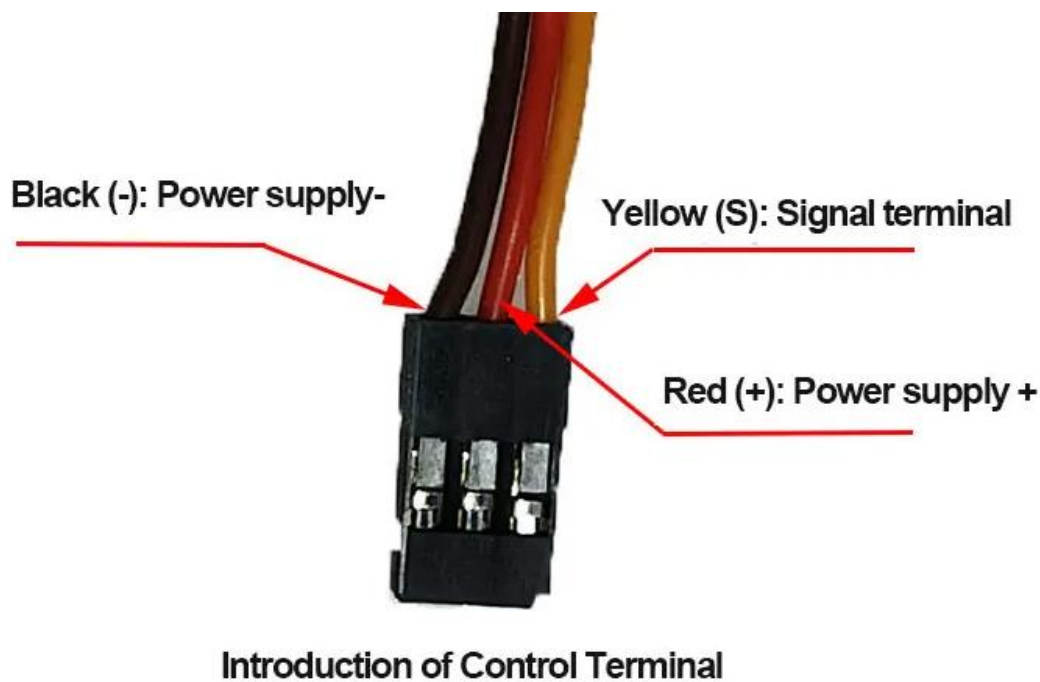


Рисунок 38 – Клемма подключения компрессора

Опционально в систему аэрации питательного водного раствора может быть включен датчик уровня кислорода в водном растворе, что обеспечит возможность мониторинга и контроля процесса аэрации, однако значительно увеличит конечную стоимость разрабатываемого нами устройства, ввиду сложности конструкции и относительно высокой стоимости данного датчика [14]. На рисунке 39 представлен датчик уровня кислорода DFROBOT DO O2 и способ его подключения к отладочной плате.

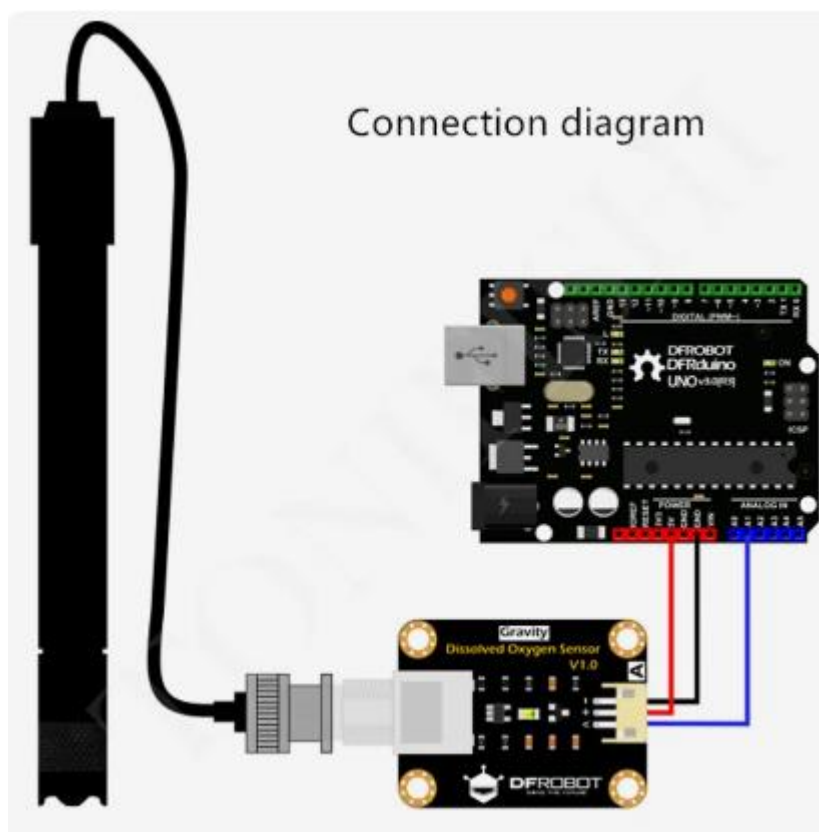


Рисунок 39 – Подключение датчика DFROBOT DO O2

Для осуществления функции искусственного освещения могут быть использованы специализированные лампы или обычные лампы накаливания. Вне зависимости от типа используемых ламп, техническое решение по управлению освещением решено было реализовать, руководствуясь правилами единообразия, под лампы напряжением 220 В AC, как наиболее распространенные.

Таким образом, включение ламп освещения будет реализовано путем коммутации через реле, рассчитанное на работу с выходным напряжением 220 В AC и напряжением управления 5 В DC. На рисунке 40 представлено реле Trema v2.0 и его основные характеристики, полностью удовлетворяющее нашим требованиям.



- Напряжение питания модуля: 5 В постоянного тока.
- Ток потребляемый модулем: до 80 мА, при наличии «1» на входе «S».
- Ток потребляемый модулем: до 1 мкА, при наличии «0» на входе «S».
- Ток поротекаемый через вход «S»: 4 мА, при наличии «1» на входе «S».
- Коммутируемые модулем выходные цепи:
  - до 30 В постоянного тока 10 А
  - до 250 В переменного тока 10 А
- Сопротивление изоляции реле: выше 1 ГОм
- Время срабатывания реле, при подаче «1» на вход «S»: до 15 мс.
- Время срабатывания реле, при подаче «0» на вход «S»: до 8 мс.
- Скорость механических переключений: до 200 операций / мин.
- Материал контактов реле: AgNi, AgCdO, AgSnO<sub>2</sub>.
- Рабочая температура: -40 ... +85 °C

Рисунок 40 – Реле Trema v2.0

На рисунке 41 представлен пример подключения реле к отладочной плате.

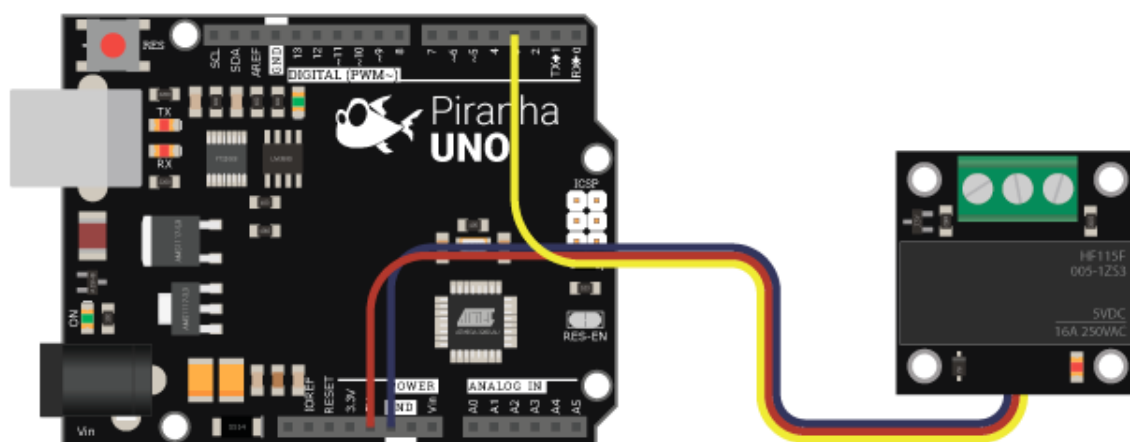


Рисунок 41 – Пример подключения реле к отладочной плате

Для осуществления функции автоматического полива будет использоваться мембранный насос 6-12 В, представленный на рисунке 42.

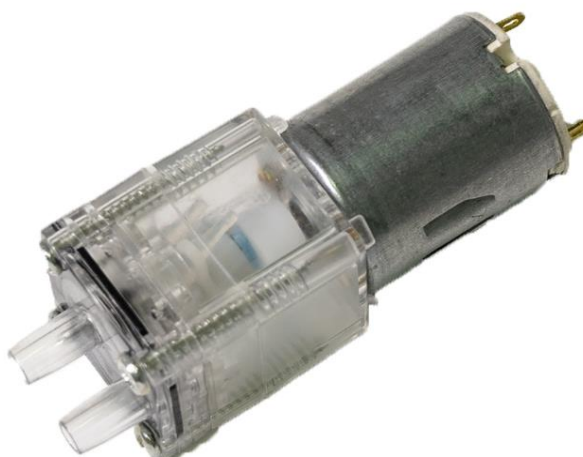


Рисунок 42 – Мембранный насос типа 280

Для обеспечения принудительной циркуляции воздуха в пределах стеллажа для гидропоники будут использованы вентиляторы обдува 5-12 В, к примеру вентилятор TWO TREES, представленный на рисунке 42.



Рисунок 42 – Вентилятор охлаждения TWO TREES

Таким образом, при выбранной компонентной базе, разрабатываемая автоматизированная система сможет выполнять функции контроля микроклимата и поддержания необходимых условий для выращивания полезных культур в гидропонике.

## 2.4 Разработка системы НМИ

Для визуализации необходимых для контроля и мониторинга данных, а также возможности взаимодействия пользователя с системой и ее настройки будет использован совместимый с аппаратной платформой Arduino дисплей, представленный на рисунке 43 [15].

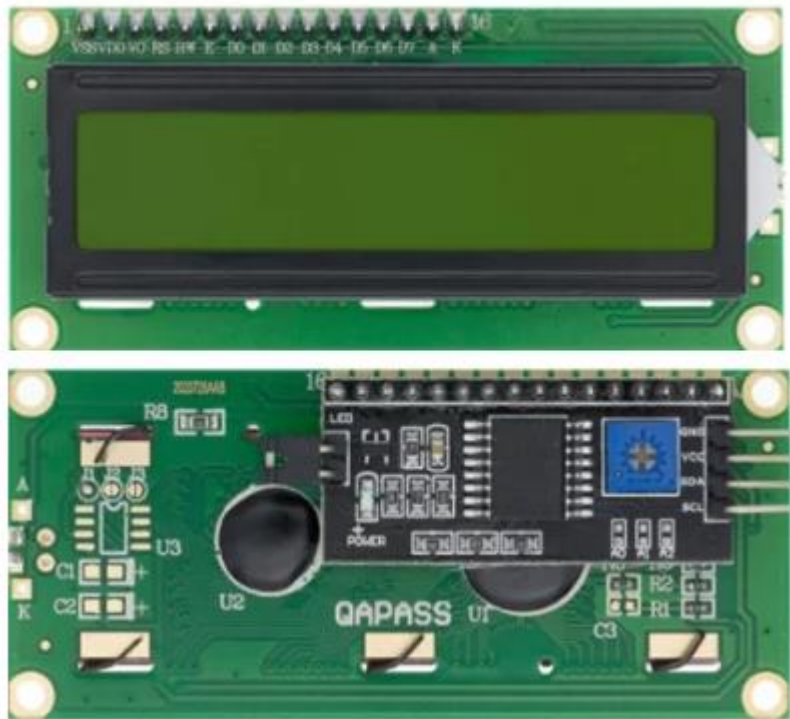


Рисунок 43 – LCD дисплей 1602

Существует 2 способа подключения данного дисплея к аппаратной платформа Arduino, с использованием интерфейса I2C и специализированных программных библиотек или при помощи 12-и проводного подключения. Примеры подключения дисплея LCD 1602 к отладочной плате представлены на рисунке 44.

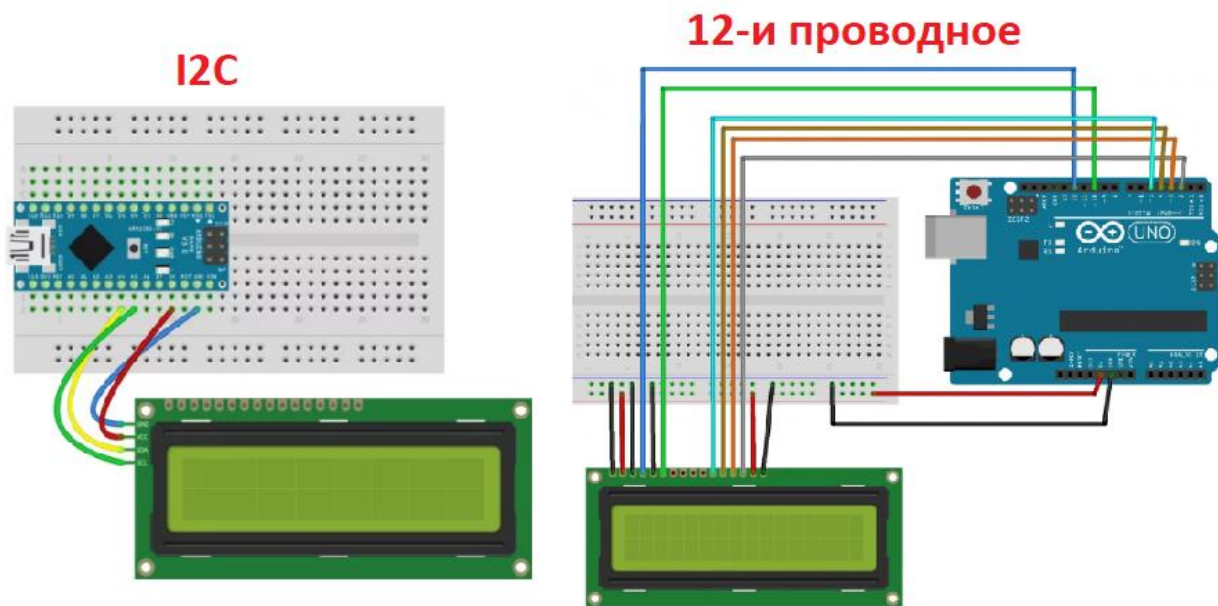


Рисунок 44 – Подключение дисплея

Ниже представлен пример программного кода для взаимодействия с данным дисплеем после подключения необходимых библиотек:

```
print(data); // вывод данных
setCursor(x, y); // выставить курсор
clear(); // команда отчистки дисплея
home(); // курсор в позицию 0-0
noDisplay(); // отключить отображение
display(); // включить отображение
blink(); // мигание курсора в текущей позиции
noBlink(); // отключение функции мигания
cursor(); // отображение курсора
noCursor(); // команда, скрывающая курсор
scrollDisplayLeft(); // пролистать экран влево на 1 столбец
```

`scrollDisplayRight();` // пролистать экран вправо на 1 столбец

`backlight();` // включение подсветки

`noBacklight();` // выключить подсветки

`createChar(uint8_t, uint8_t[]);` // команда создания специального символа

`createChar(uint8_t location, const char *charmap);` // добавить символ

На рисунке 45 представлена кодировка специализированных символов, которые потребуются нам в ходе разработки проекта.

| Lower 4 Bits | Upper 4 Bits | 0000 | 0001 | 0010 | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1001 | 1010 | 1011 | 1100 | 1101 | 1110 | 1111 |
|--------------|--------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| xxxx0000     | CS RAM (1)   |      |      | 0    | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    | 10   | 11   | 12   | 13   |
| xxxx0001     | (2)          |      |      | !    | 1    | A    | Q    | a    | q    |      |      |      |      | 。    | ア    | チ    | 4    |
| xxxx0010     | (3)          |      |      | "    | 2    | B    | R    | b    | r    |      |      |      |      | 「    | イ    | ツ    | ×    |
| xxxx0011     | (4)          |      |      | #    | 3    | C    | S    | c    | s    |      |      |      |      | 」    | ウ    | テ    | ε    |
| xxxx0100     | (5)          |      |      | \$   | 4    | D    | T    | d    | t    |      |      |      |      | 、    | エ    | ト    | μ    |
| xxxx0101     | (6)          |      |      | %    | 5    | E    | U    | e    | u    |      |      |      |      | ・    | オ    | ナ    | 1    |
| xxxx0110     | (7)          |      |      | &    | 6    | F    | V    | f    | v    |      |      |      |      | ヲ    | カ    | ニ    | ヨ    |
| xxxx0111     | (8)          |      |      | '    | 7    | G    | W    | g    | w    |      |      |      |      | ア    | キ    | ヌ    | ラ    |
| xxxx1000     | (1)          |      |      | (    | 8    | H    | X    | h    | x    |      |      |      |      | イ    | ク    | ネ    | リ    |
| xxxx1001     | (2)          |      |      | )    | 9    | I    | Y    | i    | y    |      |      |      |      | ウ    | ケ    | ル    | ニ    |
| xxxx1010     | (3)          |      |      | *    | :    | J    | Z    | j    | z    |      |      |      |      | エ    | コ    | ハ    | レ    |
| xxxx1011     | (4)          |      |      | +    | ;    | K    | L    | k    | l    |      |      |      |      | オ    | サ    | ヒ    | ロ    |
| xxxx1100     | (5)          |      |      | ,    | <    | L    | ¥    | 1    | l    |      |      |      |      | ハ    | シ    | フ    | ワ    |
| xxxx1101     | (6)          |      |      | -    | =    | M    | J    | m    | }    |      |      |      |      | ユ    | ズ    | ヘ    | ン    |
| xxxx1110     | (7)          |      |      | .    | >    | N    | ^    | n    | ÷    |      |      |      |      | ヨ    | セ    | ホ    | °    |
| xxxx1111     | (8)          |      |      | /    | ?    | O    | _    | o    | ←    |      |      |      |      | ッ    | ソ    | マ    | °    |

Рисунок 45 – Кодировка специальных символов

Для осуществления управляющих воздействий на систему управления, настройки и редактирования параметров могут быть использованы дискретные кнопки или готовые сборки в виде пультов и клавиатуры, как показано на рисунке 46.

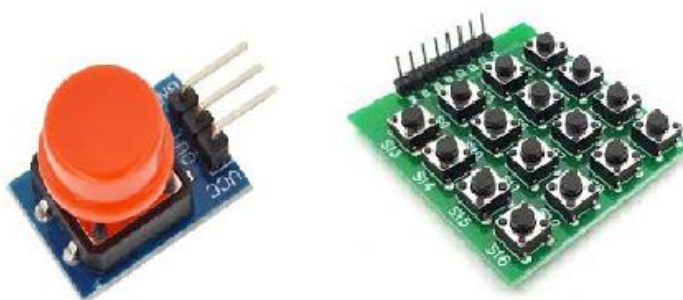


Рисунок 46 – Вариант исполнения функциональных клавиш

Подключение дискретных клавиш происходит согласно схеме, представленной на рисунке 47.

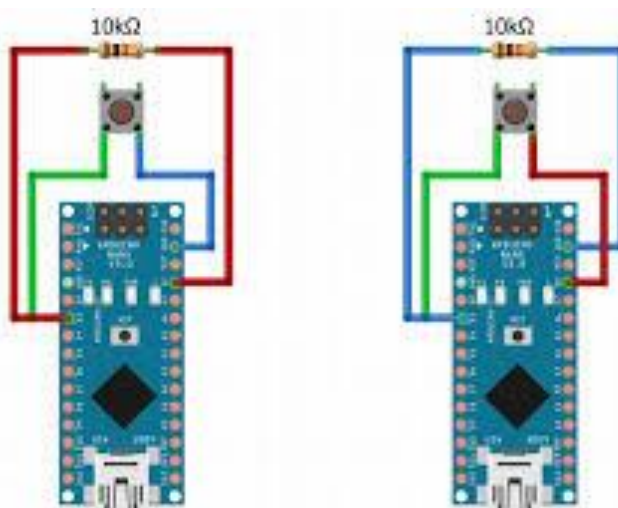


Рисунок 47 – Подключение кнопок

Пин отладочной платы, отслеживающий сигнал, должен иметь два стабильных состояния, логический ноль и логическую единицу, иными словами, подтягиваться к GND 0 В и напряжению питания + 5 В. Для этого один из выводов кнопки подключается через резистор номиналом 10 кОм к противоположному по номиналу контакту, как показано на схеме выше. Из этого следует, что если нам необходимо отслеживать низкий уровень сигнала 0 В необходимо подключить сигнальный вывод к выводу питания, а если высокий логический уровень, то к GND, что обеспечит два устойчивых логических состояния и нивелирует возможность наводок и ложных срабатываний.

Помимо схемотехнической реализации, в аппаратной платформе Arduino существуют и программные подтяжки к логическому нулю и единице, реализованные в виде функций `INTUP_PULLDOWN` и `INTUP_PULLUP` соответственно.

Также существует возможность подключения нескольких кнопок к одному аналоговому входу отладочной платы, как показано на рисунке 48.

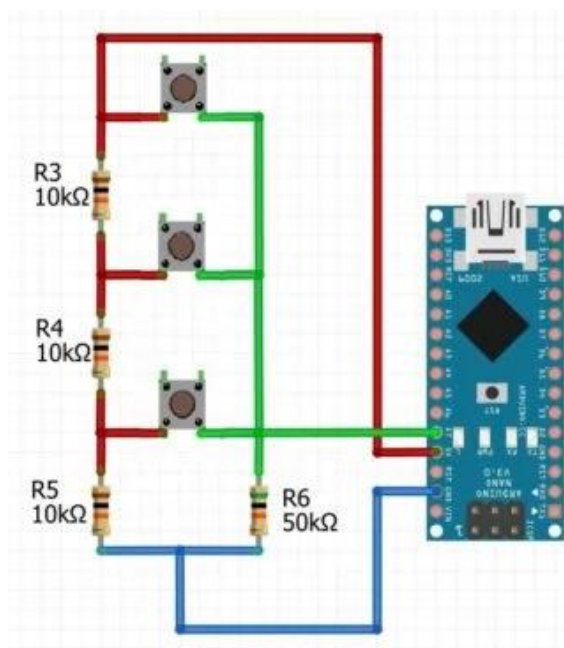


Рисунок 48 – Подключение кнопок к аналоговому входу

При такой схеме подключения используется делитель напряжения на резисторах, таким образом, в случае, когда не нажата ни одна кнопка на сигнальный вход A7 подается напряжение равное 0 В за счет подтягивающего резистора R6. Если будет нажата верхняя кнопка, то на сигнальный вход будет подано напряжение номиналом 5 В, а если средняя, то порядка 2.5 В за счет включения в схему резистора. Аналогово-цифровой преобразователь внутри микроконтроллера преобразует полученное значение напряжения в диапазон от 0 до 1023, тем самым программно мы можем различать нажатые кнопки на одном входе микроконтроллера.

При обработке сигналов с кнопок необходимо не забывать про такое пагубное влияние, как дребезг контактов, способное приводить к ложным срабатываниям и реакциям системы. Для уменьшения вероятности возникновения данного процесса необходимо использовать встроенные функции задержки при обработке сигнала, к примеру, функцию `delay()` или аппаратные прерывания, как показано на рисунке 49.

| <b>МК / номер прерывания</b>     | <b>INT 0</b> | <b>INT 1</b> | <b>INT 2</b> | <b>INT 3</b> | <b>INT 4</b> | <b>INT 5</b> |
|----------------------------------|--------------|--------------|--------------|--------------|--------------|--------------|
| ATmega 328/168 (Nano, UNO, Mini) | D2           | D3           | -            | -            | -            | -            |
| ATmega 32U4 (Leonardo, Micro)    | D3           | D2           | D0           | D1           | -            | -            |
| ATmega 2560 (Mega)               | D21          | D20          | D19          | D18          | D2           | D3           |

Рисунок 49 – Поддержка функций прерывания

Прерывания активируется за счет поступления сигнала на указанный вывод и активации специальных программных инструкций. В момент срабатывания прерывания главная программа и все сторонние настройки не выполняются, отрабатывается только программный код, указанный нами в теле прерывания.

Таким образом, разрабатываемый человеко-машинный интерфейс обеспечит возможность контроля параметров температуры, влажности воздуха и субстрата, параметров питательного раствора, а также регулировку всех необходимых программных уставок контроля, функций и действий, к примеру, включение света, полива, аэрации и обдува.

#### Выводы по разделу

В данном разделе были рассмотрены варианты выбора и установки аппаратного обеспечения системы управления. Была выбрана необходимая нам отладочная плата с микроконтроллером. Было разработано и промоделировано схемотехническое решение цепей обеспечения питания системы от сети 220 В. Был произведен подбор всей необходимой компонентной базы системы управления и систем контроля за общим микроклиматом в рабочей зоне. Был разработан человеко-машинный интерфейс, необходимый для реализации возможности контроля, мониторинга и управления системой, для которого была также подобрана вся необходимая компонентная база.

### 3 Разработка управляющей программы

Для разработки управляющей программы будет использоваться среда программирования Arduino IDE, специализированная под аппаратную платформу Arduino и схожие отладочные платы [11]. На рисунке 50 представлена рабочая среда Arduino IDE.

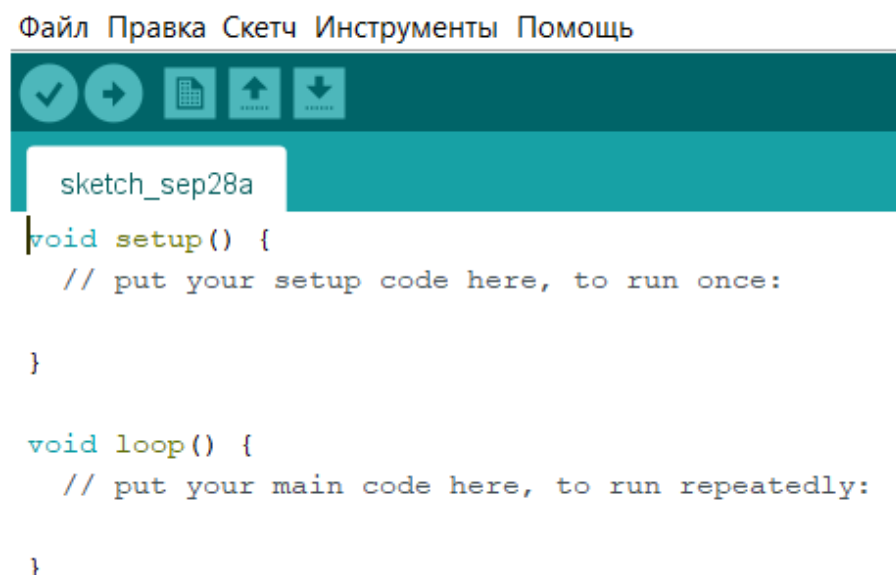


Рисунок 50 – Рабочая среда Arduino IDE

Данная среда программирования является достаточно удобной и гибкой, для работы пользователю предоставляется обширный набор встроенных функций и специализированных библиотек, с возможностью добавления новых или создание собственных под разнообразные электронные устройства [16].

Разработанный алгоритм работы системы управления автоматизированным стеллажом для гидропоники выглядит следующим образом, программа принимает прописанные уставки и начинает отслеживать состояние окружающего микроклимата и состояние почвы, поддерживая его в необходимом пользователю состоянии [17-20]. На рисунке 51 представлена упрощенная блок-схема алгоритма работы участка разработанной программы.

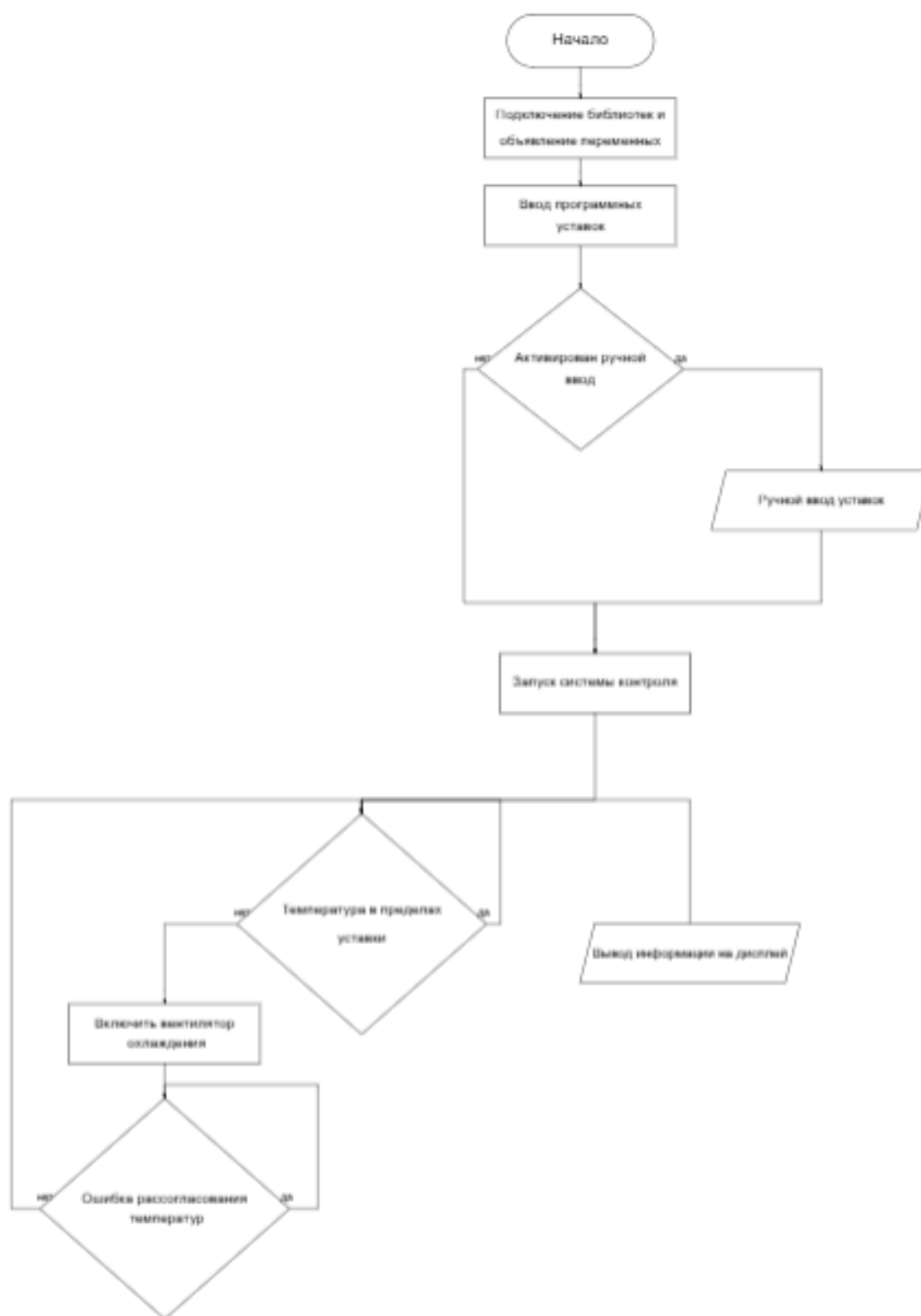


Рисунок 51 – Упрощенная блок-схема алгоритма работы системы обдува

На рисунке 52 представлен алгоритм контроля и поддержания влажности субстрата.

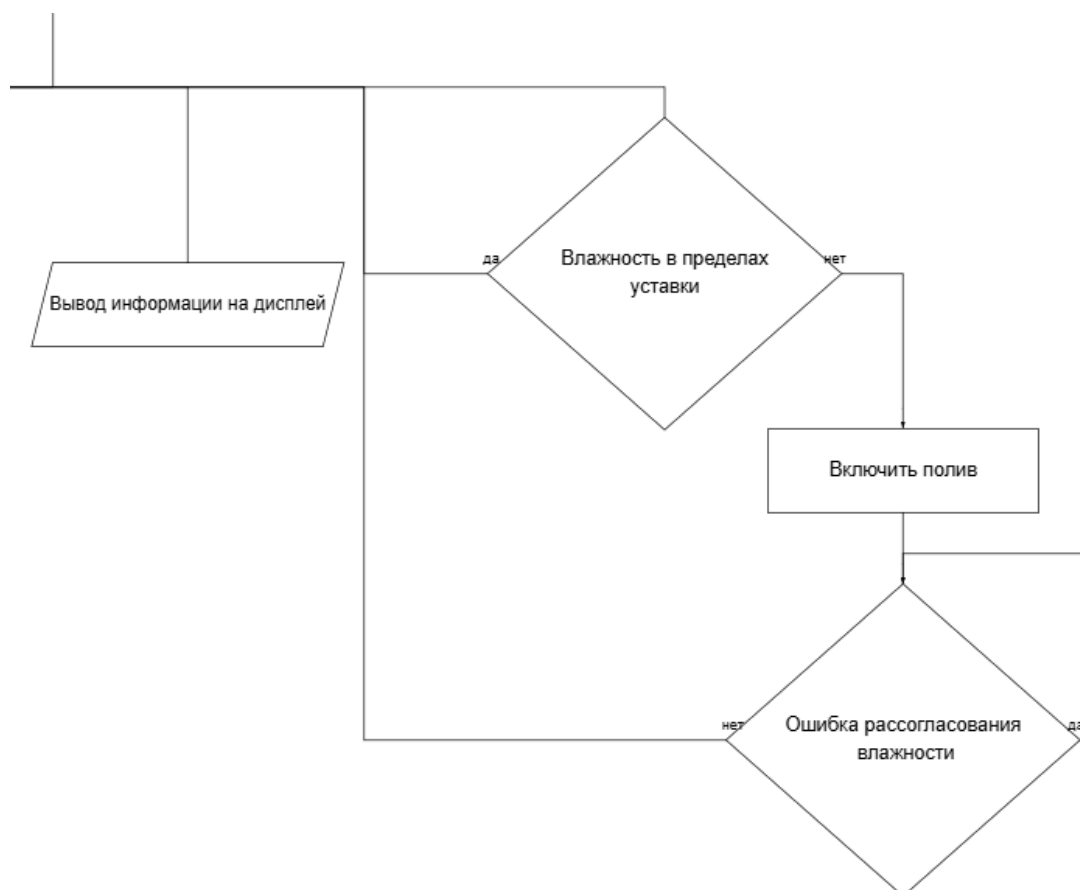


Рисунок 52 – Алгоритм контроля и поддержания влажности субстрата

Разработанные алгоритмы контроля для каждого из параметров реализованы схожим образом, исходя из этого приведем общий принцип и алгоритм работы контроля и поддержания того или иного параметра системой управления, представленный на рисунке 53.



Рисунок 53 – Алгоритм работы систем контроля

Как видно из рисунка выше, управляющая программа в режиме реального времени считывает показания датчика, контролирующего тот, или иной параметр и сравнивает полученное значение с программной уставной. В случае рассогласования двух вышеописанных значений, система выходит в состояние ошибки, на дисплей в экран ошибок выводится соответствующее информационное сообщение, и система стремится вернуть контролируемый параметр в зону программной уставки, выдавая необходимое управляющее воздействие на объект управления. После выхода значения на уставку система выключает управляющее воздействие и переходит в режим активного контроля.

Таким образом, при старте системы запускается информационный экран, где отображаются основные контролируемые параметры. При нажатии функциональной кнопки переключения экранов происходит отчистка всей выводимой информации и загрузка информации о текущих уставках основных параметров, например, температуры с возможностью ее регулировки при помощи функциональных клавиш на панели оператора.

Для реализации возможности условного переключения нескольких информационных экранов, без использования нескольких LCD дисплеев был разработан алгоритм, представленный на рисунке 54.

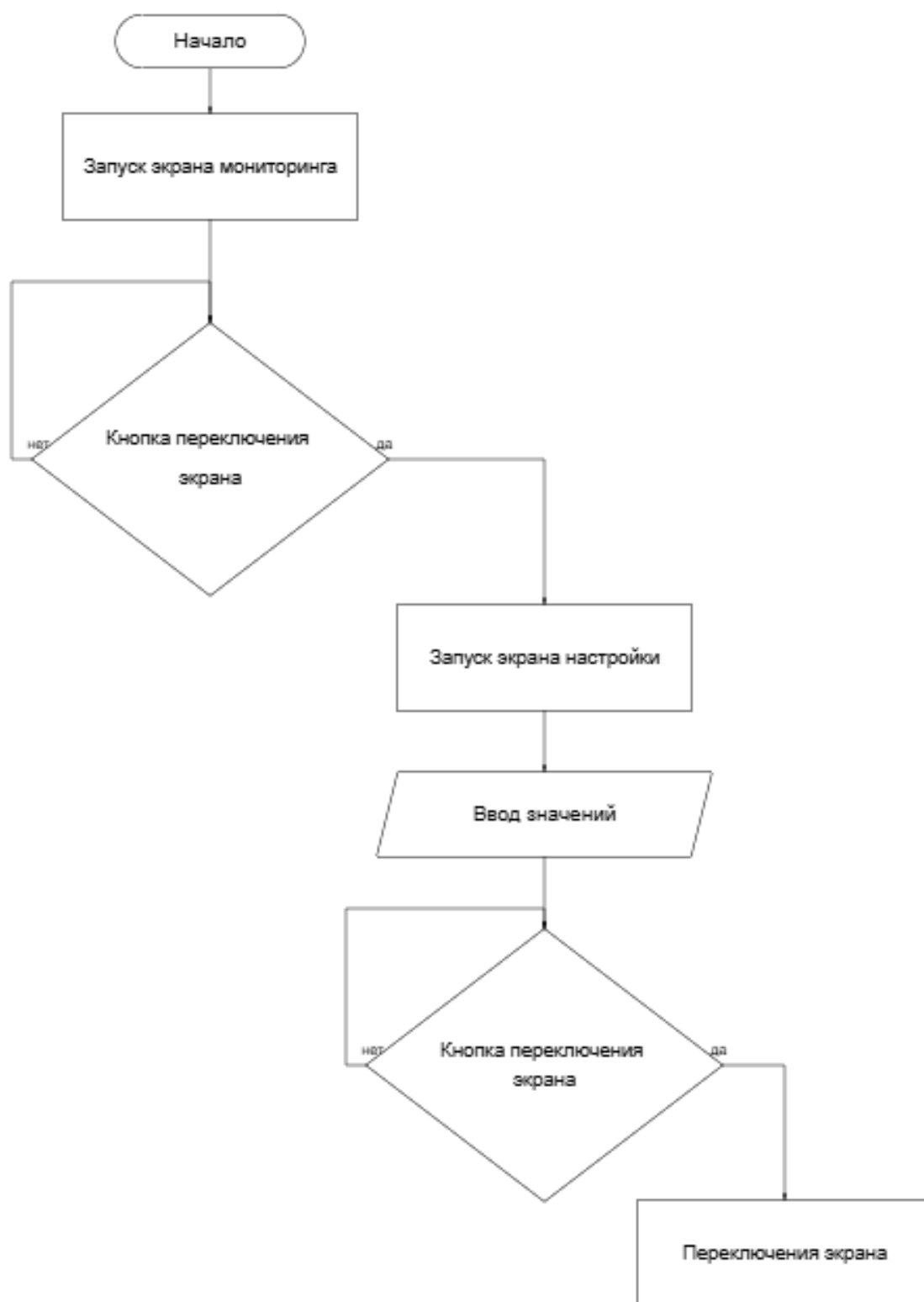


Рисунок 54 – Алгоритм переключения экранов

На рисунке 55 представлен пример реализации панели НМІ с набором необходимых функциональных клавиш и возможностью исполнения как стационарной панели, вмонтированной в электрический шкаф, так и в виде переносного пульта [3].

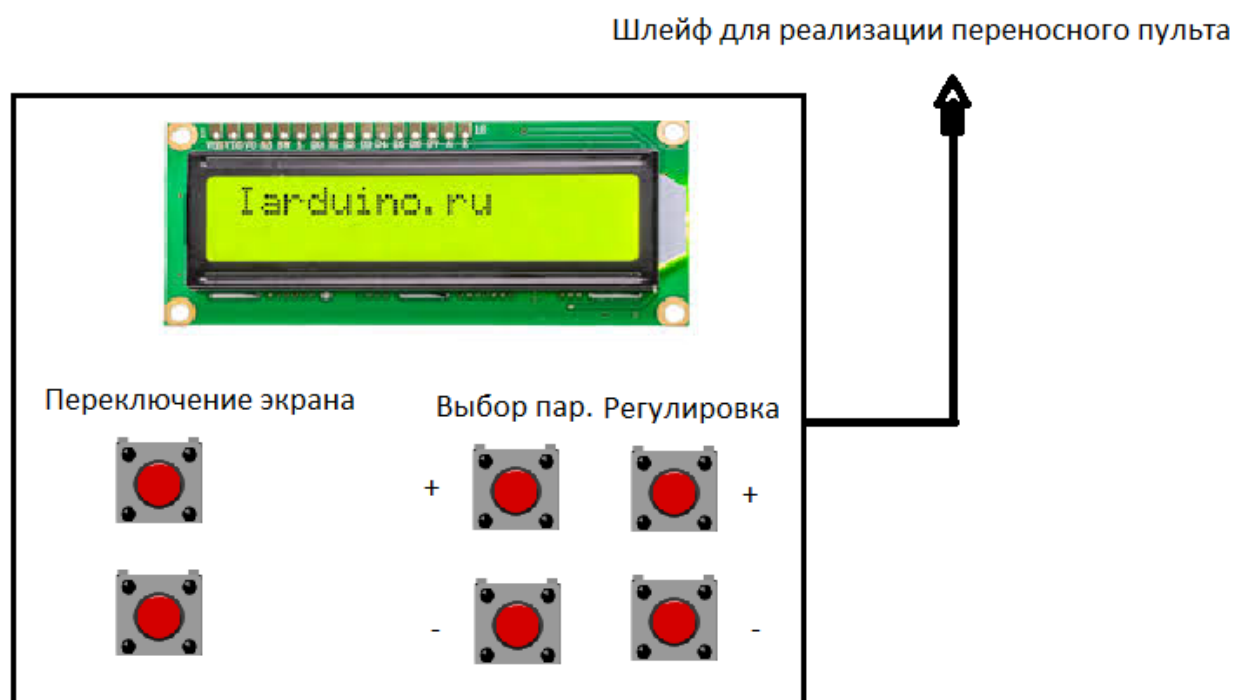


Рисунок 55 – Вариант исполнения панели НМІ

Система дополнительно может быть оснащена системами защитного контроля, к примеру, защитой от переполнения емкостей с питательным раствором или системой защиты от отключения питания за счет внедрения модуля реального времени RTC, обеспечивающего возможность сохранения введенных настроек.

На рисунке 56 представлена примерная компоновка элементов и конструкция автоматизированного стеллажа для гидропоники.

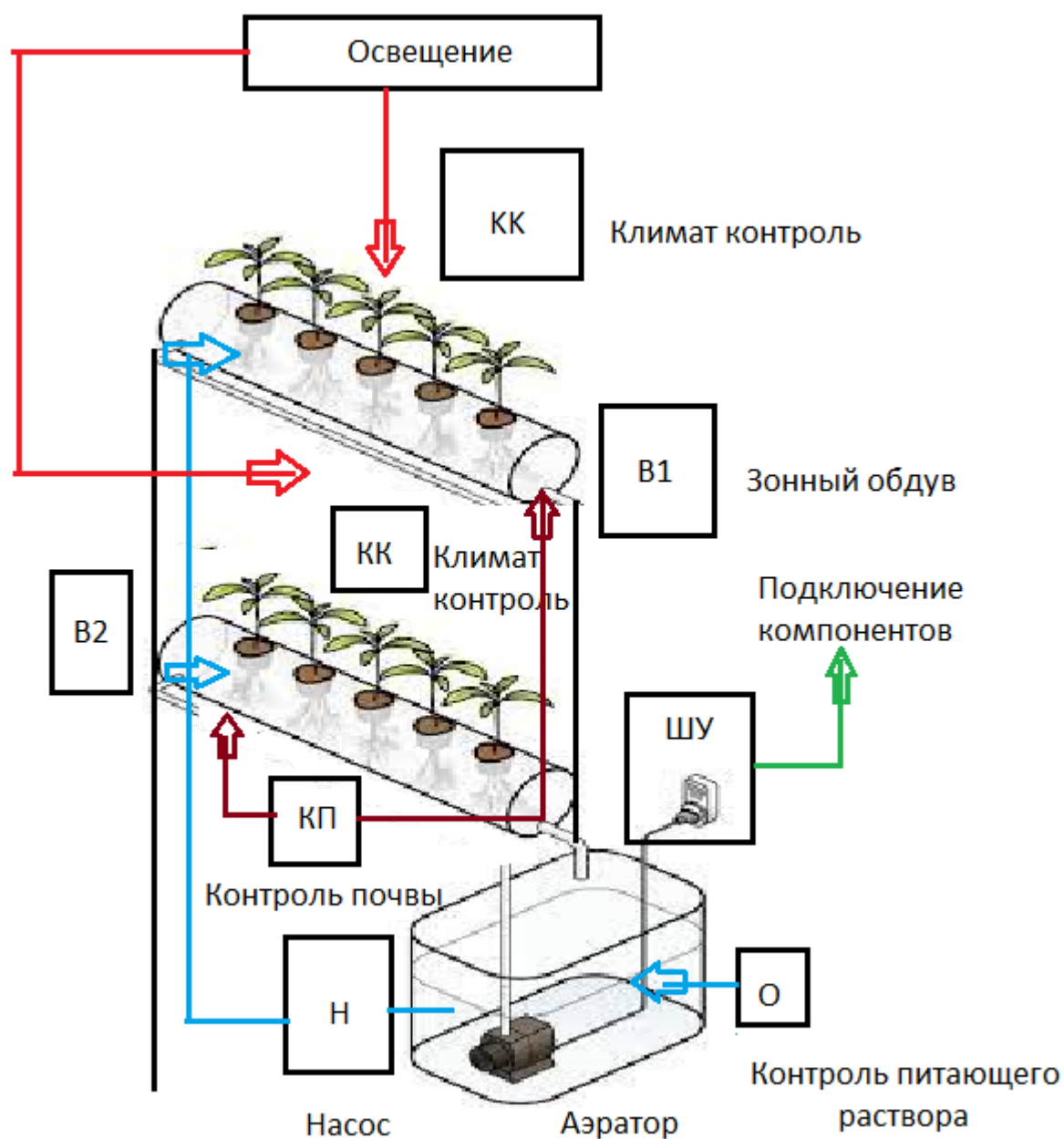


Рисунок 56 – Компоновка и структура стеллажа

#### Выводы по разделу

В данном разделе была рассмотрена реализация управляющей программы автоматизированной системы, специализированных разработанных функций и алгоритмов работы. Была рассмотрена разработанная система HMI и общая компоновка разработанного устройства.

## Заключение

В ходе выполнения выпускной квалификационной работы была достигнута поставленная цель и выполнены все поставленные ранее задачи. Разработана система автоматизированного стеллажа для гидропоники с внедренными системами контроля и автоматизации процесса контроля для выращивания полезных культур в гидропонике.

Был выполнен анализ готовых технических решений в области автоматизированных систем гидропоники, представленных на рынке в текущий момент. На основе полученных данных была определена основная концепция нашего проекта и выделен список задач, которые необходимо решить для достижения поставленной цели и успешной реализации проекта в рамках выполнения данной выпускной квалификационной работы.

Были рассмотрены варианты выбора и установки аппаратного обеспечения системы управления. Была выбрана необходимая нам отладочная плата с микроконтроллером. Было разработано и промоделировано схмотехническое решение цепей обеспечения питания системы от сети 220 В. Был произведен подбор всей необходимой компонентной базы системы управления и систем контроля. Был разработан человеко-машинный интерфейс, необходимый для реализации возможности контроля, мониторинга и управления системой.

Была рассмотрена реализация управляющей программы автоматизированной системы, специализированных разработанных функций и алгоритмов работы. Была рассмотрена разработанная система НМІ и общая компоновка разработанного устройства.

Была предоставлена вся необходимая электротехническая документация, которая, при должной доработке программной и аппаратной составляющей может быть использована для физической реализации автоматизированного стеллажа для гидропоники

## Список используемой литературы

1. Акулов, Г. И. (2020). Влияние автоматизации на эффективность производства в гидропонике. \*Проблемы агрономии и экологии\*, 9(4), 134-140.
2. «Википедия» [Электронный ресурс] // «Человеко-машинный интерфейс» URL: [https://ru.wikipedia.org/wiki/ Человеко-машинный интерфейс](https://ru.wikipedia.org/wiki/Человеко-машинный_интерфейс) (дата обращения: 20.10.2024).
3. «Википедия» [Электронный ресурс] // «Гидропоника» URL: <https://ru.wikipedia.org/wiki/Гидропоника> (дата обращения: 07.01.2025).
4. Водовозов А.М. «Микроконтроллеры для систем автоматики»: Учебное пособие / А.М. Водовозов. - Вологда: ВоГТУ, 2002. - 123 с.
5. Евстифеев А.В. Микроконтроллеры AVR семейств Tiny и Mega фирмы ATMEL, 5-е издание / А.В. Евстифеев. – Москва: Издательский дом «Додэка XXI», 2008. – 558 с.
6. Жданов, П. М., & Семёнова, Л. И. (2021). Проектирование и создание автоматизированных систем для гидропонии. \*Вестник сельскохозяйственной науки\*, 3(1), 45-59.
7. Иванов, А. В., & Петрова, Е. Н. (2020). Современные технологии автоматизации в агрономии: гидропоника и аквапоника. “Аграрная наука сегодня”, 1(15), 17-24.
8. Кузнецов, О. Ю. (2017). Гидропоника: от основы до практического применения. Москва: Издательство "Наука". 250 с.
9. Ростовцев, И. Д.\*\* (2018). Введение в гидропонику: принципы и технологии. М.: Агротехника. 120 с.
10. Соловьев, С. А.\*\* (2019). Автоматизация систем гидропонии на основе IoT. \*Вестник Московского областного государственного университета. Серия: Технические науки\*, 4(1), 34-45.
11. «Arduino IDE» [Электронный ресурс]. «Download the Arduino IDE» URL: <https://www.arduino.cc/en/main/software> (дата обращения: 16.10.2024).

12. «Arduino oficial store» [Электронный ресурс] // «Arduino UNO» URL: [clck.ru/3M8FCE](http://clck.ru/3M8FCE) (дата обращения: 11.02.2025).
13. Atmel, Corporation 8-bit AVR Microcontroller with 128K Bytes In-System Programmable Flash / Corporation Atmel. – USA: Literature Requests, 2004. – 381 с.
14. «AVR» [Электронный ресурс] // «Учебный курс. Устройство и работа портов ввода-вывода» URL: <https://easyelectronics.ru/avr-uchebnyj-kurs-ustrojstvo-i-rabota-portov-vvoda-vyvoda.html> (дата обращения: 07.01.2025).
15. «Autodesk Instructables» [Электронный ресурс] // «Arduino Controlled Smart Hydroponic Modular System» URL: <https://www.instructables.com/Arduino-Controlled-Smart-Hydroponic-Modular-System/> (дата обращения: 03.04.2025).
16. «GitHub» [Электронный ресурс] // «Simple-Hydroponics-Arduino» URL: <https://github.com/NachtRaveVL/Simple-Hydroponics-Arduino> (дата обращения: 21.03.2025).
17. «Hackster» [Электронный ресурс] // «Hydroponics System Using Arduino» URL: <https://www.hackster.io/apugog/hydroponics-system-using-arduino-ab7487> (дата обращения: 21.03.2025).
18. «MUD» [Электронный ресурс] // «5 Hydroponic Systems With Raspberry Pi and Arduino» URL: <https://www.makeuseof.com/hydroponic-systems-with-raspberry-pi-and-arduino/> (дата обращения: 16.03.2025).
19. «PEA» [Электронный ресурс] // «DIY Hydroponics Garden w/Arduino and IoT» URL: <https://www.programmingelectronics.com/arduino-hydroponic-garden/> (дата обращения: 18.03.2025).
20. «RoboCraft» [Электронный ресурс] // «Автоматизация гидропоники при помощи Arduino» URL: <https://robocraft.ru/projects/3838> (дата обращения: 07.04.2025).

Приложение А

**Перечень элементов к схеме электрической принципиальной  
стеллажа для гидропоники**

Взам. инв. №

Подп. и дата

Изм. № подл.

| Поз. обозначение                           | Наименование                                    | Кол-во | Примечание |
|--------------------------------------------|-------------------------------------------------|--------|------------|
| <u>Модули</u>                              |                                                 |        |            |
| A1A2                                       | DC/DC преобразователь напряжения на базе LM2569 | 2      |            |
| A3                                         | Датчик температуры и влажность DHT11            | 1      |            |
| A4                                         | LCD Дисплей 1602                                | 1      |            |
| A5                                         | Интерфейсный модуль I2C Arduino                 | 1      |            |
| A6                                         | Отладочная плата Arduino UNO ATmega328          | 1      |            |
| A7                                         | Емкостной датчик влажности YL69 Arduino         | 1      |            |
| A8                                         | Поплавковый уровень 5В Arduino                  | 1      |            |
| A9                                         | Модуль реального времени Arduino RST DS1307     | 1      |            |
| <u>Реле</u>                                |                                                 |        |            |
| K1K2                                       | PIC AVR DSP ARM MCU DC AC 220 В                 | 2      |            |
| K3-K6                                      | TZT 5 В 12 В 1                                  | 4      |            |
| <u>Трансформаторы</u>                      |                                                 |        |            |
| T1                                         | Трансформатор напряжения ТПК-1                  | 1      |            |
| <u>Диоды</u>                               |                                                 |        |            |
| VDS1                                       | Диодная сборка (мост) SUNMATE BR1001            | 1      |            |
| <u>Лампы</u>                               |                                                 |        |            |
| HL1HL2                                     | Ultraflash, LED, дневной белый, 40 Вт 220 В     | 2      |            |
| <u>Резисторы</u>                           |                                                 |        |            |
| R1-R5                                      | 10K CF 1/4 W 5%                                 | 5      |            |
| <u>Конденсаторы</u>                        |                                                 |        |            |
| C1                                         | JAMICON K73-17 100нФ                            | 1      |            |
| <u>Кнопки</u>                              |                                                 |        |            |
| SA1-SA5                                    | KAN0611-0431В                                   | 5      |            |
| 25-110304.619.158.02 ПЗЗ                   |                                                 |        |            |
| Автоматизированный стеллаж для гидропоники |                                                 | Стадия | Масса      |
| Перечень элементов                         |                                                 | Лист 1 | Листов 2   |
| ТГУ гр. Элбп-2002а                         |                                                 |        |            |

Рисунок А.1 – Перечень элементов

### Продолжение Приложения А

[illegible]

## Рисунок А.2 – Перечень элементов

## Приложение Б

### Листинг участка управляющей программы

```
// объявляем служебные переменные и подключаем библиотеки
int32_t Htime; //служебная переменная времени
int32_t Ltime; // служебная переменная времени
float Ttime; // служебная переменная времени
float frequency; // служебная переменная частота
float capacitance; // служебная переменная емкость
#include <MozziGuts.h> //подключение библиотеки
#include <Oscil.h> //подключение библиотеки
#include <tables/sin2048_int8.h> //подключение библиотеки
#include <RollingAverage.h> //подключение библиотеки
#include <ControlDelay.h> //подключение библиотеки
unsigned int echo_cells_1 = 32; // объявление переменной типа uns.int
unsigned int echo_cells_2 = 60; // объявление переменной типа uns.int
unsigned int echo_cells_3 = 127; // объявление переменной типа uns.int
#define CONTROL_RATE 64 // объявление служебной константы
//объявление служебных переменных
ControlDelay <128, int> kDelay;
Oscil <SIN2048_NUM_CELLS, AUDIO_RATE> aSin0(SIN2048_DATA);
Oscil <SIN2048_NUM_CELLS, AUDIO_RATE> aSin1(SIN2048_DATA);
Oscil <SIN2048_NUM_CELLS, AUDIO_RATE> aSin2(SIN2048_DATA);
Oscil <SIN2048_NUM_CELLS, AUDIO_RATE> aSin3(SIN2048_DATA);
RollingAverage <int, 32> kAverage;
int averaged;
void setup()
{
  pinMode(8,INPUT); //pin 8 – на ввод данных
  kDelay.set(echo_cells_1);
  startMozzi();
  #include <Wire.h> // Подключаем библиотеку для работы с аппаратной
  шиной I2C.
```

## Продолжение Приложения Б

```
#include <iarduino_I2C_Encoder.h>           // Подключаем библиотеку для работы с
энкодером I2C-flash.

#include <LiquidCrystal_I2C.h>             // Подключаем библиотеку для работы с LCD
дисплеем по шине I2C

#include <iarduino_RTC.h>                   // Подключаем библиотеку для работы с
модулем реального времени

#include <EEPROM.h>                         // Подключаем библиотеку для работы с
энергонезависимой памятью

#define INIT_ADDR 1023                     // номер резервной ячейки (для EEPROM)
#define INIT_KEY 26                       // ключ первого запуска. 0-254, на выбор (для
EEPROM)

iarduino_I2C_Encoder enc(0x09);           // Объявляем объект енс для работы с
функциями и методами библиотеки iarduino_I2C_Encoder, указывая адрес модуля на шине
I2C.

LiquidCrystal_I2C lcd(0x27,20,4);         // Объявляем объект библиотеки, указывая
параметры дисплея (адрес I2C = 0x27, количество столбцов = 20, количество строк = 4)

iarduino_RTC watch(RTC_DS3231);           // Объявляем объект watch для модуля на
базе чипа DS3231

// Определяем системное время:              // Время загрузки скетча.
const char* strM="JanFebMarAprMayJunJulAugSepOctNovDec"; // Определяем
массив всех вариантов текстового представления текущего месяца.

const char* sysT=__TIME__;                 // Получаем время компиляции скетча
в формате "SS:MM:HH".

const char* sysD=__DATE__;                 // Получаем дату компиляции скетча
в формате "MMM:DD:YYYY", где MMM - текстовое представление текущего месяца,
например: Jul.
```

## Продолжение Приложения Б

```
const int i[6] {(sysT[6]-48)*10+(sysT[7]-48), (sysT[3]-48)*10+(sysT[4]-48), (sysT[0]-48)*10+(sysT[1]-48), (sysD[4]-48)*10+(sysD[5]-48), ((int)memmem(strM,36,sysD,3)+3-(int)&strM[0])/3, (sysD[9]-48)*10+(sysD[10]-48)};
```

```
#define lampPin 7          // Реле (освещение) 7 пин
#define pompPin 3          // Насос долива воды 3 пин
#define levelPin 4         // Поплавковый датчик уровня 4 пин
const uint8_t pinConsupt = 2;    // Определяем № вывода Arduino, к которому
подключён датчик расхода воды
```

```
//////////НАСТРОЙКИ//////////
uint8_t timeOn = 7;           // Время ВКЛючения освещения и подсветки дисплея
---РЕКОМЕНДУЕМОЕ ВРЕМЯ ОСВЕЩЕНИЯ РАСТЕНИЙ 16 ЧАСОВ В СУТКИ---
uint8_t timeOFF = 23;         // Время ВЫКЛючения освещения и подсветки дисплея
(если установлен флаг выключать LCD на ночь)
#define lcdOffFlag true       // Флаг выключения подсветки LCD на ночь: true если
выключать ночью, false - если не выключать ночью
#define floatSensor true      // Наличие поплавкового уровнемера
uint8_t grownPlantDays = 30;  // Необходимо дней для выращивания растения
(согласно инструкции на упаковке)
uint8_t plantAge = 0;         // Текущий возраст растения на момент запуска системы
(также настраивается в меню настроек)
uint16_t powerLamp = 60;      // Мощность используемых ламп
#define waterLowLevel 620     // Уровень воды "сухого" датчика
#define waterHighLevel 365    // Уровень воды полностью погруженного в воду
датчика
uint16_t waterSetLevel = 460;  // Установленный уровень воды
#define timeLcdOn 60          // Время работы подсветки дисплея после активности,
с
#define deltaWaterLevel 5     // Разница установленного и реального уровней, при
которой включится насос
#define pompMaxTime 20        // Максимальное время работы насоса,
```

## Продолжение Приложения Б

```
uint8_t D, M, Y, h, m, s;           // Объявляем переменные для получения даты и
времени: D-день, M-месяц, Y-год, h-часы, m-минуты, s-секунды

uint8_t old_s, old_h, old_D;        // Предыдущее значение секунд, часов, дней

uint32_t timeBlink;                 // Временная переменная для периода мигания курсора

bool visible = true;                // Переменная видимости курсора

bool sChangeFlag = true, hChangeFlag = true, DChangeFlag = true; // флаг изменения
переменных - секунды, часы, дни.

bool homeChangeFlag = true; bool mainMenuChangeFlag = false; bool screen_5_Update
= false; bool screen_3_Update = false; bool screen_6_Update = false; bool screen_7_Update =
false; bool screen_4_Update = false; bool screen_8_Update = false; // Флаги отрисовки экранов

bool lampState = false; bool pompState = false; // Состояния реле - лампы и силового
ключа - насоса

uint8_t screen = 5;                 // Текущий видимый экран

int8_t cursorPosition = 0;          // Текущая позиция курсора (отвечает за выбор
активного пункта меню)

int8_t encSteps;                    // Текущее значение шагов энкодера

bool cursorActive = false;          // Флаг активности курсора (при горизонтальном
перемещении стрелки в главном меню)

float waterConsumpt = 0.0;           // Текущий расход воды

float lumConsumpt = 0.0;            // Текущий расход энергии на освещение

uint16_t waterLevel;               // Значение с датчика уровня

uint16_t waterLevelPercent;         // Уровень воды, в %

volatile uint16_t varCount = 0;     // Определяем переменную для подсчёта количества
импульсов поступивших от датчика расхода

uint8_t countTacts = 0;             // Счётчик тактов для задержки срабатывания насоса
при изменении уровня

uint32_t timeOnBackLigth;           // Время включения подсветки дисплея

uint32_t pompStartTime;             // Время включения насоса

bool pompError = false;             // Флаг ошибки насоса (работал слишком долго без
изменения уровня)

bool startFlag = true;              // Флаг при включении контроллера

bool noSetFlag = true;
```

## Продолжение Приложения Б

```
uint8_t symbol[7][8] = {          // Задаём пользовательские символы (русские
недостающие буквы)

    {B01111,B00101,B00101,B01001,B10001,B11111,B10001,B00000}, // Д
    {B11111,B10001,B10001,B10001,B10001,B10001,B10001,B00000}, // П
    {B10001,B10011,B10011,B10101,B11001,B11001,B10001,B00000}, // И
    {B10000,B10000,B10000,B11110,B10001,B10001,B11110,B00000}, // Ъ
    {B10101,B10101,B10101,B10101,B10101,B10101,B11111,B00001}, // Щ
    {B11111,B10001,B10000,B10000,B10000,B10000,B10000,B00000}, // Г
    {B00011,B00111,B00101,B00101,B01101,B01001,B11001,B00000}, // Л
};

long x = 0;

void setup() {
    pinMode(lampPin, OUTPUT);      // Реле (освещение) - выход
    pinMode(pompPin, OUTPUT);      // Силовой ключ (насос) - выход
    pinMode(levelPin, INPUT_PULLUP); // Поплавковый уровнемер - вход, подтяжка
включена

    watch.begin();                // Иницилируем RTC модуль
    //watch.settime(i[0],i[1],i[2],i[3],i[4],i[5]); // Устанавливаем время в модуль: i[0]
сек, i[1] мин, i[2] час, i[3] день, i[4] месяц, i[5] год, без указания дня недели.

    Serial.begin(9600);           // Инициализация последовательного порта
    enc.begin();                  // Иницилируем работу с энкодером
    lcd.init();                   // Иницилируем работу с LCD дисплеем
    lcd.clear();                  // Очищаем дисплей
    lcd.backlight();              // Включаем подсветку LCD дисплея
    lcd.createChar(1, symbol[0]); // Загружаем пользовательские символы в ОЗУ
дисплея
```

## Продолжение Приложения Б

```
lcd.createChar(2, symbol[1]);
lcd.createChar(3, symbol[2]);
lcd.createChar(4, symbol[3]);
lcd.createChar(5, symbol[4]);
lcd.createChar(6, symbol[5]);
lcd.createChar(7, symbol[6]);

if (EEPROM.read(INIT_ADDR) != INIT_KEY) { // первый запуск
EEPROM.write(INIT_ADDR, INIT_KEY);      // записываем ключ
eepromUpdate();
Serial.println ("Данные обновлены");
}
else{
EEPROM.get(0, waterConsumpt);
EEPROM.get(4, lumConsumpt);
EEPROM.get(8, timeOn);
EEPROM.get(9, timeOFF);
EEPROM.get(10, powerLamp);
EEPROM.get(12, plantAge);
EEPROM.get(13, grownPlantDays);
EEPROM.get(14, waterSetLevel);
Serial.println ("Данные считаны");
}

pinMode(pinConsumpt, INPUT);              // Конфигурируем вывод к которому
подключён датчик, как вход

uint8_t intConsumpt = digitalPinToInterrupt(pinConsumpt); // Определяем № прерывания
который использует вывод pinSensor

attachInterrupt(intConsumpt, funCountInt, RISING);          // Назначаем функцию
funCountInt как обработчик прерываний intConsumpt при каждом выполнении условия
RISING - переход от 0 к 1
```

## Продолжение Приложения Б

```
screen_5_Update = true;           // Для отрисовки экрана посадки при
запуске
watch.gettime();                  // Считываем текущее время из модуля.
D = watch.day;                    // Получаем текущий день месяца 1-31.
old_D = D;
}
void funCountInt(){varCount++;}   // Определяем функцию, которая
будет приращать значение счётчика импульсов с расходомера

void eepromUpdate(){
EEPROM.put(0, waterConsumpt);
EEPROM.put(4, lumConsumpt);
EEPROM.put(8, timeOn);
EEPROM.put(9, timeOFF);
EEPROM.put(10, powerLamp);
EEPROM.put(12, plantAge);
EEPROM.put(13, grownPlantDays);
EEPROM.put(14, waterSetLevel);
}

void loop(){
waterLevel = constrain(analogRead(A0), waterHighLevel, waterLowLevel); //
Ограничиваем значение с ёмкостного датчика уровня минимальным и максимальным
значением
watch.gettime();                  // Считываем текущее время из модуля.
D = watch.day;                    // Получаем текущий день месяца 1-31.
M = watch.month;                  // Получаем текущий месяц    1-12.
Y = watch.year;                   // Получаем текущий год      0-99.
h = watch.Hours;                  // Получаем текущие часы    0-23.
m = watch.minutes;                // Получаем текущие минуты   0-59.
s = watch.seconds;                // Получаем текущие секунды   0-59.
```