

Минобрнауки России

Юго-Западный государственный университет

Кафедра программной инженерии

ОТЧЕТ

о преддипломной (производственной) практике

наименование вида и типа практики

на (в) ООО «Предприятие ВТИ-Сервис»

наименование предприятия, организации, учреждения

Студента 4 курса, группы ПО-016

курса, группы

Дмитриева Даниила Вячеславовича

фамилия, имя, отчество

Руководитель практики от  
предприятия, организации,  
учреждения

Оценка

должность, звание, степень

фамилия и. о.

подпись, дата

Руководитель практики от  
университета

Оценка

К.Т.Н. доцент

должность, звание, степень

Чаплыгин А. А.

фамилия и. о.

подпись, дата

Члены комиссии

подпись, дата

фамилия и. о.

подпись, дата

фамилия и. о.

подпись, дата

фамилия и. о.

Курск 2024 г.

## СОДЕРЖАНИЕ

1	Анализ предметной области	5
1.1	Характеристика предприятия и его деятельности. Тест длинного заголовка, не должен содержать переносы	5
1.2	технологии Arduino, их классификация	5
1.3	Программное обеспечение	6
1.4	История развития Arduino	8
1.5	Использование Arduino в России	9
2	Техническое задание	11
2.1	Основание для разработки	11
2.2	Цель и назначение разработки	11
2.3	Требования к оформлению документации	12
2.4	вариант использования: использование после запуска	12
2.4.1	вариант использования: пользователь установил значение счетчика равное 1	12
2.4.2	вариант использования: пользователь установил значение счетчика равное 2	12
2.4.3	вариант использования: пользователь установил значение счетчика равное 3	13
2.4.4	вариант использования: пользователь установил значение счетчика равное 4	13
2.4.5	вариант использования: пользователь завершил выполнение 1 из режимов	13
2.4.6	вариант использования: мультиробот находится в ночном режиме, но уровень яркости растет	14
3	Технический проект	15
3.1	Общая характеристика организации решения задачи	15
3.2	Обоснование выбора технологии проектирования	15
3.2.1	Описание используемых технологий и языков программирования	15

3.2.2	Язык программирования Arduino (C++)	15
3.2.2.1	Достоинства языка C++	17
3.2.2.2	Недостатки языка C++	17
	СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	18

## 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.

## **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 Программное обеспечение**

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

PlatformIO: Альтернативная среда разработки с поддержкой различных плат и расширенными возможностями. 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.

Языки программирования 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.

**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.

**Дополнительные инструменты и экосистема 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.

**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.

**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.

**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.

**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.

## **1.4 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 (IDII) 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 исследовательской 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.

## **1.5 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 прототи-

пов 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 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.

## **2.4 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.

### **2.4.1 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).

### **2.4.2 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.

#### **2.4.3 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.

#### **2.4.4 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 (приготовьтесь), 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.

#### **2.4.5 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.

#### **2.4.6 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.

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

Arduino (C++): Основной язык для разработки скетчей Arduino. Этот язык представляет собой упрощённый вариант C++, адаптированный для ра-

боты 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 инту-



итивно 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.

### **3.2.2.1 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.

### **3.2.2.2 Недостатки языка C++**

Отсутствие автоматического управления памятью:

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

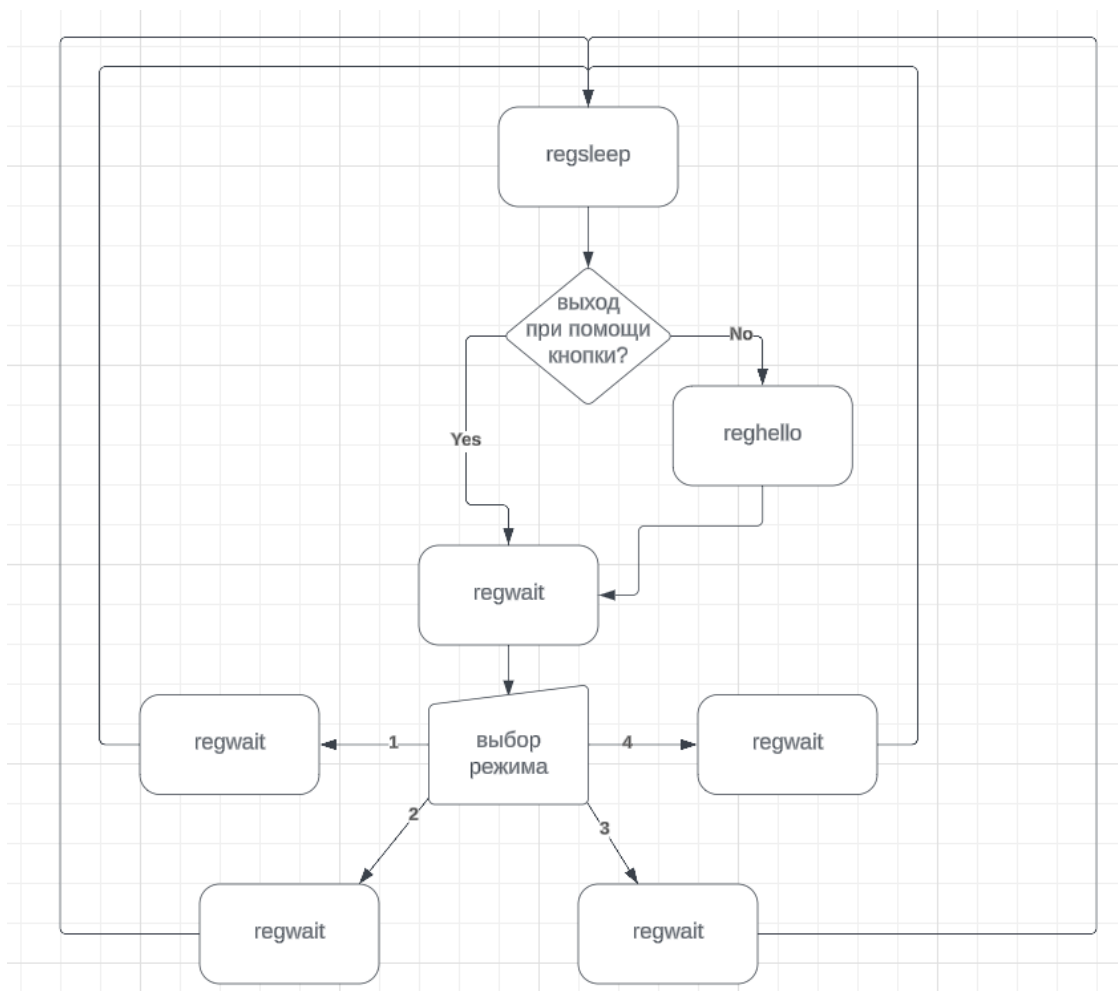


Рисунок 3.1 – схема работы кода

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 объемами данных.

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

## 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. Макробретс, М. Начинаем работать с Arduino / М. Макробретс. – 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: непосредственный.