

**Минобрнауки России**  
**Юго-Западный государственный университет**

Кафедра программной инженерии

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**  
**ПО ПРОГРАММЕ БАКАЛАВРИАТА**

**09.03.04 Программная инженерия**

(код, наименование ОПОП ВО: направление подготовки, направленность (профиль))

**«Разработка программно-информационных систем»**

**Разработка web-сайта «Русатом – Аддитивные технологии» на платформе**

**1С-Битрикс**

(название темы)

**Дипломный проект**

(вид ВКР: дипломная работа или дипломный проект)

**Автор ВКР**

**И. И. Иванов**

(подпись, дата)

(инициалы, фамилия)

Группа ПО-016

**Руководитель ВКР**

**М. В. Бобырь**

(подпись, дата)

(инициалы, фамилия)

**Нормоконтроль**

**А. А. Чаплыгин**

(подпись, дата)

(инициалы, фамилия)

**ВКР допущена к защите:**

**Заведующий кафедрой**

**А. В. Малышев**

(подпись, дата)

(инициалы, фамилия)

Курск 2024 г.

**Минобрнауки России**  
**Юго-Западный государственный университет**

Кафедра программной инженерии

УТВЕРЖДАЮ:  
Заведующий кафедрой

---

(подпись, инициалы, фамилия)

«\_\_\_\_\_» 20\_\_\_\_ г.

**ЗАДАНИЕ НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ  
РАБОТУ ПО ПРОГРАММЕ БАКАЛАВРИАТА**

Студента Иванова И.И., шифр xx-xx-xxxx, группа ПО-016

1. Тема «Разработка web-сайта «Русатом – Аддитивные технологии» на платформе 1С-Битрикс» утверждена приказом ректора ЮЗГУ от «07» апреля 2023 г. № 1505-с.

2. Срок предоставления работы к защите «13» июня 2023 г.

3. Исходные данные для создания программной системы:

3.1. Перечень решаемых задач:

- 1) проанализировать ИТ-инфраструктуру предприятия;
- 2) разработать концептуальную модель системы управления ИТ-инфраструктурой предприятия на основе подхода к управлению и организации ИТ-услуг ITSM;
- 3) спроектировать программную систему управления ИТ-инфраструктурой предприятия;
- 4) сконструировать и протестировать программную систему управления ИТ-инфраструктурой предприятия.

3.2. Входные данные и требуемые результаты для программы:

- 1) Входными данными для программной системы являются: данные справочников комплектующих, конфигураций, ПО, критериев качества SLA,

ИТ-услуг, департаментов компании; технические данные ИТ-ресурсов; данные входящих заявок на ИТ-ресурсы; данные запросов поставщикам на комплектующие.

2) Выходными данными для программной системы являются: сформированные заявки на обслуживание ИТ-ресурсов; сформированные запросы на закупку комплектующих; сведения о выполненных работах по заявкам; статусы заявок; выходные отчеты (инфографика) – по качеству услуг, по состоянию ИТ-ресурсов, по деятельности ИТ-отдела, по стоимости обслуживания ИТ-ресурсов, воронка заявок.

4. Содержание работы (по разделам):

#### 4.1. Введение

4.1. Анализ предметной области

4.2. Техническое задание: основание для разработки, назначение разработки, требования к программной системе, требования к оформлению документации.

4.3. Технический проект: общие сведения о программной системе, проект данных программной системы, проектирование архитектуры программной системы, проектирование пользовательского интерфейса программной системы.

4.4. Рабочий проект: спецификация компонентов и классов программной системы, тестирование программной системы, сборка компонентов программной системы.

#### 4.5. Заключение

4.6. Список использованных источников

5. Перечень графического материала:

Руководитель ВКР

(подпись, дата)

М. В. Бобырь

(инициалы, фамилия)

Задание принял к исполнению

(подпись, дата)

И. И. Иванов

(инициалы, фамилия)

## РЕФЕРАТ

Объем работы равен 65 страницам. Работа содержит 21 иллюстрацию, 2 таблицы, 12 библиографических источников и 0 листов графического материала. Количество приложений – 2. Графический материал представлен в приложении А. Фрагменты исходного кода представлены в приложении Б.

Перечень ключевых слов: Arduino, Система, CMS, Битрикс, Joomla, аддитивные технологии, 3D-принтеры, услуги, сервисы, информатизация, автоматизация, информационные технологии, веб-форма, Apache, классы, база данных, средства защиты информации, подсистема, компонент, модуль, сущность, информационный блок, метод, контент-редактор, администратор, пользователь, web-сайт.

Объектом разработки является web-сайт компании, занимающейся производством 3D-принтеров, выпуском оборудования для создания порошков, разработкой программного обеспечения и организацией центров аддитивного производства.

Целью выпускной квалификационной работы является привлечение клиентов, увеличение заказов, информирование о продукции и услугах путем создания сайта компании.

В процессе создания сайта были выделены основные сущности путем создания информационных блоков, использованы классы и методы модулей, обеспечивающие работу с сущностями предметной области, а также корректную работу web-сайта, разработаны разделы, содержащие информацию о компании, ее деятельности, производимой продукции и услугах, разработан сервис по заказу 3D-деталей.

При разработке сайта использовалась система управления контентом «1С-Битрикс: Управление сайтом».

Разработанный сайт был успешно внедрен в компании.

## ABSTRACT

The volume of work is 65 pages. The work contains 21 illustration, 2 tables, 12 bibliographic sources and 0 sheets of graphic material. The number of applications is 2. The graphic material is presented in annex A. The layout of the site, including the connection of components, is presented in annex B.

List of keywords: commercial website, System, CMS, Bitrix, Joomla, additive technologies, 3D printers, services, services, informatization, automation, information technology, web form, Apache, classes, database, component, module, entity, information block, method, content editor, administrator, user, web site.

The object of the research is the analysis of information technologies for the development of a production company's website.

The object of the development is the website of a company engaged in the production of 3D printers, the production of equipment for the creation of powders, software development and the organization of additive manufacturing centers.

The purpose of the final qualifying work is to attract customers, increase orders, inform about products and services by creating a company website.

In the process of creating the site, the main entities were identified by creating information blocks, classes and methods of modules were used to ensure work with the entities of the subject area, as well as the correct operation of the website, sections containing information about the company, its activities, products and services were developed, a service for ordering 3D parts was developed.

When developing the site, the content management system «1C – Bitrix: Site Management» was used.

The developed website was successfully implemented in the company.

## СОДЕРЖАНИЕ

<b>ВВЕДЕНИЕ</b>	<b>10</b>
1 Анализ предметной области	12
1.1 Характеристика предприятия и его деятельности. Тест длинного заголовка, не должен содержать переносы	12
1.2 Технологии Arduino, их классификация	12
1.3 Микроконтроллер ATmega328P	13
1.3.1 Микроконтроллер ATmega328P	13
1.3.2 Питание и управление	13
1.3.3 Программирование и отладка	14
1.3.4 Периферийные устройства и расширения	14
1.4 Программное обеспечение	14
1.4.1 Среды разработки	14
1.4.1.1 Arduino IDE	14
1.4.1.2 PlatformIO	15
1.4.2 Языки программирования	16
1.4.2.1 C++	16
1.4.2.2 Python	16
1.4.2.3 Дополнительные инструменты и экосистема	16
1.4.3.1 ArduBlock	16
1.4.3.2 Codebender	17
1.4.3.3 Firmata	17
1.4.3.4 Arduino Web Editor	17
1.4.3.5 SimulIDE	17
1.5 История развития Arduino	18
1.6 Использование Arduino в России	19
2 Техническое задание	21
2.1 Основание для разработки	21
2.2 Цель и назначение разработки	21
2.3 Функциональные требования к мультироботу	22

2.3.1 Вариант использования: использование после запуска	23
2.3.2 Вариант использования: пользователь установил значение счетчика равное 1	23
2.3.3 Вариант использования: пользователь установил значение счетчика равное 2	24
2.3.4 Вариант использования: пользователь установил значение счетчика равное 3	24
2.3.5 Вариант использования: пользователь установил значение счетчика равное 4	24
2.3.6 Вариант использования: пользователь завершил выполнение 1 из режимов	25
2.3.7 Вариант использования: мультиробот находится в ночном режиме, но уровень яркости растет	25
2.4 Нефункциональные требования к мультироботу	25
2.4.1 Требования к программному обеспечению	25
2.4.2 Требования к аппаратному обеспечению	26
2.4.3 Требования к оформлению документации	26
3 Технический проект	27
3.1 Общая характеристика организации решения задачи	27
3.2 Обоснование выбора технологии проектирования	27
3.2.1 Описание используемых технологий и языков программирования	27
3.2.2 Датчик DHT11 temperature and humidity sensor	27
3.2.3 buzzer	29
3.2.4 sharp 2Y0A02	30
3.2.5 LCD монитор MT 16S2H 2YLG PB	33
3.2.6 фотодиод	36
3.2.7 ультразвуковой датчик HC-SR04	37
3.2.8 сервопривод	39
3.2.9 Язык программирования Arduino (C++)	41
3.2.9.1 Достоинства языка C++	42

3.2.9.2 Недостатки языка C++	42
3.2.10 Система ввода-вывода данных Arduino	44
3.3 Технология загрузчика (bootloader) в Arduino	51
3.3.1 Архитектура микроконтроллера Arduino	51
4 Рабочий проект	54
4.1 Классы, используемые при разработке сайта	54
4.2 Модульное тестирование разработанного мультиробота	56
4.3 Системное тестирование разработанного устройства	57
ЗАКЛЮЧЕНИЕ	58
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	59
ПРИЛОЖЕНИЕ А Представление графического материала	62
ПРИЛОЖЕНИЕ Б Фрагменты исходного кода программы На отдельных листах (CD-RW в прикрепленном конверте)	63
	65

## ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

ide – интегрированная среда разработки

ИС – информационная система.

ИТ – информационные технологии.

КТС – комплекс технических средств.

ПО – программное обеспечение.

РП – рабочий проект.

ТЗ – техническое задание.

ТП – технический проект.

## ВВЕДЕНИЕ

Arduino – это платформа для создания электронных проектов, основанная на простоте и доступности. Она включает в себя аппаратную часть – микроконтроллеры и платы, а также программное обеспечение для их программирования. Платы Arduino: Существует множество различных моделей плат Arduino, например, Arduino Uno, Arduino Mega, Arduino Nano и другие. Каждая плата имеет свои особенности и предназначена для различных задач.

Микроконтроллеры: Сердцем каждой платы является микроконтроллер – небольшой компьютер на одном чипе, который выполняет программы, загружаемые в него.

Arduino IDE: Это интегрированная среда разработки, используемая для написания кода на языке программирования, напоминающем C++. Она позволяет легко писать, компилировать и загружать программы в микроконтроллер.

Arduino широко используется для создания разнообразных проектов – от простых до сложных. В пример можно привести робототехнику: Создание роботов, которые могут перемещаться, избегать препятствий и выполнять другие задачи. IoT: Разработка устройств, которые могут подключаться к интернету и взаимодействовать с другими устройствами. В Arduino IDE встроена возможность создания своих программно-аппаратных платформ. Этой возможностью пользуются сторонние компании, добавляющие в Arduino IDE свои наборы плат и компиляторов-загрузчиков к ним.

Благодаря своей доступности и разнообразию возможностей, Arduino стал популярным выбором для создания инновационных проектов в самых разных областях.

*Цель настоящей работы – разработка мультиробота компании для привлечения новой аудитории, увеличения заказов, рекламы продукции и услуг компании. Для достижения поставленной цели необходимо решить следующие задачи:*

- провести анализ предметной области;

- разработать концептуальную модель робота;
- спроектировать модель работы устройства;
- реализовать механику работы средствами arduino ide.

*Структура и объем работы.* Отчет состоит из введения, 4 разделов основной части, заключения, списка использованных источников, 2 приложений. Текст выпускной квалификационной работы равен 11 страницам.

*Во введении* сформулирована цель работы, поставлены задачи разработки, описана структура работы, приведено краткое содержание каждого из разделов.

*В первом разделе* на стадии описания технической характеристики предметной области приводится сбор информации о деятельности компании, для которой осуществляется разработка.

*В втором разделе* на стадии технического задания приводятся требования к разрабатываемой модели робота.

*В третьем разделе* на стадии технического проектирования представлены проектные решения для web-сайта.

*В четвертом разделе* приводится список классов и их методов, использованных при разработке системы работы, производится тестирование кода.

В заключении излагаются основные результаты работы, полученные в ходе разработки.

В приложении А представлен графический материал. В приложении Б представлены фрагменты исходного кода.

## **1 Анализ предметной области**

### **1.1 Характеристика предприятия и его деятельности. Тест длинного заголовка, не должен содержать переносы**

Arduino был создан в 2005 году группой студентов и преподавателей в Институте взаимодействия дизайна в Ивреа, Италия. Идея принадлежала Массимо Банци (Massimo Banzi) и Дэвиду Куартуйлле (David Cuartielles), которые стремились создать доступный и простой в использовании инструмент для студентов, изучающих электронику и программирование. Первоначально проект был известен как "Wiring" однако затем он был переименован в "Arduino" в честь бара в Ивреа, где разработчики часто встречались. Первая версия платы, Arduino Uno, появилась в 2005 году. Она была основана на микроконтроллере Atmega8 от Atmel (ныне часть Microchip Technology). Важно отметить, что Arduino предоставил пользователям не только аппаратную часть, но и программную среду разработки (IDE), что сделало процесс создания проектов интуитивно понятным.

### **1.2 Технологии Arduino, их классификация**

Arduino – это платформа, состоящая из аппаратной и программной частей, предназначенная для создания проектов на основе микроконтроллеров. Она используется как в образовательных целях, так и в профессиональной разработке прототипов. Платформа Arduino включает в себя разнообразные модели плат, программное обеспечение (Arduino IDE) и обширное сообщество разработчиков. Основным преимуществом Arduino является модульный подход: Arduino позволяет разрабатывать проекты по модульному принципу, что упрощает добавление новых функций и компонентов. Например, можно создать модуль управления двигателем, который затем интегрируется в более крупную систему. Классификация технологий Arduino может быть представлена следующим образом: Платы Arduino - это основа на которой будет делаться устройство, у них есть много различных видов, они также обладают широким спектром возможностей для модификаций, будь то Arduino Nano:

Компактная версия, удобная для интеграции в более мелкие проекты, или же Arduino Leonardo: Обладает встроенной поддержкой USB, что позволяет ей эмулировать клавиатуру или мышь. Сенсоры и модули: различные устройства которые могут быть подключены к плате для того чтобы работать с различными типами данных, будь то датчики окружающей среды, которые дают возможность получить информацию об окружающей влажности или температуре. Коммуникационные модули, которые дают возможность работать с wi-fi, bluetooth. Это лишь некоторые из вариантов подключаемых модулей.

### **1.3 Микроконтроллер ATmega328P**

Arduino Uno — это популярная платформа для разработки микроконтроллерных приложений, основанная на микроконтроллере ATmega328P от компании Atmel. Основные компоненты и архитектура Arduino Uno включают следующие элементы:

#### **1.3.1 Микроконтроллер ATmega328P**

Данный микроконтроллер имеет:

- 8-битное RISC-ядро, работающее на частоте 16 МГц;
- 32 КБ флэш-памяти для хранения прошивки;
- 2 КБ оперативной памяти (SRAM);
- 1 КБ EEPROM для хранения данных.
- Интерфейсы ввода/вывода:
  - 14 цифровых выводов (из которых 6 могут быть использованы в качестве ШИМ-выходов);
  - 6 аналоговых входов для чтения аналоговых сигналов;
  - UART, I2C и SPI интерфейсы для связи с периферийными устройствами.

#### **1.3.2 Питание и управление**

- источники питания: плата может быть запитана от USB-порта или внешнего источника питания (7-12 В);

- регулятор напряжения: стабилизатор напряжения 5 В для микроконтроллера и периферийных устройств;
- резервное питание: встроенный резистор и диод для защиты и стабилизации входящего напряжения.

### **1.3.3 Программирование и отладка**

- загрузчик: предустановленный бутлоадер позволяет прошивать микроконтроллер через USB-интерфейс без необходимости использования внешнего программатора;
- среда разработки: Arduino IDE предоставляет простой интерфейс для написания, компиляции и загрузки программ (скетчей) в микроконтроллер;
- библиотеки: Arduino предоставляет обширную библиотеку для работы с различными датчиками, модулями и периферийными устройствами.

### **1.3.4 Периферийные устройства и расширения**

Arduino Uno поддерживает подключение различных расширительных плат (шилдов), которые расширяют функциональность базовой платы, добавляя новые возможности, такие как беспроводная связь, управление моторами, дисплеи и многое другое.

## **1.4 Программное обеспечение**

### **1.4.1 Среды разработки**

#### **1.4.1.1 Arduino IDE**

Arduino IDE – основная среда разработки для программирования плат Arduino. Arduino IDE предоставляет простой и интуитивно понятный интерфейс, который позволяет пользователям легко писать, загружать и отлаживать код. Она поддерживает широкий спектр плат Arduino и имеет встроенную библиотеку примеров, что делает её отличным инструментом как для новичков, так и для опытных разработчиков. В Arduino IDE доступны разнообразные библиотеки и функции, такие как Serial Monitor для отладки, возмож-

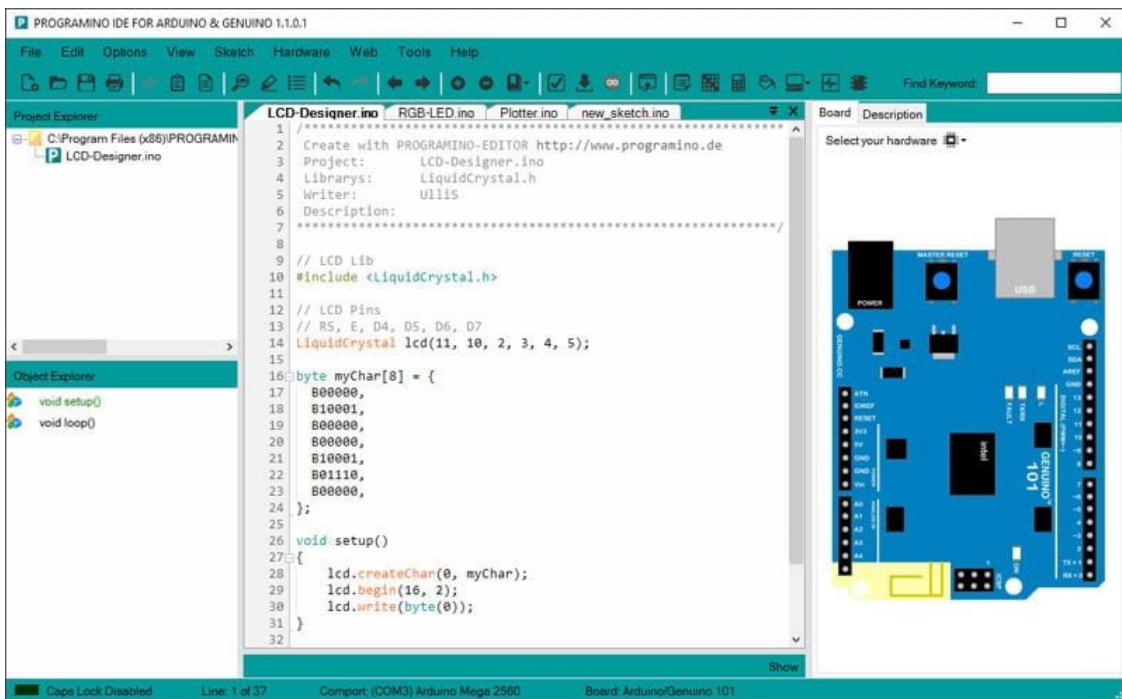


Рисунок 1.1 – Среда разработки Arduino IDE

ности расширения через дополнительные плагины и интеграция с онлайн-ресурсами для поиска и установки библиотек.

На рисунке 1.1 представлена среда разработки Arduino IDE.

### 1.4.1.2 PlatformIO

Альтернативной средой разработки с поддержкой различных плат и расширенными возможностями является PlatformIO. PlatformIO интегрируется с популярными редакторами кода, такими как Visual Studio Code, и предлагает дополнительные функции, включая управление зависимостями, поддержку множества различных микроконтроллеров и удобные инструменты для тестирования и отладки. PlatformIO особенно полезна для проектов, требующих более сложной конфигурации и интеграции с различными платформами. В дополнение к этому, PlatformIO поддерживает удалённое управление и мониторинг, CI/CD, автоматизацию сборки и другие возможности, упрощающие процесс разработки.

## **1.4.2 Языки программирования**

### **1.4.2.1 C++**

Основной язык для разработки скетчей Arduino. Этот язык представляет собой упрощённый вариант C++, адаптированный для работы с микроконтроллерами. В Arduino IDE разработчики пишут программы, называемые "скетчами" которые затем загружаются на плату Arduino для выполнения. Синтаксис языка C++ позволяет эффективно работать с аппаратными ресурсами платы и использовать многочисленные библиотеки для расширения функциональности. К примеру, библиотеки для работы с датчиками, модулями связи и управления устройствами существенно упрощают разработку проектов.

### **1.4.2.2 Python**

Используется с платами, поддерживающими MicroPython (например, ESP32). MicroPython - это реализация языка Python, оптимизированная для работы на микроконтроллерах. Платы, такие как ESP32 и ESP8266, поддерживают выполнение кода на Python, что делает разработку более гибкой и удобной для тех, кто предпочитает Python. Использование Python позволяет быстро прототипировать идеи и интегрировать микроконтроллеры с другими системами и сервисами. Python предлагает простоту и читабельность кода, что ускоряет процесс разработки и отладки, а также обеспечивает доступ к огромному количеству библиотек и фреймворков, таких как Flask для веб-разработки или Pandas для анализа данных.

## **1.4.3 Дополнительные инструменты и экосистема**

### **1.4.3.1 ArduBlock**

Это графический интерфейс для Arduino IDE, предназначенный для облегчения обучения программированию. ArduBlock предоставляет блоки для различных функций и позволяет создавать программы, просто перетаскивая

эти блоки на рабочее поле. Это особенно полезно для начинающих и детей, делая процесс обучения более интуитивным и визуальным.

#### **1.4.3.2 Codebender**

Это онлайн-среда разработки для Arduino, которая позволяет писать и компилировать код прямо в браузере. Codebender поддерживает множество различных плат Arduino и предлагает возможность сохранять проекты в облаке, что упрощает совместную работу и доступ к проектам с любого устройства.

#### **1.4.3.3 Firmata**

Это протокол для обмена данными между программами на компьютере и микроконтроллерами Arduino. Firmata позволяет использовать Arduino в качестве периферийного устройства, управляемого из программы на компьютере, написанной на любом языке, поддерживающем работу с последовательным портом, таких как Python, Processing, или Max/MSP.

#### **1.4.3.4 Arduino Web Editor**

Это онлайн-версия Arduino IDE, которая позволяет писать, загружать и управлять проектами через веб-браузер. Arduino Web Editor интегрирован с Arduino Create и предоставляет доступ к последним версиям библиотек и примеров, обеспечивая удобную работу с проектами из любого места.

#### **1.4.3.5 SimulIDE**

Это эмулятор для Arduino и других микроконтроллеров, который позволяет тестировать и отлаживать код без необходимости физического устройства. SimulIDE предоставляет визуальное представление схемы и позволяет моделировать работу различных компонентов и датчиков.

Эти инструменты и среды разработки делают экосистему Arduino универсальной и доступной для пользователей с любым уровнем опыта, от новичков до профессиональных разработчиков.

## 1.5 История развития Arduino

Arduino - это открытая аппаратная и программная платформа, которая используется для создания различных электронных проектов. История её развития началась в начале 2000-х годов и включает несколько ключевых этапов.

В 2003 году проект Arduino был запущен в Interaction Design Institute Ivrea (IDI) в Италии. Его основателями стали Массимо Банци, Дэвид Куартильес, Том Иго, Джанлука Мартино и Дэвид Мелис. Цель заключалась в создании простой и дешёвой микроконтроллерной платформы для студентов и любителей. Первый прототип назывался "Wiring" который позднее был упрощен и переименован в "Arduino".

Первая плата Arduino была выпущена в 2005 году под названием Arduino Serial. Она использовала микроконтроллер Atmega8 и подключалась к компьютеру через последовательный интерфейс. Позже появилась Arduino Diecimila, которая была оснащена микроконтроллером Atmega168 и подключением через USB.

В этот период Arduino стал невероятно популярным среди любителей электроники и профессионалов. Появились новые модели плат, такие как Arduino Uno, Arduino Mega и другие. В 2011 году была основана компания Arduino LLC для управления производством и развитием платформы. Arduino IDE (среда разработки) стала доступна для бесплатного скачивания, что способствовало распространению платформы.

С 2013 года платформа Arduino продолжила расширяться. Появились новые модели, такие как Arduino Due (основанная на 32-битном микроконтроллере), Arduino Leonardo (с поддержкой USB) и многие другие. Также появились специализированные платы для интернета вещей (IoT), такие как Arduino Yun и Arduino MKR.

Сегодня Arduino - это мощная экосистема, включающая разнообразные платы, модули и аксессуары. Платформа используется в образовательных учреждениях, хобби-проектах, профессиональной разработке и исследова-

тельской деятельности. Открытая архитектура и активное сообщество разработчиков способствуют непрерывному развитию и улучшению платформы.

Arduino остаётся одной из наиболее популярных и доступных платформ для разработки электроники, предоставляя пользователям возможность легко создавать и программировать свои собственные устройства.

## 1.6 Использование Arduino в России

Arduino активно используется в России с начала 2010-х годов и нашел широкое применение в различных областях, от образования и научных исследований до промышленности и хобби-проектов.

В образовательных учреждениях России Arduino стал важным инструментом для обучения программированию и электронике. Многочисленные школы, колледжи и университеты внедрили курсы по Arduino в свои учебные программы. Это позволяет студентам и школьникам получать практические навыки работы с микроконтроллерами и создавать собственные проекты.

В научных исследованиях и разработках Arduino также нашел широкое применение. Исследователи используют платформу для создания прототипов и проведения экспериментов. Например, Arduino используется в области робототехники, биомедицинских исследований и автоматизации научных лабораторий.

В промышленности Arduino применяется для создания автоматизированных систем и контроля процессов. Малый и средний бизнес в России использует платформу для разработки и внедрения решений в области интернета вещей (IoT), умного дома и промышленной автоматизации. Простота и доступность платформы позволяют быстро разрабатывать и адаптировать решения под конкретные нужды бизнеса.

Среди энтузиастов и хобби-изобретателей Arduino стал популярным инструментом для реализации различных проектов. Российские любители электроники создают с помощью Arduino роботов, автоматизированные системы управления, световые инсталляции и другие интересные устройства.

Сообщества и форумы, посвященные Arduino, активно обсуждают проекты и делятся опытом.

В России также проводятся различные мероприятия, посвященные Arduino и разработке электроники. Фестивали, конкурсы и мастер-классы помогают популяризировать платформу и объединять людей, увлеченных технологическими инновациями. Эти мероприятия способствуют обмену знаниями и вдохновляют на создание новых проектов.

Таким образом, Arduino нашел широкое применение в России и продолжает развиваться. Платформа стала важным инструментом для обучения, исследований, промышленного применения и хобби-проектов, способствуя развитию технического творчества и инноваций в стране.

## **2 Техническое задание**

### **2.1 Основание для разработки**

Основанием для разработки является задание на выпускную квалификационную работу бакалавра «Программная реализация управления мультиробота на базе Arduino».

### **2.2 Цель и назначение разработки**

Основной задачей выпускной квалификационной работы является разработка кода работы мультиробота, способного работать в различных режимах, которые предоставляют пользователю спектр возможностей, будь то нахождение объектов, измерения расстояния до цели, проведение досуга, измерение температуры и влажности окружающей среды и определение их значений с последующим выводом информации, также данный может являться базой для модификации, создания пользовательской усовершенствованной версии. Предполагается что данным устройством будут пользоваться как профессионалы в работе с Arduino, так и обычные пользователи заинтересованные в изучении робототехники.

Задачами данной разработки являются:

- реализация кода переключения между режимами работы мультиробота;
- реализация вывода информации на lcd монитор подключенный к плате arduino;
- реализация подключения ультразвуковых датчиков с последующим расчетом расстояния до объекта перед роботом;
- реализация подключения и вывода информации об окружающей среде;
- реализация режима для проведения досуга.

## **2.3 Функциональные требования к мультироботу**

На основании анализа предметной области разрабатываемый мультиробот должен иметь следующие функции:

- расчет дистанции до цели на основе данных с датчиков;
- возможность переключения между разными режимами работы устройства;
- переход устройства в энергосберегающий режим, основываясь на уровне освещенности;
- поддержка режима приветствия пользователя;
- поддержка воспроизведения заранее определенного звукового сигнала;
- определение температуры окружающей среды.

Система будет состоять из следующих физических модулей:

- Arduino-совместимый фотодиод – это полупроводниковый прибор, который уменьшает свое электрическое сопротивление при воздействии света;
- LCD монитор вывода – отображает изображение с помощью жидкокристаллических кристаллов, которые реагируют на электрические сигналы;
- датчик температуры Dallas – это цифровой датчик температуры, который выпускает компания Maxim Integrated (ранее Dallas Semiconductor);
- ультразвуковой датчик – это электронное устройство, которое использует ультразвуковые волны для измерения расстояния до объекта и обнаружения его присутствия;
- сервопривод (двигатель) – это устройство, используемое для управления положением механизма или устройства посредством обратной связи;
- светодиоды – это полупроводниковые приборы, которые преобразуют электрическую энергию в свет;
- активный сигнализатор (Buzzer) – это электромеханическое устройство, которое используется для издания звуковых сигналов.

На рисунке 2.1 приведена концептуальная модель мультиробота.

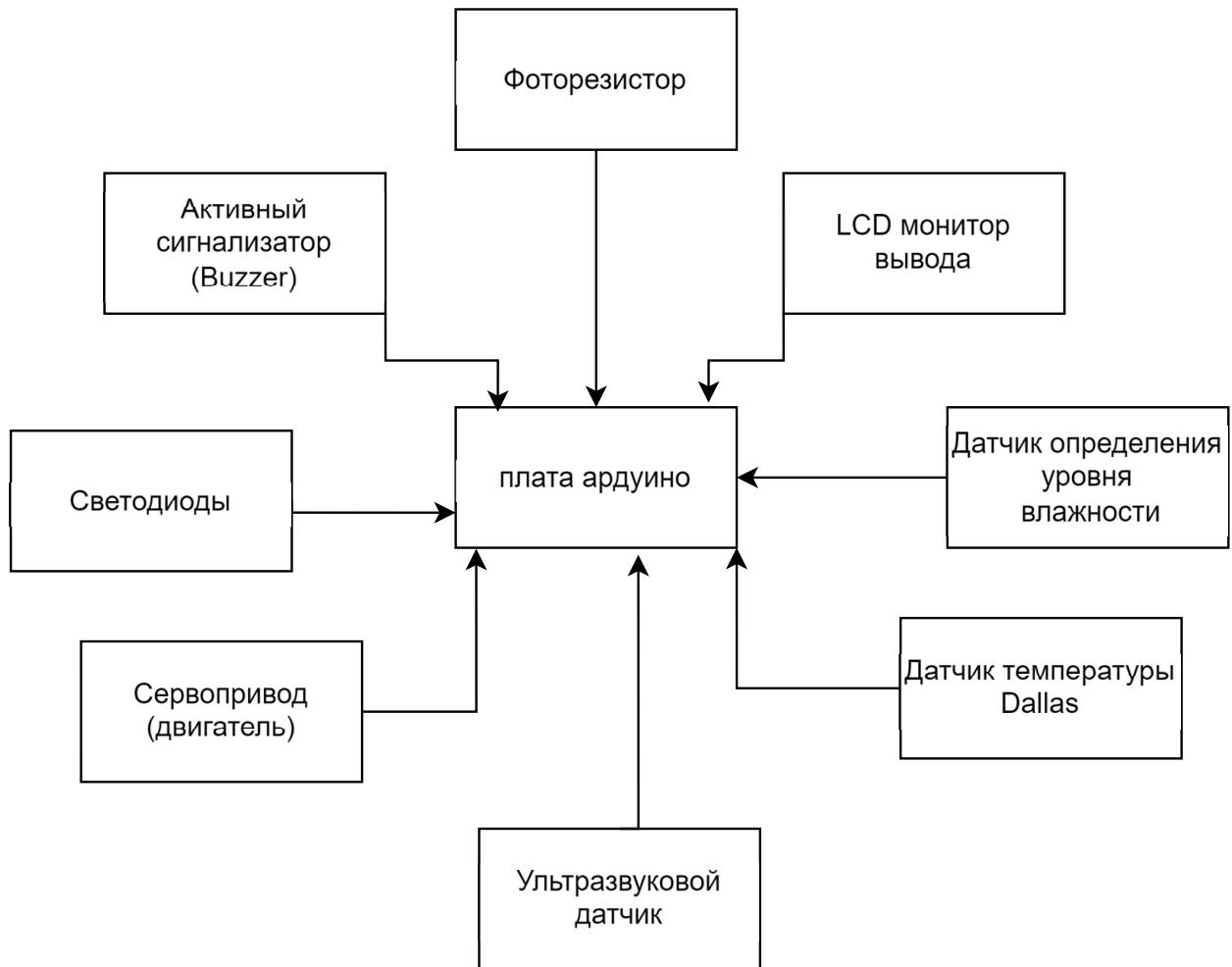


Рисунок 2.1 – Концептуальная модель мультиробота

### 2.3.1 Вариант использования: использование после запуска

После запуска мультиробот выводит на lcd монитор сообщение с просьбой выбрать режим, после чего при нажатии на кнопку будет меняться значение текущего режима, которое будет зависеть от счетчика нажатий, пользователь может выбрать любой режим от 3 до 6(для пользователя они представлены, в виде счетчика от 1 до 4), если пользователь нажимает на кнопку более 4 раз, то значение счетчика сбрасывается до 1.

### 2.3.2 Вариант использования: пользователь установил значение счетчика равное 1

После выхода из режима ожидания мультиробот переходит в режим вывода информации о температуре, термометр считывает показания из окружающей среды, выводит на lcd монитор текущее значение температуры и

если она ниже 20 градусов то на lcd монитор выводится сообщение "холодно" также в этот момент загорается 1 из 3 светодиодов(зеленый). Если температура между 20 и 27 градусами, то включается также желтый светодиод, а на lcd монитор выводится сообщение (нормально). Если температура выше 27 градусов, то в дополнение к первым 2 светодиодам загорится и последний(красный), а на монитор будет выведено сообщение (тепло).

### **2.3.3 Вариант использования: пользователь установил значение счетчика равное 2**

После выхода из режима ожидания мультиробот переходит в режим игры "экран судьбы суть игры заключается в том что пользователь получает случайное число, в зависимости от значения которого на экран выводится соответствующее ему сообщение.

### **2.3.4 Вариант использования: пользователь установил значение счетчика равное 3**

После выхода из режима ожидания мультиробот переходит в режим поиска объекта перед собой, с помощью ультразвукового датчика устройство определяет своё местоположение относительно объекта, путем приема отражающихся звуковых волн, которые сам же издает, после нахождения объекта сообщение о том что (предмет найден) будет выведено на lcd монитор, также мультиробот выведет информацию о расстоянии до найденного объекта, которое вычисляет используя время отражения сигнала, также выводится угол между объектом и мультироботом.

### **2.3.5 Вариант использования: пользователь установил значение счетчика равное 4**

После выхода из режима ожидания мультиробот переходит в режим игры на скорость, суть режима заключается в том что робот выведет на монитор сообщение (приготовтесь), после чего последует звуковой сигнал оповещающий о начале игры, также будет выведено количество очков, в этот момент

пользователь должен быстро проводить каким-либо объектом перед роботом, каждый раз когда действие будет выполнено количество очков будет увеличено на 1, после достижения значения в 5 очков робот сигнализирует о конце игры с помощью звукового сигнала, после чего выведет время, за которое пользователь завершил игру.

### **2.3.6 Вариант использования: пользователь завершил выполнение 1 из режимов**

После выхода из выбираемых пользователем режимов мультиробот считывает уровень текущей освещенности с помощью фоторезистора, если он низок, то устройство переходит в режим сна, пользователь может увидеть это на lcd мониторе, если будет нажата кнопка, то робот перейдет в режим ожидания новой команды.

### **2.3.7 Вариант использования: мультиробот находится в ночном режиме, но уровень яркости растет**

Если пользователь не пробуждает мультиробота из режима сна вручную, то в тот момент когда уровень яркости поднимется выше определенного значения, робот самостоятельно перейдет в режим приветствия, в данном режиме мультиробот издаст с помощью буззера мелодию, а также выведет пользователю на lcd монитор сообщения Привет!, Встаю!. После выполнения осуществляется переход в режим ожидания.

## **2.4 Нефункциональные требования к мультироботу**

### **2.4.1 Требования к программному обеспечению**

Для реализации выгрузки кода в устройство необходимо использование языка C++ на основе интерпретатора Arduino IDE.

## **2.4.2 Требования к аппаратному обеспечению**

Необходима операционная система, поддерживающая среду разработки Arduino IDE. Требуется физический экземпляр мультиробота, созданного на основе платы Arduino Mega

. Робот должен в себя включать физические модули, совместимые с платформой Arduino, приведенные в концептуальной модели.

. Для выгрузки кода в устройство нужно использовать Bootloader или программатор.

## **2.4.3 Требования к оформлению документации**

Разработка программной документации и программного изделия должна производиться согласно ГОСТ 19.102-77 и ГОСТ 34.601-90. Единая система программной документации.

### **3 Технический проект**

#### **3.1 Общая характеристика организации решения задачи**

Необходимо спроектировать и разработать код, который должен способствовать переключению режимов у мультиробота.

Режимы представляют собой набор взаимосвязанных последовательностей действий, которые позволяют выводить информацию с различных модулей Arduino содержащие данные об окружающей среде и выполняющие различный спектр действий. Код должен быть загружен в соответствующую модель мультиробота на базе arduino, код представляет собой файл с расширением .ino, может быть загружен через любой порт, с соответствующим разъёмом.

#### **3.2 Обоснование выбора технологии проектирования**

На сегодняшний день Arduino является популярным выбором среди энтузиастов и хоббистов для создания уникальных устройств и решений. Также используется для создания систем автоматизации, управления освещением, климатом и т.д. Arduino – это мощная и гибкая платформа, подходящая как для образовательных целей, так и для профессионального использования за счет своей модульности, которая позволяет создать устройство, которое можно будет оперативно модифицировать под требующиеся цели.

##### **3.2.1 Описание используемых технологий и языков программирования**

В процессе разработки кода используются C++, в контексте разработки на Arduino имеет свои особенности, которые влияют на его достоинства и недостатки.

##### **3.2.2 Датчик DHT11 temperature and humidity sensor**

DHT11 — это широко используемый цифровой датчик, предназначенный для измерения температуры и влажности.

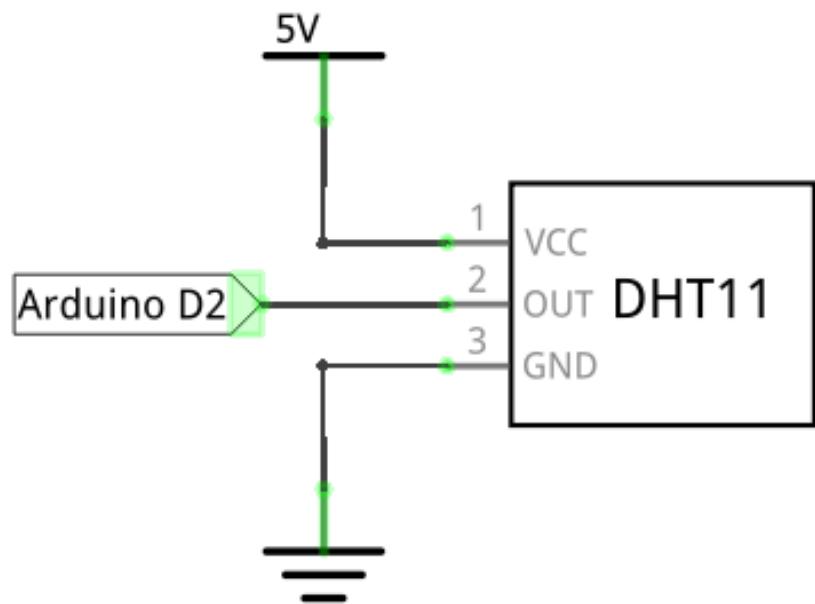


Рисунок 3.1 – принципиальная схема подключения dht11

Основные характеристики:

Температурный диапазон: 0-50°C

Точность измерения температуры: ±2°C

Диапазон влажности: 20-90

Точность измерения влажности: ±5

Напряжение питания: 3.5-5.5V

Интерфейс: Однопроводный цифровой интерфейс.

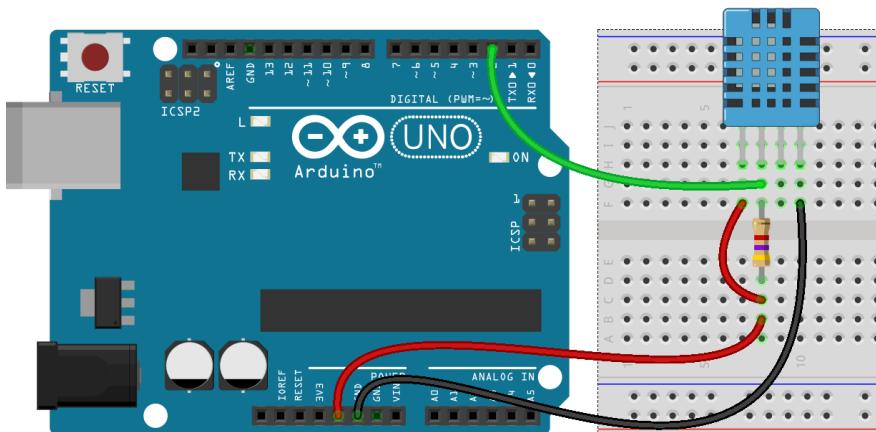


Рисунок 3.2 – макетная схема подключения dht11

Датчик DHT11 состоит из емкостного датчика влажности и термистора для измерения температуры. Данный датчик находит применение в различных устройствах и системах:

Метеостанции: Измерение температуры и влажности для домашних метеостанций.

Умные дома: Управление климатическими системами.

Агрономия: Контроль влажности и температуры в теплицах.

В случае необходимости, данный датчик можно заменить на приведенные ниже аналоги:

DHT22: Более точный аналог DHT11, измеряет температуру в диапазоне от -40 до 80°C и влажность от 0 до 100

AM2302: Это тот же DHT22, но в другом корпусе.

BME280: Датчик, измеряющий не только температуру и влажность, но и давление.

### 3.2.3 **buzzer**

Buzzer (зуммер) — это электронное устройство, предназначенное для генерации звуковых сигналов. В контексте Arduino, зуммеры часто используются для создания простых звуковых сигналов, уведомлений или мелодий в проектах. Они могут быть двух типов: активные и пассивные.

Основные виды зуммеров Активные зуммеры:

Принцип работы: Содержат встроенный генератор сигнала, который автоматически создает звук при подаче напряжения. Преимущества: Легкость в использовании, требуется только подключить питание. Недостатки: Меньший контроль над звуком и частотой. Пассивные зуммеры:

Принцип работы: Требуют внешнего источника сигнала, например, от Arduino, для генерации звука. Преимущества: Большой контроль над частотой и типом звука. Недостатки: Требуется дополнительное программирование и подключение.

В данном устройстве используется пассивный зуммер.

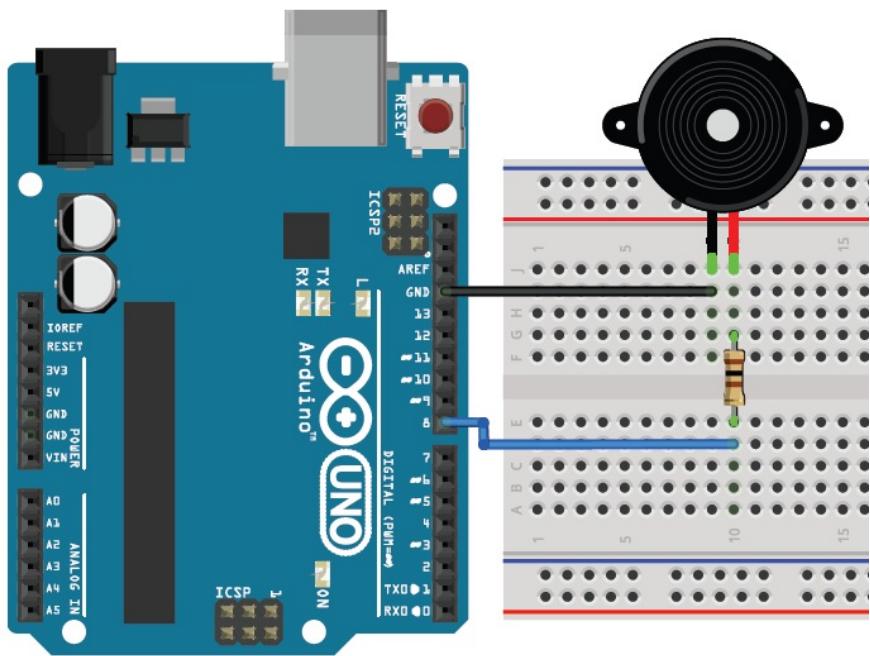


Рисунок 3.3

Зуммеры основаны на пьезоэлектрическом эффекте, где пьезоэлектрический кристалл выбириует при подаче на него переменного напряжения, создавая звуковые волны.

### 3.2.4 sharp 2Y0A02

Sharp 2Y0A02 — это инфракрасный датчик расстояния, используемый для измерения расстояния до объектов в диапазоне от 20 до 150 см. Этот датчик широко применяется в робототехнике, системах безопасности и других областях, требующих точного измерения расстояния.

Основные характеристики

Диапазон измерения: 20-150 см.

Выходной сигнал: Аналоговый (напряжение пропорционально расстоянию до объекта).

Напряжение питания: 4.5-5.5 В.

Ток потребления: 33 мА (типичное значение).

Размеры: Компактный размер, что позволяет легко интегрировать датчик в различные устройства.

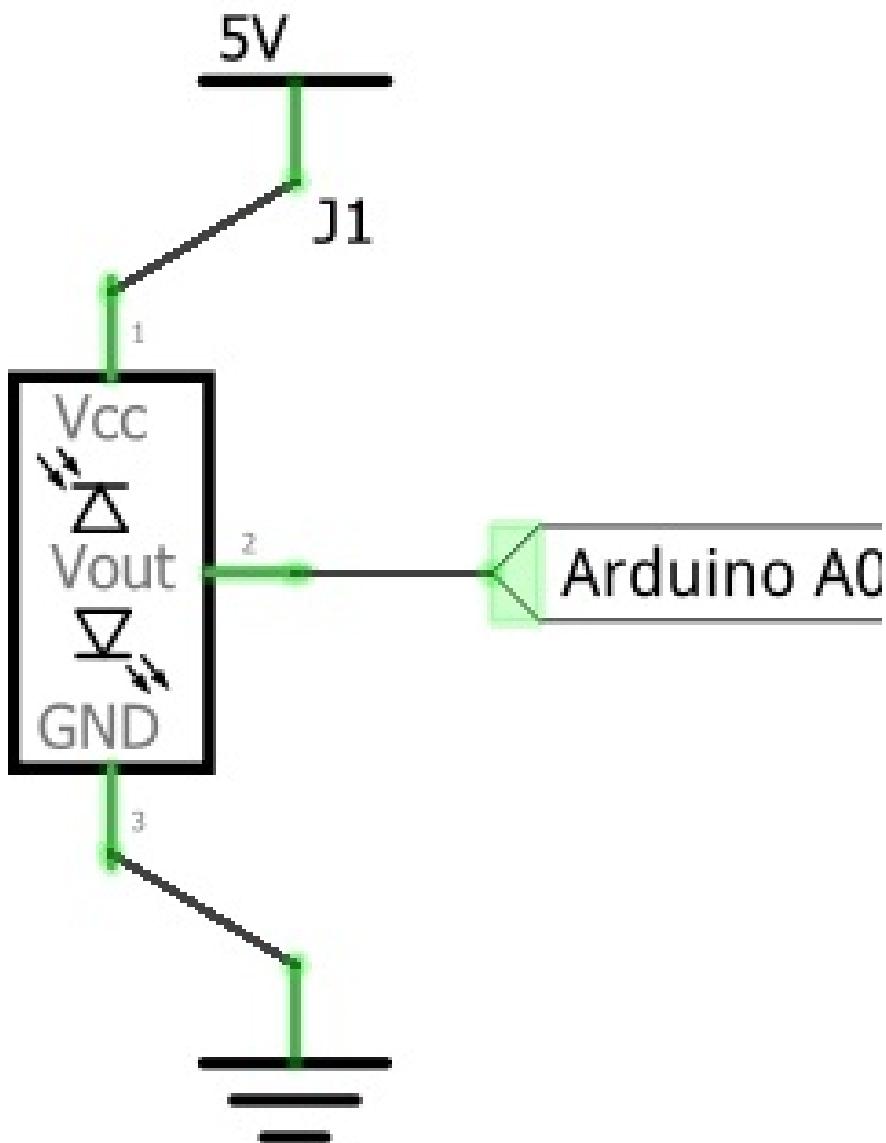


Рисунок 3.4

Датчик Sharp 2Y0A02 использует инфракрасный луч для измерения расстояния. Он состоит из инфракрасного светодиода (излучателя) и позиции чувствительного фототранзистора (приемника). Излучаемый инфракрасный свет отражается от объекта и попадает на приемник, который определяет расстояние по углу отражения света.

#### Преимущества

**Точность:** Высокая точность измерений в указанном диапазоне.

**Легкость использования:** Простое подключение и использование с микроконтроллерами, такими как Arduino.

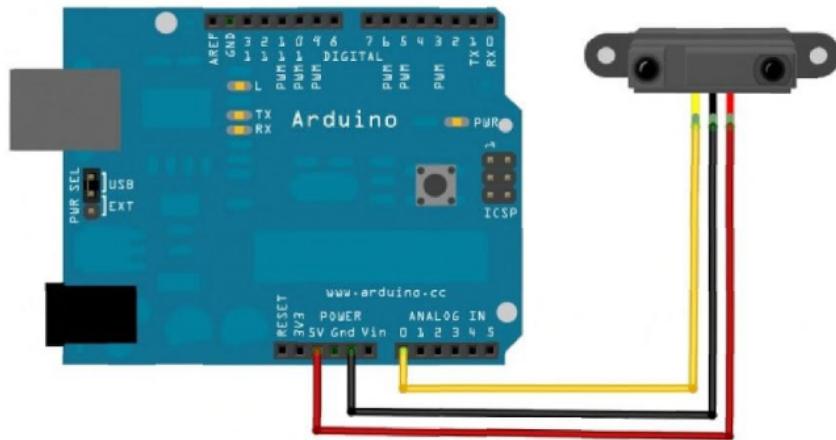


Рисунок 3.5 – макетная схема подключения

**Компактность:** Малый размер позволяет интегрировать датчик в компактные устройства.

#### Ограничения

**Чувствительность к свету:** Работает на основе инфракрасного света, поэтому может быть чувствителен к внешним источникам инфракрасного излучения.

**Углы обзора:** Точность измерений может снижаться при измерении под большими углами.

Для замены или альтернативы датчику расстояния Sharp 2Y0A02 существует несколько аналогичных устройств, которые также могут использоваться для измерения расстояния с использованием инфракрасной технологии или других методов. Вот некоторые из них:

#### Sharp GP2Y0A21YK0F:

Диапазон измерения: 10-80 см

Выходной сигнал: Аналоговый

Этот датчик предлагает более короткий диапазон измерения по сравнению с GP2Y0A02, но обеспечивает хорошую точность и быстрое время отклика.

#### Sharp GP2Y0A710K0F:

Диапазон измерения: 100-550 см

Выходной сигнал: Аналоговый

Подходит для измерения больших расстояний, где требуется более широкий диапазон, чем у GP2Y0A02.

VL53L0X от STMicroelectronics:

Диапазон измерения: До 200 см

Выходной сигнал: I2C интерфейс

Это лазерный датчик расстояния с высокой точностью и малым энергопотреблением. Обеспечивает цифровой выход данных, что может быть полезно для некоторых применений.

При выборе аналога важно учитывать следующие факторы:

Диапазон измерения: Убедитесь, что диапазон измерения соответствует требованиям вашего проекта.

Тип выходного сигнала: Аналоговый или цифровой выход в зависимости от совместимости с вашим микроконтроллером. Стоимость: Цена может варьироваться в зависимости от характеристик и возможностей датчика. Простота интеграции: Убедитесь, что датчик легко интегрируется в вашу существующую систему. Эти аналоги предоставляют разнообразные варианты для различных приложений, требующих измерения расстояния, и могут быть выбраны в зависимости от конкретных требований

### **3.2.5 LCD монитор MT 16S2H 2YLG PB**

В данном устройстве применяется Жидкокристаллический модуль MT-16S2H.

Данный модуль состоит из БИС контроллера управления и ЖК панели. Модуль позволяет отображать 2 строки по 16 символов в каждой. Символы отображаются в матрице 5x8 точек. Между символами имеются интервалы шириной в одну отображаемую точку.

Каждому отображаемому на ЖКИ символу соответствует его код в ячейке ОЗУ модуля.

Модуль позволяет:

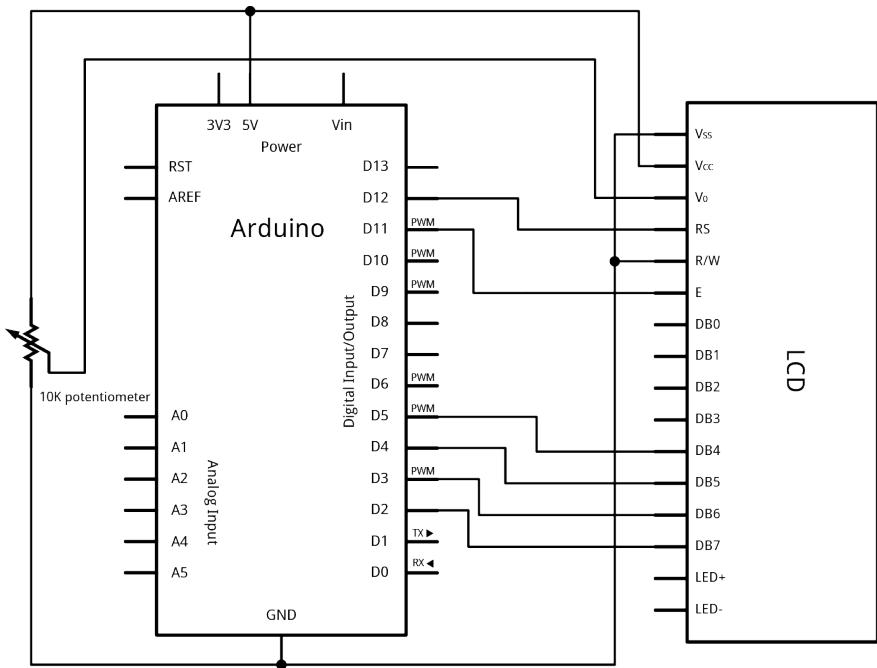


Рисунок 3.6 – принципиальная схема подключения lcd монитора

- модуль имеет программно переключаемые две страницы встроенного знакогенератора (алфавиты: русский, украинский, белорусский, казахский и английский; см. табл. 5–0 и 5–1).
- работать как по 8-ми, так и по 4-х битной шине данных (задается при инициализации);
- принимать команды с шины данных (перечень команд приведен в таблице 4);
- записывать данные в ОЗУ с шины данных;
- читать данные из ОЗУ на шину данных;
- читать статус состояния на шину данных (см. табл. 4);
- запоминать до 8 изображений символов, задаваемых пользователем;
- выводить мигающий (или не мигающий) курсор двух типов;
- управлять контрастностью и подсветкой;

В случае необходимости данный дисплей можно заменить на аналогичный, есть большое количество возможных вариантов, ниже представлены некоторые из них:

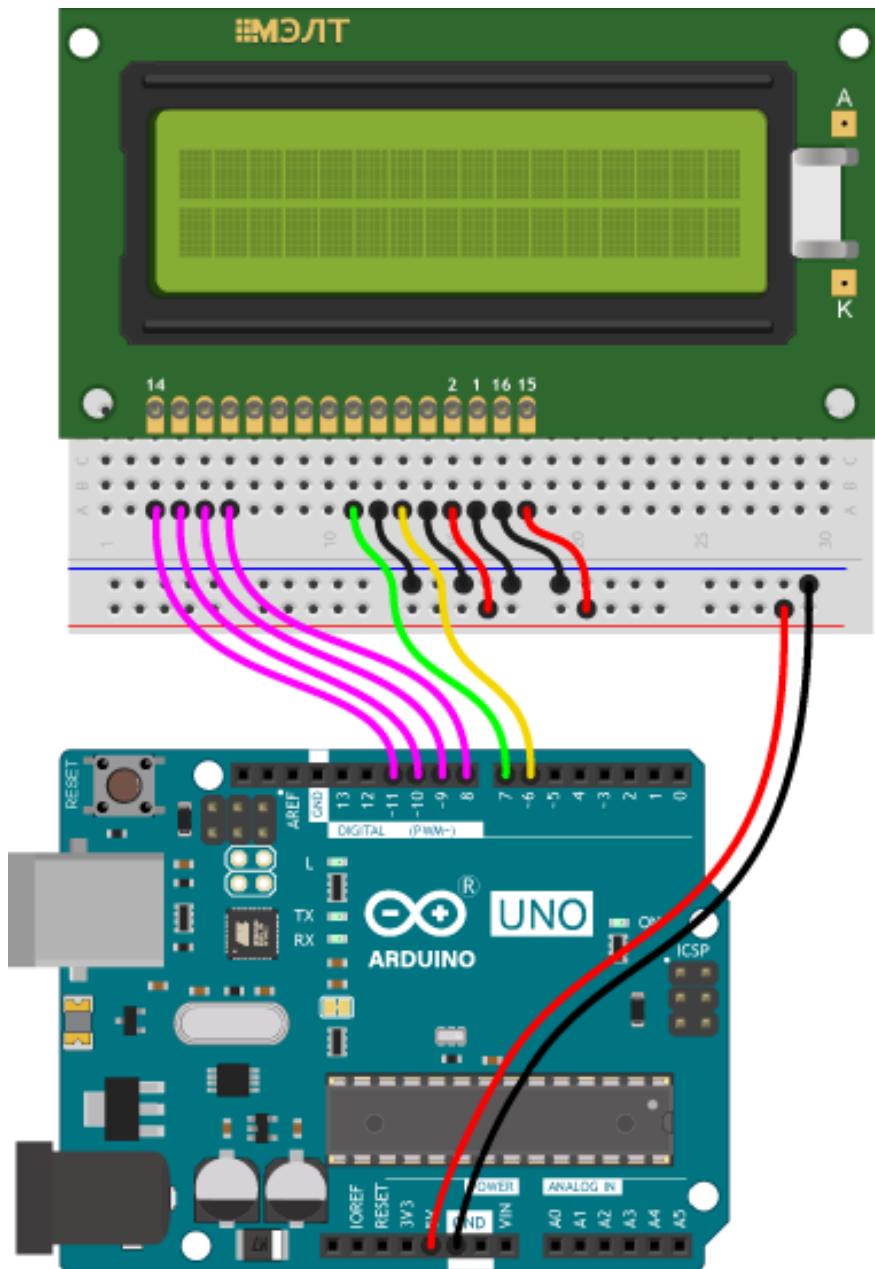


Рисунок 3.7

BCB1602-03 от Blaze: ЖК дисплей 16x2 с англо-русскими символами и подсветкой. Он предлагает аналогичные характеристики .

SDCB1602GYZ-R10 от Sinda: Еще один 16x2 символьный дисплей с англо-русскими символами и подсветкой.

WH1602L: Это 16x2 символьный ЖК дисплей, который также является популярной альтернативой и может быть найден на 3D ContentCentral.

NHD-C0216CIZ-FSW-FBW-3V3: Подсвечиваемый 16x2 дисплей, который можно найти на 3D ContentCentral, является хорошей альтернативой.

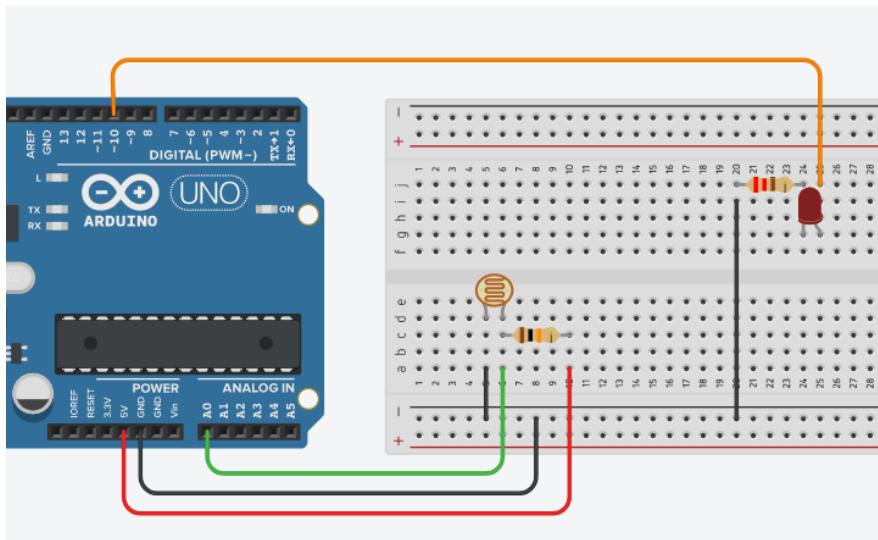


Рисунок 3.8 – макетная схема подключения фоторезистора

### 3.2.6 фоторезистор

Фоторезистор основан на принципе, согласно которому проводимость полупроводникового материала увеличивается при воздействии света. Чем больше света падает на фоторезистор, тем меньше его сопротивление.

Применение фоторезистора: Автоматическое освещение: Включение

или выключение света в зависимости от уровня освещенности. Умные шторы: Открытие или закрытие штор в зависимости от солнечного света. Системы безопасности: Обнаружение движения или изменение освещенности. Умный сад: Измерение уровня освещенности для автоматического управления поливом и освещением растений. Детектор освещенности: Создание устройств, реагирующих на изменение освещенности в помещении или на улице. Фоторезистор является эффективным компонентом для проектов

позволяющим измерять уровень освещенности и создавать интерактивные устройства. Он может стать неотъемлемым компонентом для реализации множества проектов, от автоматического освещения до умных домашних систем.

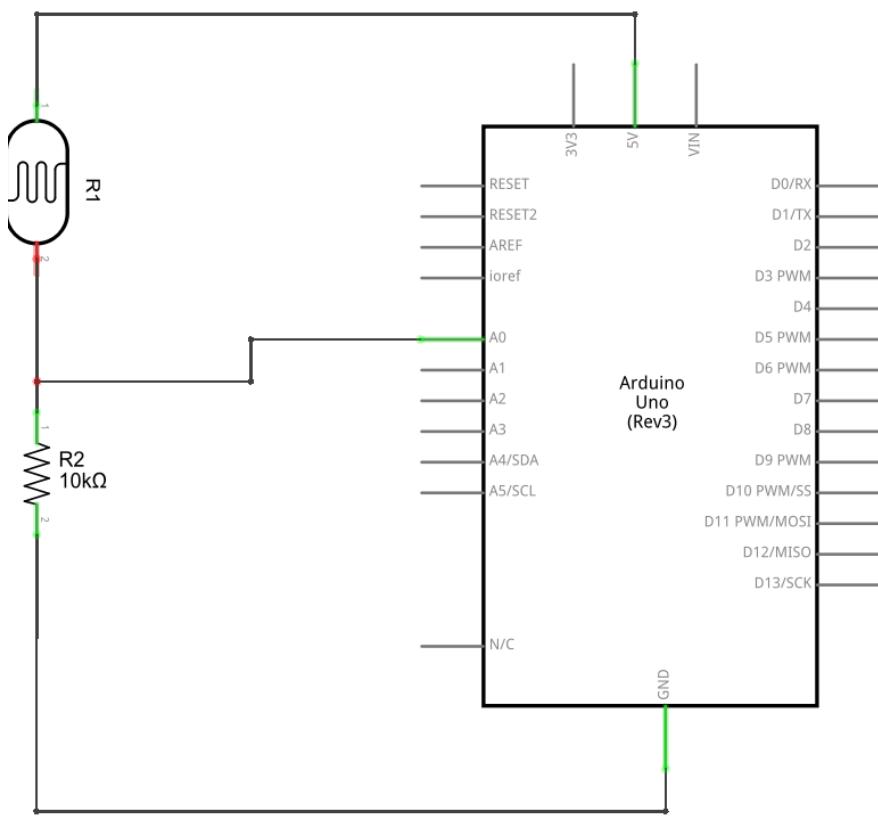


Рисунок 3.9 – принципиальная схема подключения фотодиода

### 3.2.7 Ультразвуковой датчик HC-SR04

HC-SR04 — это ультразвуковой датчик расстояния, широко используемый в различных проектах для измерения расстояния до объектов. Этот датчик работает по принципу эхолокации, используя ультразвуковые волны для определения расстояния до препятствий.

**Основные характеристики**

Диапазон измерения: 2 см - 400 см

Точность измерения: до 3 мм

Рабочее напряжение: 5V DC

Ток потребления: 15 mA

Угол обзора: около 15 градусов

Интерфейс: цифровой, управление через пины Trigger и Echo

HC-SR04 использует принцип эхолокации для измерения расстояния.

Вот как это работает:

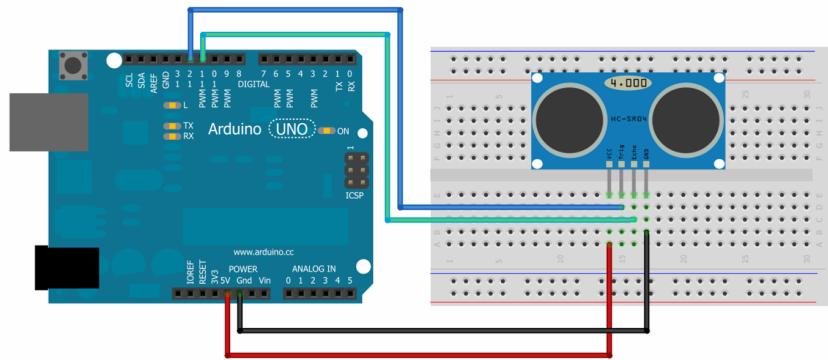


Рисунок 3.10

**Передача сигнала:** Датчик посылает ультразвуковой импульс (8 циклов 40 кГц) через передающий пин (Trigger).

**Отражение сигнала:** Импульс отражается от объекта и возвращается обратно к датчику.

**Прием сигнала:** Возвращенный импульс принимается приемным пином (Echo).

**Вычисление расстояния:** Датчик измеряет время, прошедшее между отправкой и получением импульса, и на основе этого вычисляет расстояние.

Данный датчик применяется в различных сферах деятельности:

**Робототехника:** Используется для избегания препятствий, измерения расстояний и навигации.

**Системы безопасности:** Для обнаружения движущихся объектов и определения присутствия.

**Автомобильные системы:** Парковочные ассистенты для определения расстояния до препятствий.

Ультразвуковой датчик HC-SR04 является эффективным решением для измерения расстояния в различных проектах, тем не менее, существуют и другие датчики, которые могут выполнять аналогичные функции с разными характеристиками и дополнительными возможностями.

Parallax PING))):

Диапазон измерения: 2 см - 300 см

**Особенности:** Более высокая точность, использование одного пина для эхо и триггера.

Датчик от Parallax использует один и тот же pin для отправки и получения сигнала, что упрощает подключение и сокращает количество используемых пинов на микроконтроллере.

#### MaxBotix MB1010 LV-MaxSonar-EZ1:

Диапазон измерения: 15 см - 645 см Особенности: Высокая точность, три варианта выхода (аналоговый, последовательный и ШИМ).

Этот датчик отличается высокой чувствительностью и надежностью. Он предоставляет три различных метода вывода данных, что делает его универсальным для различных применений.

#### Seeed Studio Grove Ultrasonic Ranger:

Диапазон измерения: 3 см - 400 см

Особенности: Совместимость с системой Grove, простота подключения.

Датчик от Seeed Studio легко интегрируется с системой Grove, что упрощает подключение и использование в проектах с использованием Grove-коннекторов.

Выбор аналогичного датчика для HC-SR04 зависит от конкретных требований.

### 3.2.8 Сервопривод

Сервопривод (или сервомотор) — это устройство, которое используется для точного контроля углового положения, скорости и ускорения. Сервоприводы широко применяются в робототехнике, моделировании, автоматике и других областях, где требуется точное управление движением.

Сервопривод получает управляющий сигнал (обычно PWM — широтно-импульсная модуляция), который определяет, на какой угол должен быть повернут вал. Контроллер сервопривода сравнивает текущий угол (определенный потенциометром) с заданным углом и корректирует положение вала, поворачивая его в нужную сторону

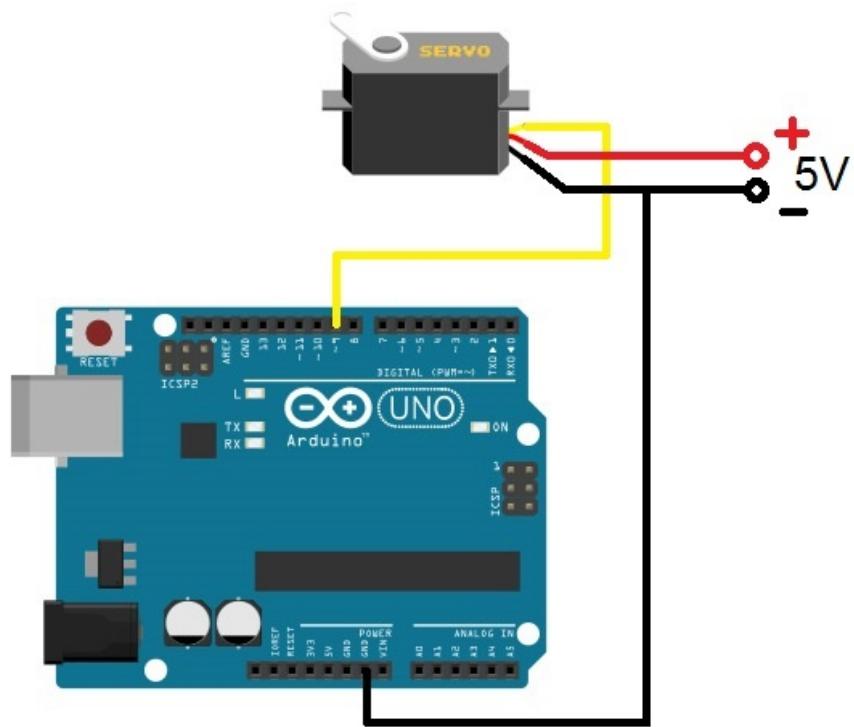


Рисунок 3.11 – макетная схема подключения сервопривода

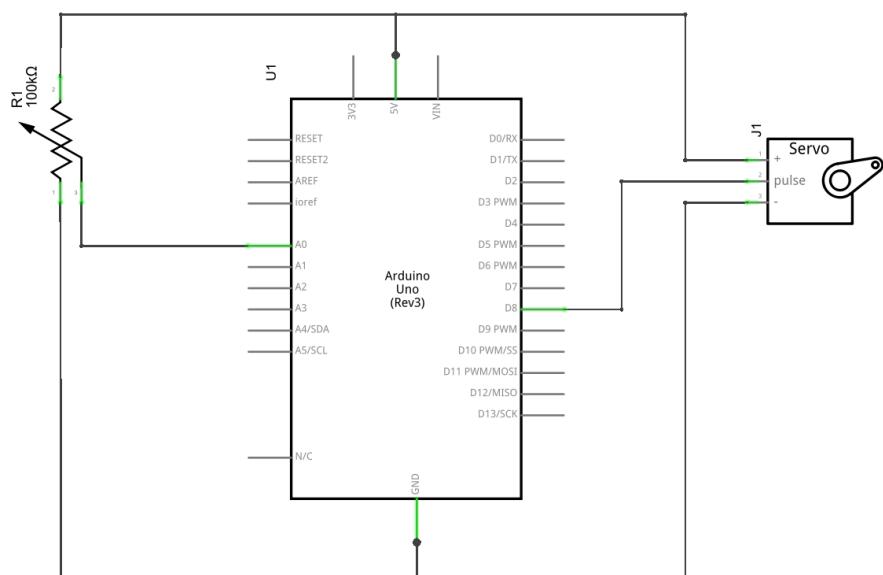


Рисунок 3.12

### **3.2.9 Язык программирования Arduino (C++)**

**Arduino (C++):** Основной язык для разработки скетчей Arduino. Этот язык представляет собой упрощённый вариант C++, адаптированный для работы с микроконтроллерами. В Arduino IDE разработчики пишут программы, называемые ”скетчами” которые затем загружаются на плату Arduino для выполнения. Синтаксис языка C++ позволяет эффективно работать с аппаратными ресурсами платы и использовать многочисленные библиотеки для расширения функциональности. К примеру, библиотеки для работы с датчиками, модулями связи и управления устройствами существенно упрощают разработку проектов.

Разработка на языке C++ предоставляет возможность использовать объектно-ориентированное программирование, что способствует модульности и повторному использованию кода. Структура языка C++ позволяет создавать сложные программы с чётким разделением функций и данных, что облегчает управление проектами и улучшает читаемость кода. Кроме того, язык C++ поддерживает низкоуровневое программирование, что позволяет разработчикам управлять аппаратными ресурсами платы на более детализированном уровне.

Arduino IDE предоставляет удобный способ работы с C++ благодаря встроенным функциям и библиотекам. В среде разработки доступны многочисленные библиотеки, такие как Wire для работы с I2C, SPI для последовательного периферийного интерфейса, Servo для управления сервоприводами и многие другие. Эти библиотеки упрощают работу с различными периферийными устройствами и расширяют возможности микроконтроллера.

В процессе разработки скетчей на C++ в Arduino IDE используется двухфазная компиляция, что позволяет оптимизировать код перед загрузкой на плату. Это обеспечивает более эффективное использование памяти и ресурсов микроконтроллера. Поддержка макросов и препроцессора в C++ позволяет создавать гибкие и адаптируемые программы, что особенно полезно при работе с различными конфигурациями оборудования.

Язык C++ в контексте Arduino включает использование типичных конструкций, таких как циклы, условия и функции, а также специфических функций, таких как `digitalRead()`, `digitalWrite()` и `analogRead()`, для взаимодействия с пинами микроконтроллера. Это делает язык C++ в Arduino интуитивно понятным для тех, кто уже знаком с базовыми концепциями программирования, и позволяет быстро освоить разработку электронных проектов.

Таким образом, использование языка C++ для разработки на платформе Arduino предоставляет мощные инструменты для создания разнообразных проектов, от простых до сложных, обеспечивая при этом высокий уровень контроля над аппаратной частью.

### **3.2.9.1 Достоинства языка C++**

Высокая производительность:

C++ позволяет работать на низком уровне, что особенно важно для работы с микроконтроллерами. Это даёт возможность оптимизировать код для лучшего использования ресурсов. Обширная библиотека:

Существует множество готовых библиотек для работы с различными датчиками, модулями и устройствами, что ускоряет процесс разработки. Мощные возможности ООП:

Использование объектно-ориентированного программирования позволяет создавать модульный и легко расширяемый код, что упрощает управление крупными проектами. Прямой доступ к железу:

C++ позволяет напрямую взаимодействовать с аппаратной частью Arduino, что даёт полный контроль над микроконтроллером. Сообщество и документация:

Большое сообщество разработчиков и обширная документация упрощают решение возникающих проблем и обучение.

### **3.2.9.2 Недостатки языка C++**

Отсутствие автоматического управления памятью:

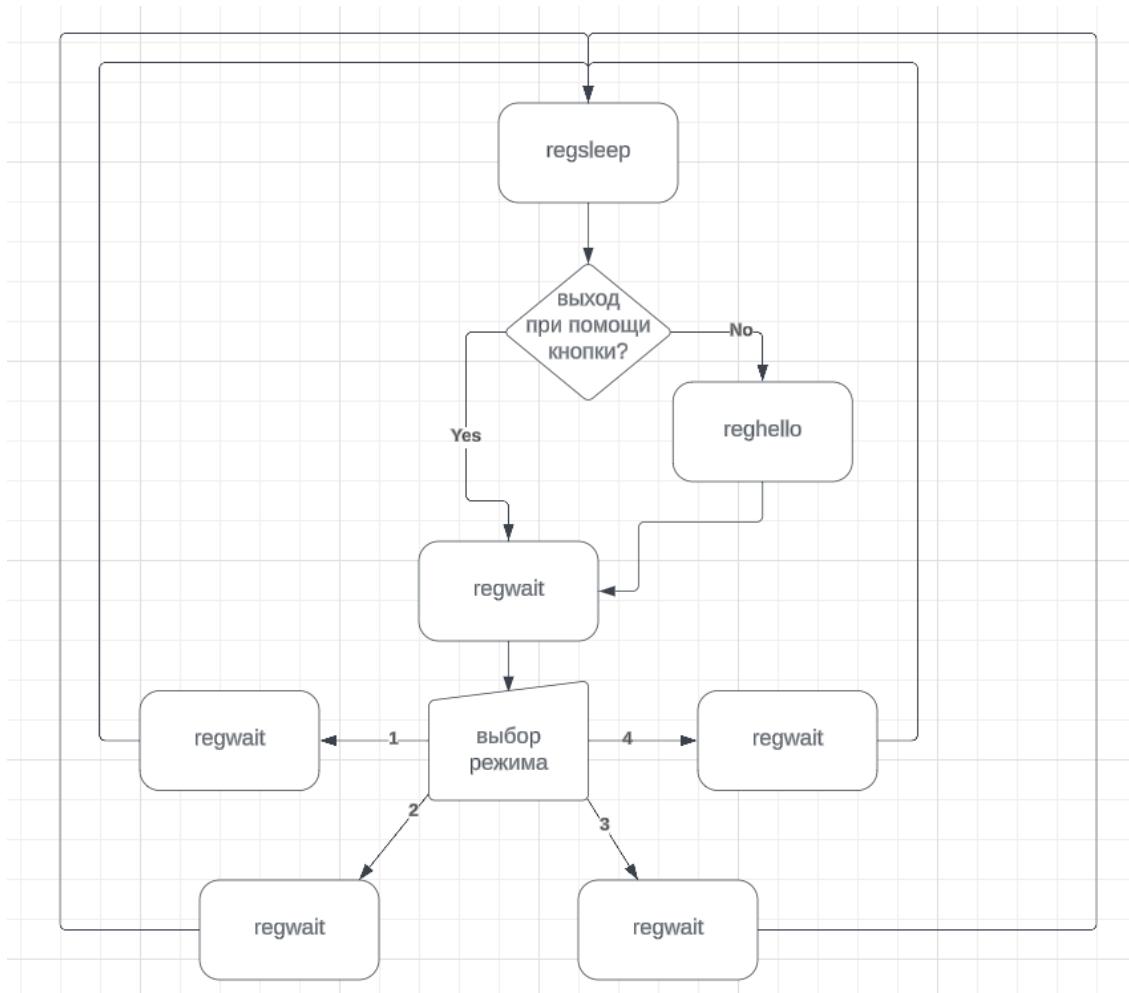


Рисунок 3.13 – Схема работы кода

Необходимо вручную управлять памятью, что может привести к утечкам памяти и другим ошибкам, особенно на микроконтроллерах с ограниченными ресурсами. Размер кода:

Из-за особенностей C++ с точки зрения управления объектами и их создания, итоговый код может занимать больше памяти по сравнению с языками более высокого уровня. Проблемы с совместимостью:

Некоторые функции C++ могут быть недоступны или работать иначе в контексте Arduino, что требует дополнительной проверки и адаптации кода. Ограничения производительности:

Несмотря на высокую производительность, некоторые возможности языка могут быть ограничены производительностью микроконтроллера, особенно при выполнении сложных вычислений или работы с большими объёмами данных.

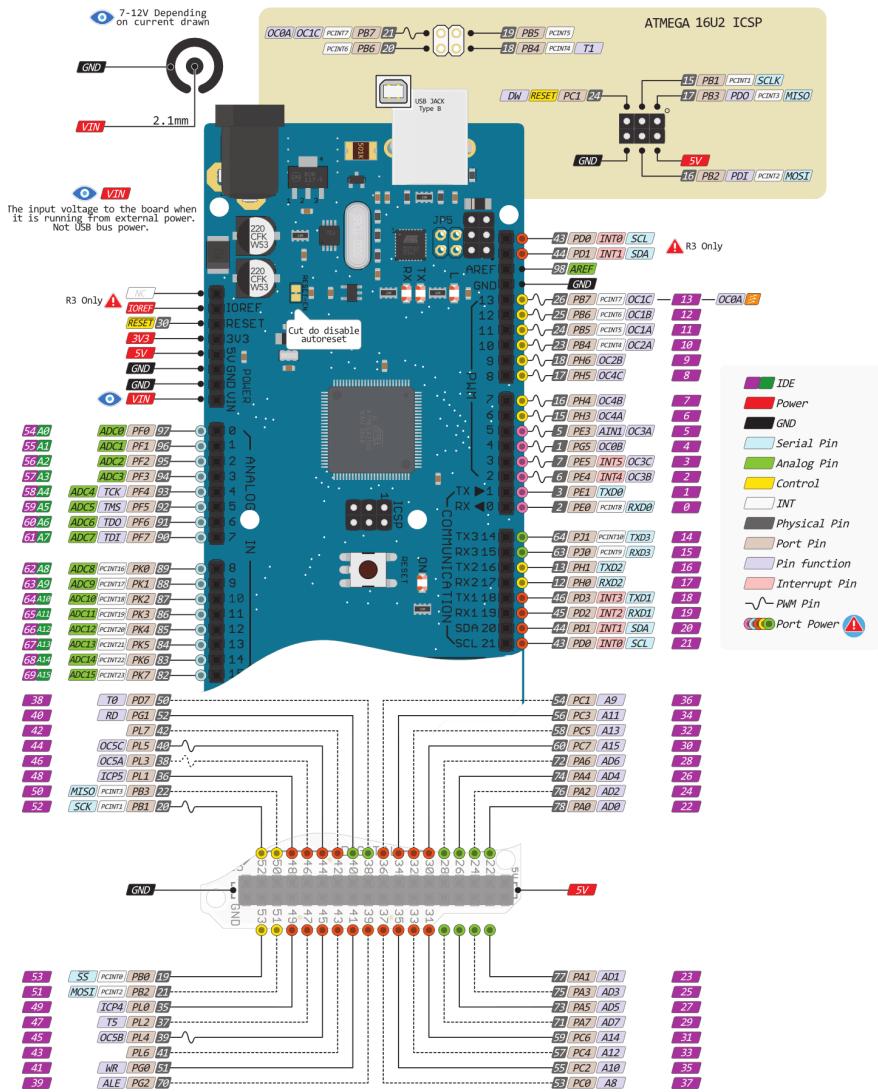


Рисунок 3.14 – Структура портов Arduino Mega

На рисунке 3.1 представлена упрощенная схема работы кода данными между сценариями компонента при вызове компонента на странице сайта.

### 3.2.10 Система ввода-вывода данных Arduino

Arduino позволяет выводить данные различных типов с помощью встроенных в плату портов ввода-вывода.

Arduino Mega включает различные типы выходов и входов, которые могут быть использованы для подключения к различным датчикам, дополнительным механизмам и другим компонентам. В таблицах ниже описаны аналоговые и цифровые выходы платы Arduino Mega.

На рисунках 3.1,3.2 изображена структура портов Arduino Mega.

Таблица 3.1 – Аналоговые выходы

Pin	Функция	Тип	Описание
1	2	3	4
1	VIN	Питание	Входное напряжение (7-12В)
2	5V	Питание	Стабилизированное напряжение 5В
3	3.3V	Питание	Стабилизированное напряжение 3.3В (максимальный ток 50 мА)
4	GND	Питание	Земля
5	IOREF	Опорное напряжение	Опорное напряжение для работы цифровых выводов
6	RESET	Сброс	Низкий уровень сбрасывает микроконтроллер
7	AREF	Аналоговая опора	Аналоговое опорное напряжение для аналоговых входов
8	A0	Аналоговый вход/ADC	Аналоговый вход 0
9	A1	Аналоговый вход/ADC	Аналоговый вход 1
10	A2	Аналоговый вход/ADC	Аналоговый вход 2

Продолжение таблицы 3.1

1	2	3	4
11	A3	Аналоговый вход/ADC	Аналоговый вход 3
12	A4	Аналоговый вход/ADC	Аналоговый вход 4
13	A5	Аналоговый вход/ADC	Аналоговый вход 5
14	A6	Аналоговый вход/ADC	Аналоговый вход 6
15	A7	Аналоговый вход/ADC	Аналоговый вход 7
16	A8	Аналоговый вход/ADC	Аналоговый вход 8
17	A9	Аналоговый вход/ADC	Аналоговый вход 9
18	A10	Аналоговый вход/ADC	Аналоговый вход 10
19	A11	Аналоговый вход/ADC	Аналоговый вход 11
20	A12	Аналоговый вход/ADC	Аналоговый вход 12
21	A13	Аналоговый вход/ADC	Аналоговый вход 13
22	A14	Аналоговый вход/ADC	Аналоговый вход 14
23	A15	Аналоговый вход/ADC	Аналоговый вход 15

Таблица 3.2 – Цифровые выходы

Pin	Функция	Тип	Описание
1	2	3	4
1	D0	Цифровой/GPIO	Цифровой вывод 0 / RX0 (Serial)
2	D1	Цифровой/GPIO	Цифровой вывод 1 / TX0 (Serial)
3	D2	Цифровой/GPIO	Цифровой вывод 2 / Interrupt 0 / PWM
4	D3	Цифровой/GPIO	Цифровой вывод 3 / Interrupt 1 / PWM
5	D4	Цифровой/GPIO	Цифровой вывод 4 / PWM
6	D5	Цифровой/GPIO	Цифровой вывод 5 / PWM
7	D6	Цифровой/GPIO	Цифровой вывод 6 / PWM
8	D7	Цифровой/GPIO	Цифровой вывод 7 / PWM
9	D8	Цифровой/GPIO	Цифровой вывод 8 / PWM
10	D9	Цифровой/GPIO	Цифровой вывод 9 / PWM
11	D10	Цифровой/GPIO	Цифровой вывод 10 / PWM
12	D11	Цифровой/GPIO	Цифровой вывод 11 / PWM

Продолжение таблицы 3.2

1	2	3	4
13	D12	Цифровой/GPIO	Цифровой вывод 12 / PWM
14	D13	Цифровой	Встроенный све- тодиод / PWM
15	D14	Цифровой/GPIO	Цифровой вывод 14 / TX3 (Serial)
16	D15	Цифровой/GPIO	Цифровой вывод 15 / RX3 (Serial)
17	D16	Цифровой/GPIO	Цифровой вывод 16 / TX2 (Serial)
18	D17	Цифровой/GPIO	Цифровой вывод 17 / RX2 (Serial)
19	D18	Цифровой/GPIO	Цифровой вывод 18 / TX1 (Serial)
20	D19	Цифровой/GPIO	Цифровой вывод 19 / RX1 (Serial)
21	D20	Цифровой/GPIO	Цифровой вывод 20 / SDA (I2C)
22	D21	Цифровой/GPIO	Цифровой вывод 21 / SCL (I2C)
23	D22	Цифровой/GPIO	Цифровой вывод 22
24	D23	Цифровой/GPIO	Цифровой вывод 23
25	D24	Цифровой/GPIO	Цифровой вывод 24
26	D25	Цифровой/GPIO	Цифровой вывод 25

Продолжение таблицы 3.2

1	2	3	4
27	D26	Цифровой/GPIO	Цифровой вывод 26
28	D27	Цифровой/GPIO	Цифровой вывод 27
29	D28	Цифровой/GPIO	Цифровой вывод 28
30	D29	Цифровой/GPIO	Цифровой вывод 29
31	D30	Цифровой/GPIO	Цифровой вывод 30
32	D31	Цифровой/GPIO	Цифровой вывод 31
33	D32	Цифровой/GPIO	Цифровой вывод 32
34	D33	Цифровой/GPIO	Цифровой вывод 33
35	D34	Цифровой/GPIO	Цифровой вывод 34
36	D35	Цифровой/GPIO	Цифровой вывод 35
37	D36	Цифровой/GPIO	Цифровой вывод 36
38	D37	Цифровой/GPIO	Цифровой вывод 37
39	D38	Цифровой/GPIO	Цифровой вывод 38
40	D39	Цифровой/GPIO	Цифровой вывод 39

Продолжение таблицы 3.2

1	2	3	4
41	D40	Цифровой/GPIO	Цифровой вывод 40
42	D41	Цифровой/GPIO	Цифровой вывод 41
43	D42	Цифровой/GPIO	Цифровой вывод 42
44	D43	Цифровой/GPIO	Цифровой вывод 43
45	D44	Цифровой/GPIO	Цифровой вывод 44 / PWM
46	D45	Цифровой/GPIO	Цифровой вывод 45 / PWM
47	D46	Цифровой/GPIO	Цифровой вывод 46 / PWM
48	D47	Цифровой/GPIO	Цифровой вывод 47
49	D48	Цифровой/GPIO	Цифровой вывод 48
50	D49	Цифровой/GPIO	Цифровой вывод 49
51	D50	Цифровой/GPIO	Цифровой вывод 50
52	D51	Цифровой/GPIO	Цифровой вывод 51
53	D52	Цифровой/GPIO	Цифровой вывод 52
54	D53	Цифровой/GPIO	Цифровой вывод 53

### **3.3 Технология загрузчика (bootloader) в Arduino**

Загрузчик (bootloader) является важным компонентом экосистемы Arduino, обеспечивающим простоту и удобство загрузки программ на микроконтроллер. Что такое загрузчик?

Загрузчик – это небольшая программа, которая хранится в специальной области флеш-памяти микроконтроллера. Его основная задача – принимать пользовательские программы (скетчи) через последовательный интерфейс и записывать их во флеш-память микроконтроллера для последующего выполнения.

Как работает загрузчик?

**Первичная загрузка:** При включении питания или перезагрузке микроконтроллера сначала выполняется загрузчик.

**Проверка последовательного порта:** Загрузчик проверяет, поступают ли данные через последовательный порт (например, через USB).

**Прием данных:** Если данные поступают, загрузчик принимает их, проверяет на корректность и записывает во флеш-память.

**Запуск пользовательской программы:** После завершения записи загрузчик передает управление пользовательской программе, если таковая присутствует.

Зачем нужен загрузчик?

**Упрощение процесса загрузки кода:** Загрузчик позволяет загружать программы через стандартные интерфейсы, такие как USB или UART, без необходимости использования внешнего программатора.

**Легкость обновления:** Программы могут обновляться легко и быстро через последовательный интерфейс, что особенно полезно в процессе разработки и отладки.

#### **3.3.1 Архитектура микроконтроллера Arduino**

На рисунке 3.3 изображена схема архитектуры микроконтроллера Arduino.

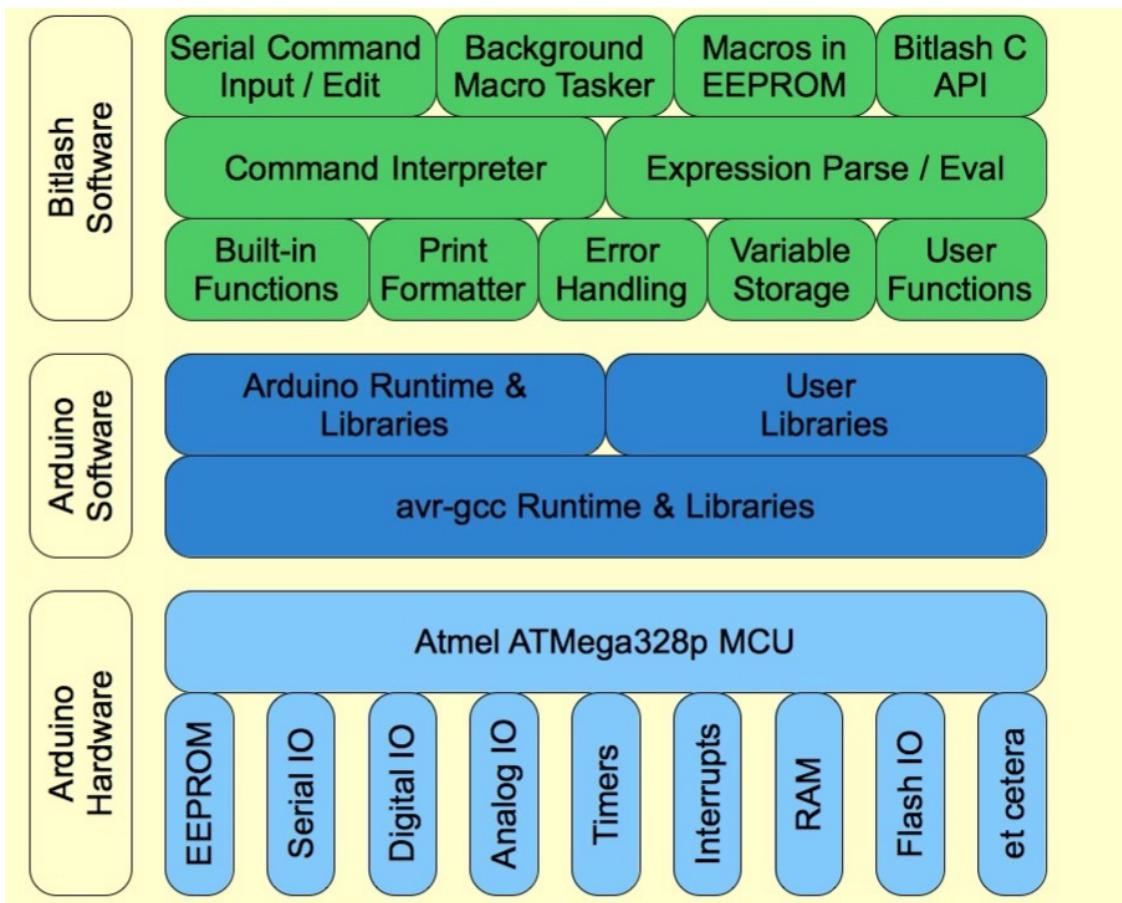


Рисунок 3.15 – Схема архитектуры микроконтроллера Arduino

В основе архитектуры контроллера лежат аппаратные и программные компоненты, взаимодействующие для выполнения различных задач.

- аппаратные компоненты: ATmega328P MCU;
- EEPROM: память для долговременного хранения данных;
- Serial IO: последовательные интерфейсы для связи с другими устройствами;
- Digital IO: цифровые входы/выходы для управления периферийными устройствами.
- Analog IO: аналоговые входы для работы с аналоговыми сигналами.
- Timers: таймеры для различных временных операций.
- Interrupts: прерывания для обработки внешних событий.
- RAM: оперативная память для временного хранения данных.
- Flash IO: память для хранения программного кода.
- прочее: дополнительные периферийные устройства и функции.
- программные компоненты: Arduino Software;

- Arduino Runtime & Libraries: среда выполнения и стандартные библиотеки для работы с аппаратными компонентами.
  - User Libraries: пользовательские библиотеки для расширения функциональности. avr-gcc Runtime & Libraries
    - компилятор и библиотеки для работы с кодом на языке C/C++, Bitlash Software;
    - Serial Command Input/Edit: ввод и редактирование команд через последовательный интерфейс.
    - Background Macro Tasker: выполнение макросов в фоновом режиме;
    - Macros in EEPROM: хранение макросов в EEPROM;
    - Bitlash C API: API для работы с Bitlash на языке C;
    - Command Interpreter: интерпретатор команд;
    - Expression Parse/Eval: разбор и выполнение выражений;
    - Built-in Functions: встроенные функции;
    - Print Formatter: форматирование вывода;
    - Error Handling: обработка ошибок;
    - Variable Storage: хранение переменных;
    - User Functions: пользовательские функции.

Эта архитектура позволяет Arduino Mega быть гибкой и мощной платформой для разработки широкого спектра микроконтроллерных приложений.

## 4 Рабочий проект

### 4.1 Классы, используемые при разработке сайта

Можно выделить следующий список классов и их методов, использованных при разработке web-приложения (таблица ??).

Переменная	Тип	Описание
lcd	Liquid Crystal Rus	Объект для управления ЖК-дисплеем с поддержкой русского языка
one_ wire_ temp	#define	Пин для подключения датчика температуры Dallas 1820
oneWire	OneWire	Объект для работы с шиной OneWire для датчиков температуры
sensorTemp	Dallas Temperature	Объект для управления датчиком температуры Dallas 1820
LED _ R	int	Пин для подключения красного светодиода
LED _ Y	int	Пин для подключения желтого светодиода
LED _ G	int	Пин для подключения зеленого светодиода
motor	Servo	Объект для управления сервоприводом
Trig	#define	Пин для отправки сигнала ультразвукового датчика
Echo	#define	Пин для получения сигнала ультразвукового датчика
imTime	unsigned int	Время, затраченное на получение эхо-сигнала
dist	unsigned int	Расстояние до объекта, измеренное ультразвуковым датчиком

<b>Переменная</b>	<b>Тип</b>	<b>Описание</b>
DHT11_PIN	#define	Пин для подключения датчика влажности DHT11
led	int	Пин для подключения светодиода
regim	int	Переменная для хранения режима работы
noch	int	Переменная для хранения режима ожидания
flag1	int	Флаг для перехода в режим ожидания
den	int	Переменная для считывания яркости окружающей среды
foto	int	Переменная для фотодатчика
temperature	int	Переменная для хранения температуры
val	int	Переменная для случайного значения
rast	int	Среднее расстояние, измеренное ультразвуковым датчиком
rast1	int	Вспомогательное значение расстояния, измеренное ультразвуковым датчиком
rast2	int	Вспомогательное значение расстояния, измеренное ультразвуковым датчиком
rast3	int	Вспомогательное значение расстояния, измеренное ультразвуковым датчиком
distMax	int	Максимальное измеренное расстояние
i_ugol	int	Угол поворота сервопривода
time	float	Время в секундах
time1	float	Время начала измерений в миллисекундах
flag2	bool	Логический флаг для различных условий

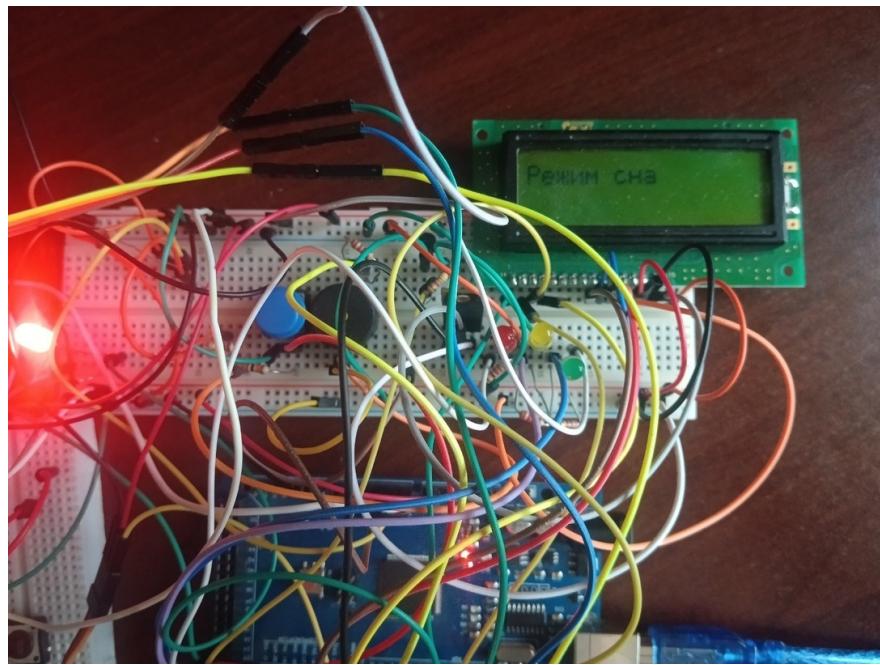


Рисунок 4.1

<b>Переменная</b>	<b>Тип</b>	<b>Описание</b>
sig	int	Сигнал для определения нажатия кнопки
Counter	int	Счетчик нажатий кнопки
lastSignal	int	Последний сигнал для определения нажатия кнопки
button	#define	Пин для подключения кнопки
fotoresistor	#define	Пин для подключения фоторезистора
termo	#define	Пин для подключения термодатчика
LCD	#define	Пин для управления ЖК-дисплеем
buzzer	#define	Пин для подключения зуммера
Ultr_dist	float	Функция для измерения расстояния с помощью ультразвукового датчика

## 4.2 Модульное тестирование разработанного мультиробота

Модульный тест для класса `regsleep` представлен на рисунке ??.

Модульный тест для класса `regwait` представлен на рисунке 4.3.

Модульный тест для класса `regspeed` представлен на рисунке 4.3.

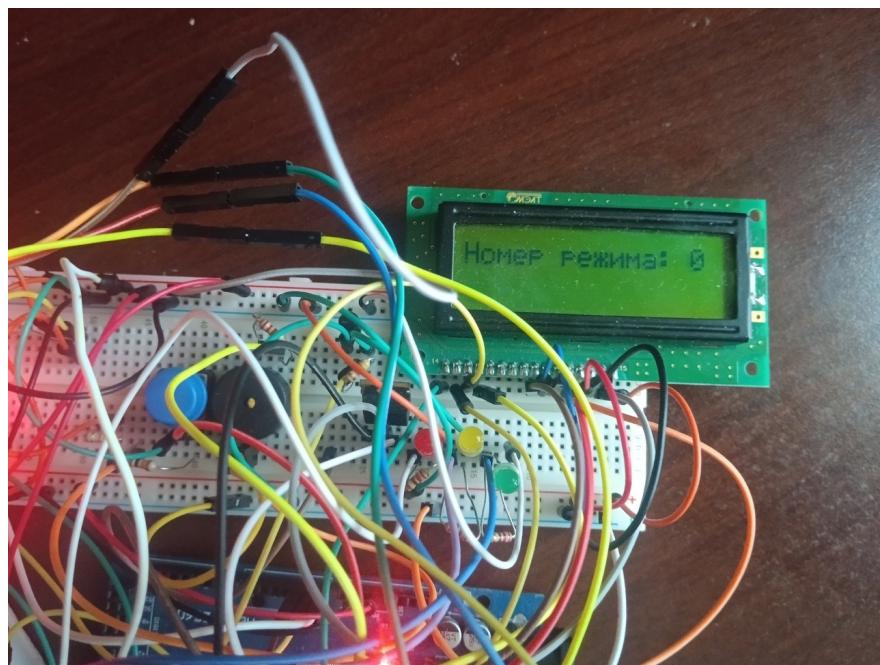


Рисунок 4.2

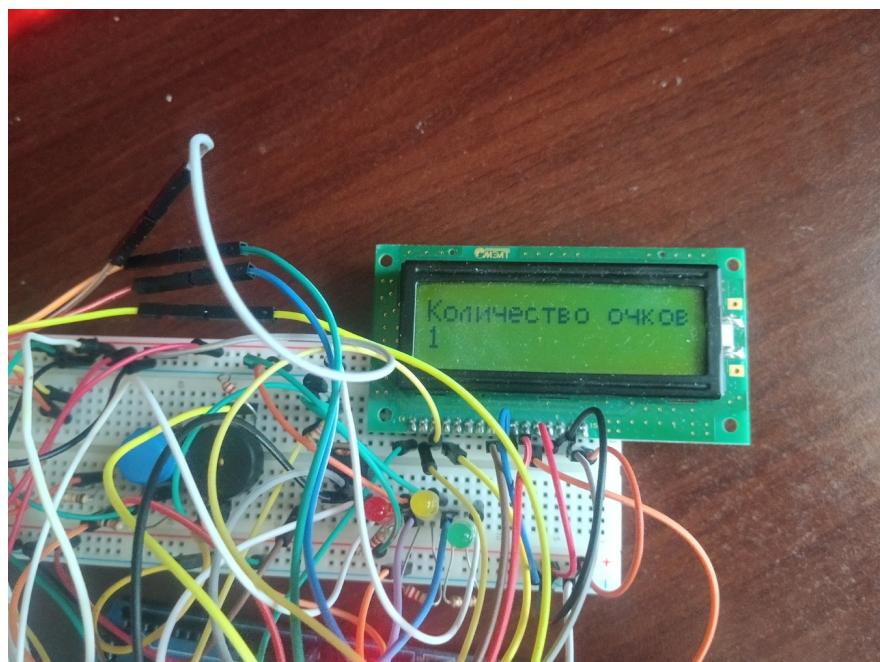


Рисунок 4.3

Модульный тест для класса `regekran` представлен на рисунках 4.3.

Модульный тест для класса `regpoisk` представлен на рисунке 4.3.

Модульный тест для класса `regtemp` представлен на рисунке 4.3.

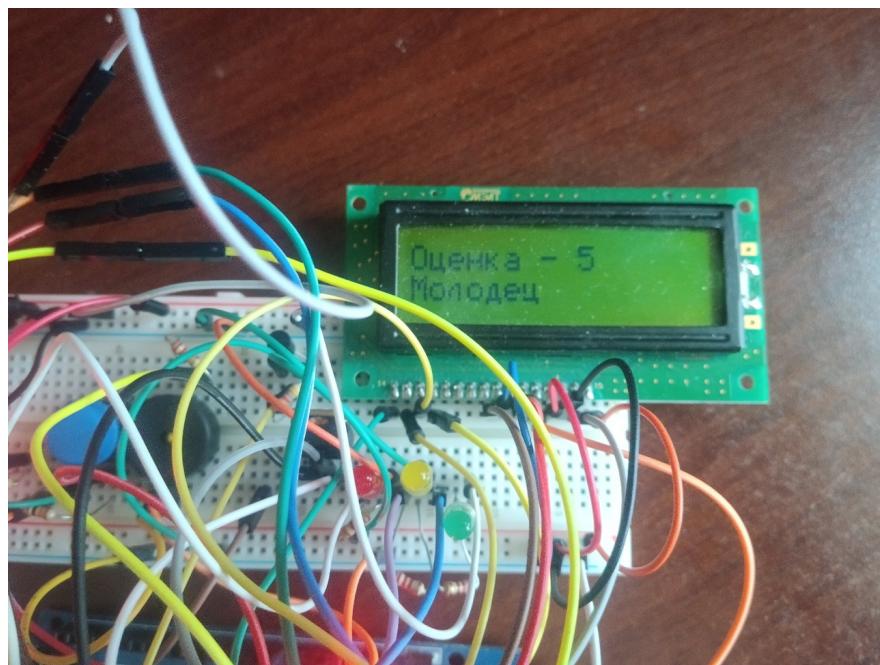


Рисунок 4.4

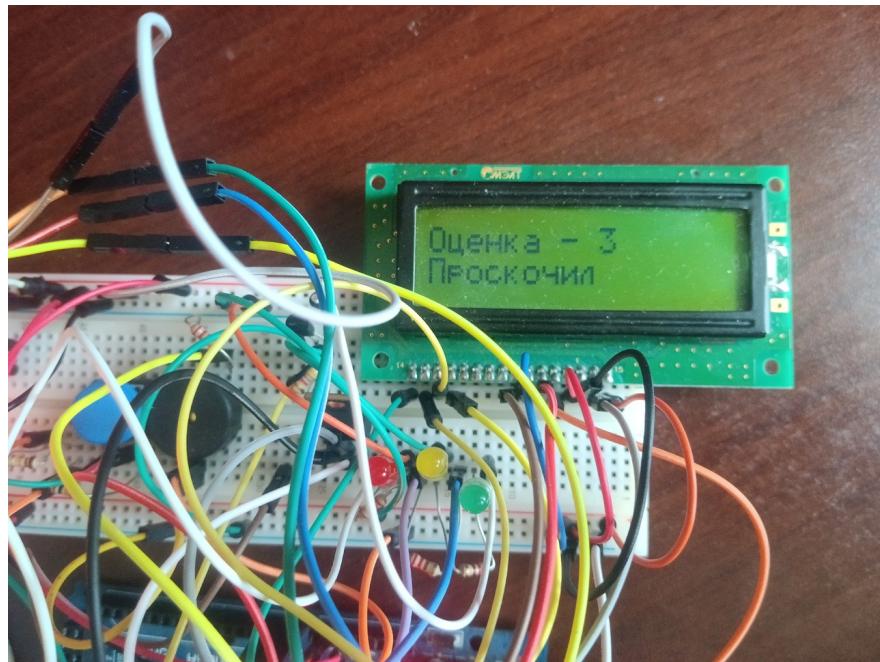


Рисунок 4.5

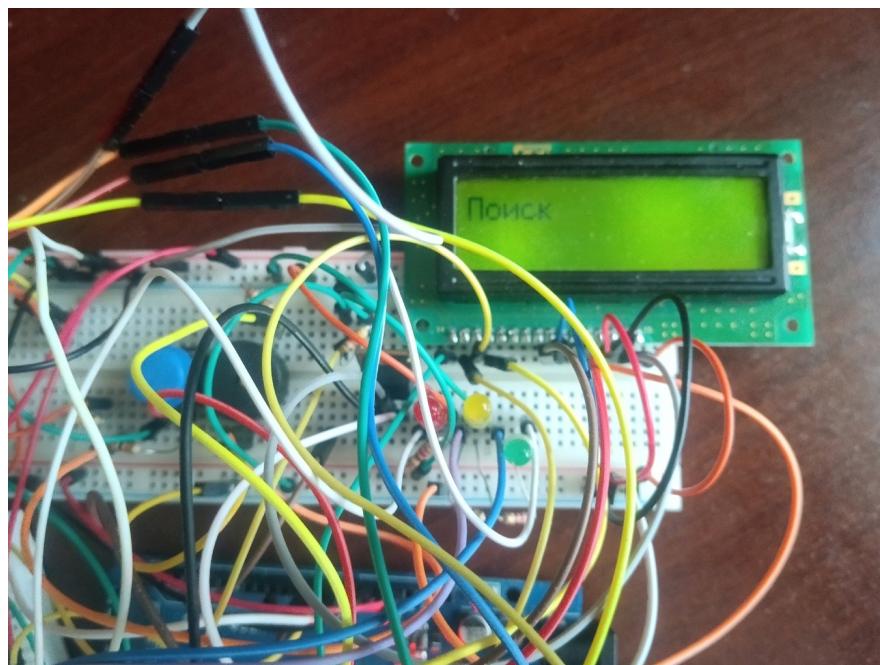


Рисунок 4.6



Рисунок 4.7

### 4.3 Системное тестирование разработанного устройства

На рисунке 4.3 представлен собранный прототип мультиробота.

На рисунке 4.3 представлен динамический вывод заголовков, включающий в себя искомые фразы при поиске фраз.

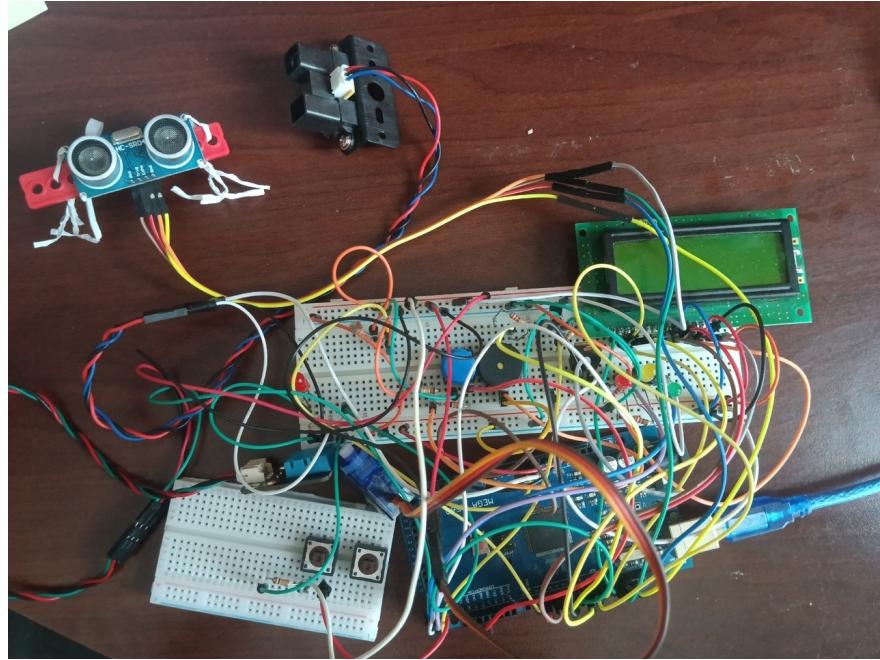


Рисунок 4.8

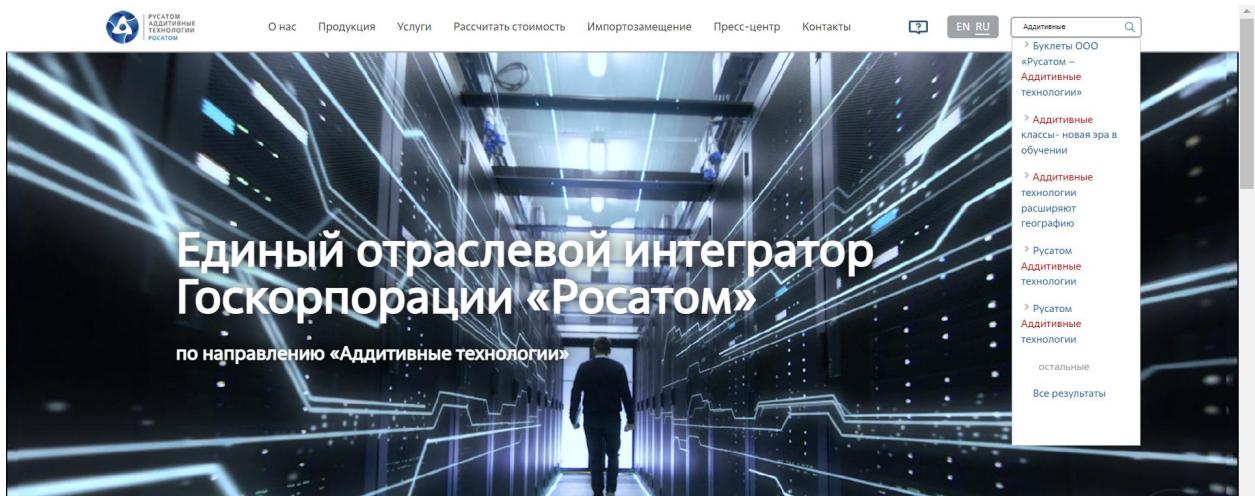


Рисунок 4.9 – Динамический вывод заголовков

На рисунке 4.4 представлен ввод данных для публикации новости.

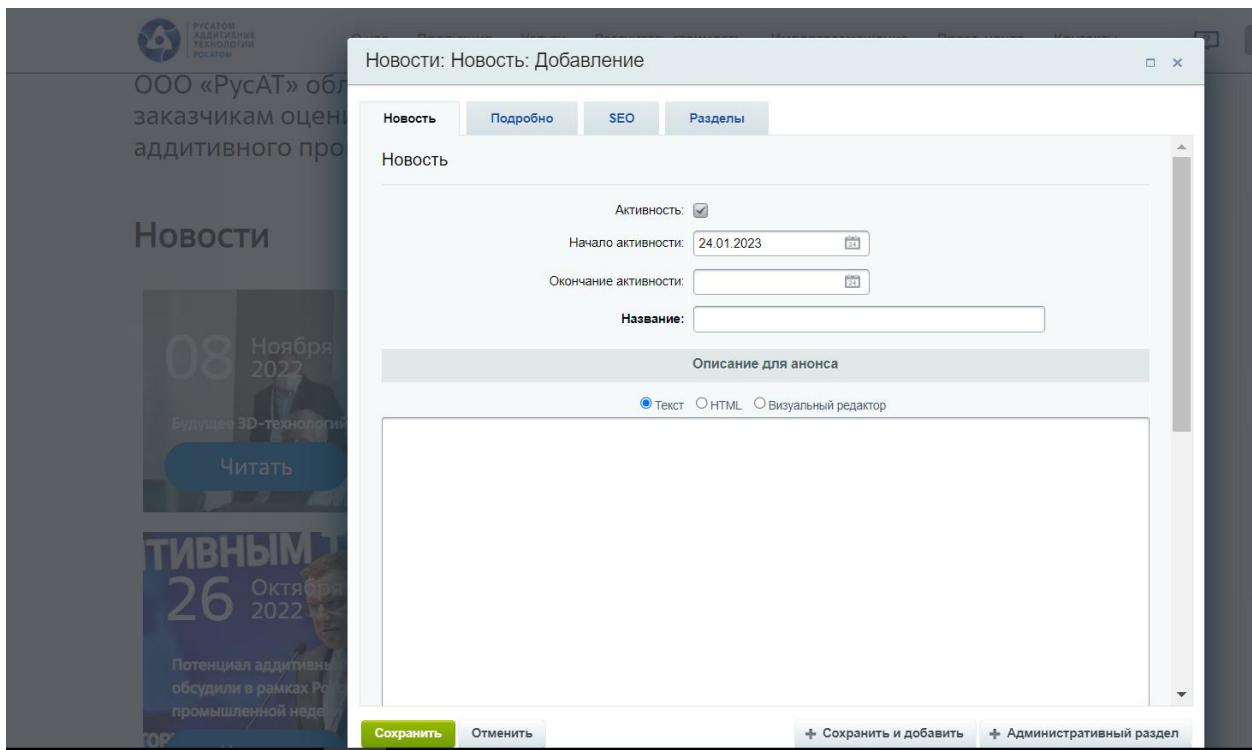


Рисунок 4.10 – Ввод данных для публикации очень-очень длинной, интересной и полезной новости

## ЗАКЛЮЧЕНИЕ

Преимущества аддитивных технологий заключается в разнообразии процессов, позволяющих применять их в различных областях производства. Существенным ограничением же является и экономическая составляющая, которая не позволит внедрить аддитивное производство повсеместно.

Компании, видя, как развиваются информационные технологии, пытаются использовать их выгодно для своего бизнеса, запуская свой сайт для того, чтобы заявить о своем существовании, проинформировать потенциального клиента об услугах или продуктах, которые предоставляет. Для продвижения компании «Русатом – Аддитивные технологии» был разработан веб-сайт на основе системы «1С-Битрикс: Управление сайтом».

Основные результаты работы:

1. Проведен анализ предметной области. Выявлена необходимость использовать 1С-Битрикс.
2. Разработана концептуальная модель web-сайта. Разработана модель данных системы. Определены требования к системе.

3. Осуществлено проектирование web-сайта. Разработана архитектура серверной части. Разработан пользовательский интерфейс web-сайта.

4. Реализован и протестирован web-сайт. Проведено модульное и системное тестирование.

Все требования, объявленные в техническом задании, были полностью реализованы, все задачи, поставленные в начале разработки проекта, были также решены.

Готовый рабочий проект представлен адаптивной версткой сайта. Сайт находится в публичном доступе, поскольку опубликован в сети Интернет.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Марголис, М. Книга рецептов по Arduino / М. Марголис. – Санкт-Петербург: Питер, 2019. – 800 с. – ISBN 978-5-4461-0922-6. – Текст: непосредственный.
2. Монк, С. Программирование Arduino: начальные шаги / С. Монк. – Москва: Эксмо, 2020. – 352 с. – ISBN 978-5-699-94193-5. – Текст: непосредственный.
3. Блум, Дж. Изучаем Arduino: инструменты и техники волшебства инженерии / Дж. Блум. – Москва: Вильямс, 2021. – 450 с. – ISBN 978-5-8459-1834-7. – Текст: непосредственный.
4. Шмидт, М. Руководство по Arduino / М. Шмидт. – Санкт-Петербург: Питер, 2021. – 288 с. – ISBN 978-5-4461-1265-3. – Текст: непосредственный.
5. Шерц, П., Монк, С. Практическая электроника для изобретателей / П. Шерц, С. Монк. – Москва: Диалектика, 2019. – 960 с. – ISBN 978-5-907114-24-0. – Текст: непосредственный.
6. Макробертс, М. Начинаем работать с Arduino / М. Макробертс. – Санкт-Петербург: Питер, 2019. – 528 с. – ISBN 978-5-4461-1028-4. – Текст: непосредственный.
7. Карвинен, Т., Карвинен, К., Валлокари, В. Сенсоры для Arduino и Raspberry Pi / Т. Карвинен, К. Карвинен, В. Валлокари. – Санкт-Петербург: Питер, 2020. – 448 с. – ISBN 978-5-4461-1104-5. – Текст: непосредственный.
8. Боксалл, Дж. Arduino Workshop: введение с 65 проектами / Дж. Боксалл. – Москва: Вильямс, 2018. – 432 с. – ISBN 978-5-8459-1878-1. – Текст: непосредственный.
9. Геддес, М. Справочник проектов Arduino: 25 практических проектов / М. Геддес. – Санкт-Петербург: Питер, 2021. – 272 с. – ISBN 978-5-4461-1306-3. – Текст: непосредственный.
10. Андерсон, Р., Церво, Д. Pro Arduino / Р. Андерсон, Д. Церво. – Москва: Эксмо, 2021. – 520 с. – ISBN 978-5-699-94563-6. – Текст: непосредственный.

11. Монк, С. Программирование Arduino: следующие шаги / С. Монк.  
– Санкт-Петербург: Питер, 2020. – 392 с. – ISBN 978-5-4461-1204-2. – Текст:  
непосредственный.
12. Платт, Ч. Электроника: обучение через открытие / Ч. Платт. –  
Москва: Диалектика, 2019. – 560 с. – ISBN 978-5-907114-31-8. – Текст: непо-  
средственный.

## ПРИЛОЖЕНИЕ А

### Представление графического материала

Графический материал, выполненный на отдельных листах, изображен на рисунках А.1–А.0.

## ПРИЛОЖЕНИЕ Б

### Фрагменты исходного кода программы

#### main.tex

```
1 \input{setup.tex}
2
3 % Режим шаблона (должен быть включен один из трех)
4 \BKPtrue
5 \%Практикtrue
6 \%Курсоваяtrue
7
8 \newcommand{\Дисциплина}{<<Проектирование и архитектура программных систем>>}
9   % для курсовой
10 \newcommand{\КодСпециальности}{09.03.04} % Курсовая
11 \newcommand{\Специальность}{Программная инженерия} % Курсовая
12 \newcommand{\Тема}{Разработка web-сайта «Русатом – Аддитивные технологии» на
13   платформе} % ВКР Курсовая
14 \newcommand{\ТемаВтораяСтрока}{1С-Битрикс}
15 \newcommand{\ГдеПроводитсяПрактика}{ООО «Предприятие ВТИ-Сервис»} % для
16   практики
17 \newcommand{\РуководительПрактПредпр}{} % для практики
18 \newcommand{\ДолжнРуководительПрактПредпр}{} % для практики
19 \newcommand{\РуководительПрактУнивер}{Бобырь М. В.} % для практики
20 \newcommand{\ДолжнРуководительПрактУнивер}{к.т.н. доцент} % для практики
21 \newcommand{\Автор}{И. И. Иванов}
22 \newcommand{\АвторРод}{Иванова И.И.}
23 \newcommand{\АвторПолностьюРод}{Дмитриева Даниила Вячеславовича} % для
24   практики
25 \newcommand{\Шифр}{xx-xx-xxxx}
26 \newcommand{\Курс}{4 } % для практики
27 \newcommand{\Группа}{ ПО-016}
28 \newcommand{\Руководитель}{М. В. Бобырь} % для ВКР и курсовой
29 \newcommand{\Нормоконтроль}{А. А. Чаплыгин} % для ВКР
30 \newcommand{\ЗавКаф}{А. В. Малышев} % для ВКР
31 \newcommand{\ДатаПриказа}{«07» апреля 2023~г.} % для ВКР
32 \newcommand{\НомерПриказа}{1505-с} % для ВКР
33 \newcommand{\СрокПредоставления}{«13» июня 2023~г.} % для ВКР, курсового
34
35 \begin{document}
36 \maketitle
37 \ifПрактика{}\else{
38   \input{ЛистЗадания}
39   \input{Реферат}\fi
40 \tableofcontents
41 \input{Обозначения}
42 \ifПрактика{}\else{\input{Введение}}\fi
43 \input{Анализ}
44 \input{ТехЗадание}
45 \input{ТехПроект}
46 \ifПрактика{}\else{
47   \input{РабочийПроект}
48   \input{Заключение}
49 }\fi
```

```
46 \input{СписокИсточников}
47 \ifBKP{\input{Плакаты}}\fi
48 \ifПрактика{}\else{\input{Код}}\fi
49 \end{document}
```

## **Место для диска**