

Министерства науки и высшего образования Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Физический факультет
Кафедра общей и экспериментальной физики

ДОПУСТИТЬ К ЗАЩИТЕ В ГЭК

Руководитель ООП
д-р физ.-мат. наук, профессор
О.Н. Чайковская
« 16 » июня 2020 г.

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

СОЗДАНИЕ АВТОМАТИЗИРОВАННОЙ ЛАБОРАТОРНОЙ УСТАНОВКИ С
УДАЛЕННЫМ ДОСТУПОМ НА БАЗЕ ПЛАТФОРМЫ ARDUINO
по основной образовательной программе подготовки магистров
"Фундаментальная и прикладная физика"
направление подготовки 03.04.02 – Физика

Колесников Даниил Александрович

Научные руководители ВКР
д-р физ.-мат. наук, профессор
Горчаков Л.В. Горчаков
« 16 » июня 2020 г.

канд. физ.-мат. наук
Печерицын А.А. Печерицын
« 16 » июня 2020 г.

Автор работы
студент группы № 051861
Д.А. Колесников

Томск-2020

МИНОБРНАУКИ РОССИИ
Федеральное государственное автономное образовательное
учреждение высшего образования
«Национальный исследовательский Томский государственный университет»
Физический факультет (ФФ)
Кафедра общей и экспериментальной физики

УТВЕРЖДАЮ

Зав. кафедрой общей и
экспериментальной физики

 Демкин В.П.

« 07 » сентября 2019г.

ЗАДАНИЕ

по подготовке ВКР _____ *магистра*

студенту _____ Колесникову Даниилу Александровичу _____ группы № 051861

фамилия, имя, отчество

1. Тема ВКР работы: Создание автоматизированной лабораторной установки с удаленным доступом на базе платформы Arduino

2. Срок сдачи студентом выполненной ВКР:

а) на кафедре _____ до **11.06.2020**

б) в ГЭК _____ до **20.06.2020**

3. Исходные данные к работе:

Целью данной работы является создание автоматизированного лабораторного практикума по физике с удаленным доступом на базе платформы Arduino.

Задачи:

1. Сборка лабораторной установки для определения скорости звука в воздухе методом стоячих волн в трубе.
2. Анализ вариантов и разработка архитектуры сети удаленного доступа.
3. Программирование аппаратных модулей, входящих в лабораторную установку.
4. Создание сайта для дистанционной работы с лабораторной установкой.
5. Тестирование лабораторной установки.

цели и задачи исследования,

Объект исследования: лабораторные системы для определения скорости звука в воздухе методом стоячих волн в трубе на базе платформы Arduino

объекты и методы исследования,

Тестирование лабораторной установки, апробация лабораторного практикума в учебном процессе

методы оценки достоверности результатов

4. Краткое содержание работы:

Сборка лабораторной установки и разработка архитектуры сети удаленного доступа, написание программ для модулей Arduino, входящих в установку – до 26.12.2019, создание веб-сайта для работы с лабораторной установкой, тестирование установки – до 15.04.2020.

дать перечень основных разделов, сроки их выполнения и ожидаемые результаты

5. Указать предприятие, организацию по заданию которого выполняется работа _____
Кафедра общей и экспериментальной физики ФФ ТГУ

6. Перечень графического материала (с точным указанием обязательных чертежей)

Презентация к докладу

7. Дата выдачи задания « 07 » сентября 2019 г.

Руководители ВКР

д-р физ.-мат. наук, профессор

Горчаков

Л.В. Горчаков

канд. физ.-мат. наук

Печерицын

А.А. Печерицын

Задание принял к исполнению 07.09.2019

Колесников

Д.А. Колесников

дата, подпись студента

АННОТАЦИЯ

Магистерская диссертация посвящена вопросам создания удаленных лабораторий, реализующих преимущества цифровых измерительных комплексов на базе платформы Arduino при их использовании в лабораторных практикумах по физике. Результатом данной работы является создание автоматизированного лабораторного практикума с удаленным доступом. Практикум включает в себя одну лабораторную работу для определения скорости звука в воздухе методом стоячих волн в трубе.

Данный практикум представляет собой комплекс технических, программных и методических средств, обеспечивающих автоматизированное проведение лабораторных и экспериментальных исследований на реальном физическом оборудовании удаленно. Взаимодействие пользователя с экспериментальной установкой осуществляется посредством веб-интерфейса.

Предлагаемая схема обеспечения удаленного доступа к лабораторному оборудованию учитывает существующие потребности в администрировании пользователей и возможности работы с установкой, как по локальной сети, так и по сети Интернет.

В работе был проведен анализ и проверка работоспособности созданного лабораторного практикума.

ОГЛАВЛЕНИЕ

Введение	3
1 Литературный обзор	5
1.1 Варианты построения лабораторных систем	5
1.2 Дистанционное управление лабораторией на базе платформы Arduino через сеть	12
1.3 Теоретические сведения	16
2 Инструменты и методы решения задачи	19
2.1 Лабораторная установка	19
2.2 Используемые аппаратные компоненты	20
2.2.1 Плата Arduino	20
2.2.2 Датчик звука	23
2.2.3 Пьезоизлучатель звука	23
2.3 Используемые программные средства	24
2.3.1 База данных Firebase Realtime	25
2.4 Описание практической части	28
2.4.1 Программирование Arduino	28
2.4.2 Разработка сайта	33
3 Методика использования лаборатории	37
3.1 Общие положения	37
3.2 Взаимодействие с установкой	38
Заключение	40
Список использованных источников и литературы	41
Приложение А Настройка Arduino IDE	46
Приложение Б Код программы для основного модуля платы	51

ВВЕДЕНИЕ

В настоящее время наблюдается тенденция к активному применению технологий дистанционного образования в учебном процессе. Весьма важным является продвижение дистанционных технологий в лабораторные практикумы и в учебный эксперимент в целях повышения эффективности и снижения материальных затрат на обучение в сфере инженерного образования. Для этого создаются автоматизированные лабораторные практикумы с удаленным доступом (АЛП УД), которые позволяют проводить учебные эксперименты на реальном физическом оборудовании, находясь в любой точке планеты через глобальную сеть Интернет [1]. (Кроме названия «автоматизированный лабораторный практикум с удалённым доступом» в литературе используются названия «лаборатория удалённого доступа», «система автоматизированного лабораторного практикума», «Интернет-лаборатория».) Особую актуальность это приобретает при переходе ВУЗов на дистанционный режим работы. В современных условиях при использовании последних достижений в информационных и телекоммуникационных технологиях АЛП УД находят применение в учебном процессе, как в системе дистанционного и открытого образования [2, 3], так и при использовании традиционной очной технологии обучения.

В данной магистерской диссертации была поставлена задача исследовать существующие решения построения лабораторных систем с возможностью удаленного доступа, выделить особенности данных систем и создать бюджетную лабораторную установку с удаленным доступом для определения скорости звука в воздухе методом стоячих волн в трубе [4].

В данной работе используется микроконтроллер Arduino, обладающий всеми необходимыми функциональными возможностями для решения задач, возникающих при разработке лабораторных установок, требующих автономной непрерывной работы. Кроме того, платформа Arduino имеет ряд преимуществ: открытая архитектура и открытый исходный код,

кроссплатформенность, низкое потребление, малые габариты и низкая стоимость.

Целью данной работы является создание автоматизированного лабораторного практикума по физике с удаленным доступом на базе платформы Arduino.

При создании АЛП УД требуется:

- во-первых, применение специальных технических средств, как для автоматизации лабораторной установки, так и для связи управляющего компьютера или сервера с удалённым пользователем, проводящим эксперимент в режиме сетевого управления;
- во-вторых, разработка прикладного программного обеспечения или использование в отдельных случаях специализированных пакетов программ;
- в-третьих, методическая поддержка лабораторных учебно-научных экспериментов.

В соответствии с этим для достижения цели магистерской диссертации были поставлены следующие задачи:

1. сборка лабораторной установки;
2. анализ вариантов и разработка архитектуры сети удаленного доступа;
3. программирование аппаратных модулей, входящих в установку;
4. создание сайта для дистанционной работы с установкой;
5. тестирование лабораторной установки.

1 ЛИТЕРАТУРНЫЙ ОБЗОР

1.1 Варианты построения лабораторных систем

В соответствии с Отраслевым стандартом [5] под лабораторией удалённого доступа понимается комплекс технических, программных и методических средств, обеспечивающих автоматизированное проведение лабораторных и экспериментальных исследований непосредственно на физических объектах и (или) математических моделях с использованием удалённого компьютерного доступа. В настоящей работе рассматриваются только такие лаборатории, в которых ведутся реальные эксперименты на физических стендах. Именно в них могут решаться задачи реального повышения качества практической подготовки [6].

Простейшая конфигурация дистанционного лабораторного практикума включает лабораторную установку, устройство для передачи данных с установки на компьютер-сервер, через который обеспечивается связь с рабочим местом клиента, подсоединенным к сети Интернет. На компьютере клиента установлено программное обеспечение, которое позволяет управлять экспериментальной установкой, считывать экспериментальные данные и обрабатывать эти данные.

Автоматизированная измерительно-управляющая система лаборатории с удаленным доступом реализуется с помощью устройств ввода/вывода аналоговых и цифровых сигналов. Эти устройства в большинстве систем вырабатывают аналоговые сигналы (напряжение, заряд или ток), и для подключения к вычислительному оборудованию необходимо использовать и создавать специальные устройства для согласования и преобразования аналоговых сигналов в цифровые прежде, чем они могут быть введены в ЭВМ, микропроцессор, контроллер, находящиеся в контуре управления системы. Устройство, позволяющие связать компьютер с экспериментальным стендом, называется устройством сопряжения с объектом или просто УСО [7,

8, 9, 10]. Лаборатории удаленного доступа могут быть построены с использованием как плат сопряжения, встраиваемых в измерительные приборы или в компьютер (приборный интерфейс GPIB, измерительно-управляющие платы Plugin-Card), так и отдельных устройств в виде программируемых логических контроллеров, магистрально-модульных систем, комбинированных многоуровневых иерархических систем и др. Рассмотрим некоторые их характеристики и особенности применения.

Магистрально-модульные системы реализуют магистрально-модульный принцип организации измерительно-управляющих систем. Здесь под магистралью понимают официально зарегистрированный международный или национальный стандарт на логические, механические и электрические параметры, обеспечивающий совместимость модулей, выполненных в данном стандарте. Модульность системы обеспечивает возможность объединения модулей с различными характеристиками для обеспечения необходимого количества и качества входных и выходных каналов.

В магистрально-модульных системах используются средства сопряжения, не зависящие от платформы и представляющие совокупность специальных элементов (модулей) с магистральным принципом организации. Обмен информацией между модулями происходит по правилам, которые не зависят от типа ЭВМ и определяются конкретной системой. Для передачи данных между системой и ЭВМ используется специальный интерфейс или контроллер, структура которого определяется типом ЭВМ.

Среди преимуществ данного подхода можно выделить: возможность создания гибких масштабируемых измерительных комплексов; создание банков совместимых компонентов системы; при смене архитектуры комплекса необходимо заменить только один элемент системы — контроллер.

К недостаткам магистрально-модульных систем можно отнести их избыточность для автоматизации простых объектов и высокую стоимость.

Применение магистрально-модульных систем целесообразно для сложных многоканальных экспериментальных стендов, когда требуется высокая точность измерений при повышенном быстродействии всей системы.

Примерами магистрально-модульных систем являются система КАМАК, канал общего пользования (КОП), общая шина (ОШ), MULTYBUS-I, MULTYBUS-II, И-41, Q-bus, VME-bus, VMX-bus, IEEE-488, HP-IB-bus, FASTBUS, и др. Из современных магистрально-модульных систем выделяется продукция National Instruments, объединяющая десятки различных модулей сбора данных, универсальные драйвера под все распространенные операционные системы и систему графического программирования LabVIEW. Из систем, выпускаемых отечественными производителями, можно выделить систему Н-2000, выпускаемую ЗАО «Л-Кард».

Примером использования магистрально модульных систем при создании АЛП УД является созданная в МГТУ им. Н. Э. Баумана Интерактивная Диалоговая Удаленная Система (ИНДУС) [11], представляющая собой программно-аппаратный комплекс на базе компьютера PC/AT и магистрально-модульного устройства сопряжения КАМАК для удаленного выполнения реального эксперимента по диагностике низкотемпературной плазмы.

В автоматизированных системах с небольшим числом каналов связи по большей части применяются измерительно-управляющие платы Plugin-Card, также часто называемые просто платами ЦАП-АЦП (цифро-аналоговых и аналого-цифровых преобразователей). Преимущества таких плат состоит в том, что между компьютером и устройствами ввода-вывода отсутствуют какие-либо промежуточные звенья, а вследствие расположения непосредственно в слотах на материнской плате IBM/PC-совместимого компьютера может быть существенно (максимум - до 132 Мбайт/с) повышена скорость обмена данными и командами. Платы имеют собственные микропроцессорные средства обработки, быстрые буфера-

накопители типа FIFO (First Input – First Output), а технические характеристики некоторых из них практически ничем не уступают приведенным выше для функциональных модулей магистрально-модульных систем.

Недостатки использования такого подхода: а) как правило, относительно дешевые платы не имеют выходного буфера в ЦАП, что вызывает необходимость работы в реальном времени; б) относительная дороговизна самих плат и необходимость использовать дорогое лицензионное программное обеспечение для импортных плат в) питание плат происходит от блока питания ПЭВМ, отсюда невысокая мощность сигналов управления. Несколько снизить стоимость автоматизации стенда позволяет использование внешних систем сбора данных, подключаемых к ПЭВМ через последовательный порт или USB-шину, однако при этом снижается быстродействие системы.

Измерительно-управляющие комплексы на базе плат ЦАП-АЦП целесообразно создавать и использовать: в системах, не требующих высокого быстродействия; при удалении комплекса от объекта исследования не далее нескольких метров; при небольшом количестве каналов измерения-управления и плат сопряжения, необходимых для автоматизации данного стенда, т.к. число слотов расширения в ПЭВМ ограничено.

Примеры: в [12] описывается физический практикум на базе платы ЦАП-АЦП «ZET-210» по изучению явлений сложения и разложения сигналов и цепей переменного тока; в [13] обсуждаются вопросы использования современных информационно-измерительных технологий на базе плат ЦАП-АЦП при совершенствовании комплексных лабораторных работ.

Реже, чем два предыдущих класса устройств связи управляющего компьютера с объектом, для автоматизации экспериментальных стендов в целях создания АЛП УД применяются программируемые логические контроллеры (ПЛК). Их применение оправдано в сложных разветвленных

измерительно-управляющих подсистемах, когда необходимо разгрузить главный компьютер от рутинной работы (сбор – накопление данных, предварительная обработка и т.п.). Скорость обмена данными, которую поддерживают ПЛК, относительно невысока. Типичный представитель этого класса устройств – микропроцессор Intel i80C51 [14].

Пример использования ПЛК при создании лабораторных стендов можно найти в работе [15], в которой был разработан учебно-методический комплекс на базе ПЛК Delta DVP-SX для изучения промышленных датчиков расхода.

Среди других способов связи ПЭВМ с экспериментальным стендом, которые могут быть использованы при создании автоматизированных лабораторий удаленного доступа, широкое применение нашел интерфейс, предложенный фирмой Hewlett-Packard (HP) в 1965 г. Его первое название - Hewlett-Packard Interface Bus (HP-IB). Он был разработан как многоцелевой приборный контроллер в измерительных системах. Комбинация измерительной техники и персонального компьютера дала пользователям принципиально новое качество работы с измерительными системами. В HP-IB были заложены прогрессивные технические принципы: достаточно высокая скорость передачи, приемлемое число приборов на шине, гибкость топологии системы, достаточно большие расстояния между приборами. Интерфейс получил большее распространение ввиду его полной открытости и документированности. В процессе стандартизации этого интерфейса в 1975 году он был переименован в GPIB (General Purpose Interface Bus) и официально одобрен Институтом инженеров по электротехнике и электронике (IEEE) как стандарт IEEE 488-1975 [IEEE-488.1], который в 1987 году стал стандартом ANSI/IEEE 488.1 и был расширен в области программного обеспечения до версии 488.2.

В результате получился надежный и эффективный канал передачи данных. Приведем основные технические характеристики приборного интерфейса GPIB:

- общая длина до 20м;
- число подключаемых модулей, определяемое нагрузочной способностью шины (без расширителей шины) - не более 15;
- общее число приемником и источников не более 31 (при однобайтовой адресации);
- максимальная скорость передачи не более 1 Мбайт/сек, реальные скорости составляют порядка 250...500 Кбайт/сек.

Наиболее очевидные недостатки GPIB: низкая скорость передачи данных; относительно высокая стоимость интерфейса; необходимость поддержки встроенного программного обеспечения 488.x.

Однако, несмотря на недостатки и мощную конкуренцию со стороны архитектур VMEbus и др., GPIB интерфейс получил широкое распространение ввиду простоты использования, постоянного развития аппаратной поддержки GPIB, разработки новых GPIB-совместимых приборов. Поскольку шина GPIB полностью стандартизована и протестирована, большинство производителей автоматизированных измерительных систем и приборов встраивают в свои изделия интерфейсы GPIB в качестве основного канала передачи данных.

Примеры создания дистанционных лабораторий на базе GPIB интерфейсов можно найти в работах [16, 17].

При автоматизации эксперимента используются также и комбинации различных УСО. Так, например, в [18, 19], использовался обмен данными ПЭВМ со специальным силовым блоком питания и управления через последовательный порт в стандарте RS-232, а сам блок MP-20M содержал логический контроллер, поддерживающий ряд макрокоманд измерения и управления.

Из бюджетных вариантов построения измерительно-управляющих систем можно выделить решения, заключающиеся в использовании дискретных электронных компонент и относительно мелких недорогих одноплатных компьютеров, в большинстве случаев строящихся на базе

микроконтроллеров. Одним из наиболее удачных представителей такого класса устройств является в настоящее время платформа Arduino. Существует несколько других брендов с довольно похожими концепциями, однако Arduino является одним из самых известных и широко распространенных.

Измерительно-управляющие комплексы на базе плат Arduino целесообразно создавать и использовать для автоматизации простых объектов, не требующих высокой точности измерений. При этом во многих случаях разработка может проходить только с использованием плат Arduino, без обычных настольных или аналогичных платформ ПК. Такой подход имеет много преимуществ, таких как невысокая цена проекта, модульность, автономность, небольшие размеры [20]. Гибридные решения, включая Arduino или аналогичные платы в сочетании с другими более мощными компьютерными платформами, могут быть оптимальными для более сложных проектов. Программирование и разработка приложений для Arduino и подобных плат поддерживаются большим сообществом разработчиков и пользователей, которые предоставляют различные библиотеки, примеры решений, форумы, охватывающие различные тематические и прикладные аспекты. Примеры реализации физических экспериментов с применением платформы Arduino представлены в работах [21, 22, 23, 24].

В следующем разделе рассматриваются некоторые аппаратные и программные конфигурации удаленных лабораторий на базе Arduino и совместимых плат.

1.2 Дистанционное управление лабораторией на базе платформы Arduino через сеть

Разработка схемы удаленного доступа проводилась на основе выбора платформы Arduino с учетом ее особенностей. Анализ существующих вариантов схем удаленного доступа с использованием других подходов представлен в работе [25] и здесь не рассматривается.

Общая схема удаленного доступа к экспериментальной установке, реализованной с помощью плат Arduino, в большинстве случаев включает в себя:

- плату общего назначения для сбора данных и управления устройствами
- датчики и исполнительные механизмы экспериментальной установки
- шилд для исполнительных механизмов экспериментальной установки
- шилд или другую плату общего назначения для связи по локальной сети (LAN/WLAN)
- сервер, предоставляющий веб-доступ

Шилд – это плата расширения. Шилды можно разделить на полноразмерные и отдельные модули. Полноразмерные своими очертаниями повторяют форму платы Arduino, будь то UNO, Nano или MEGA. Отдельные модули – это платы произвольной формы, созданные для выполнения определенного набора функций. И те и другие могут быть как общего назначения, так и для выполнения узконаправленных задач.

Оснастив плату Arduino Ethernet или WiFi шилдом, можно запрограммировать её в простой веб-сервер, и с помощью доступа к этому серверу через браузер, запущенный на любом компьютере, который подключен к той же сети, что и Arduino, взаимодействовать с

экспериментальной установкой. Такой веб-сервер, таким образом, заменяет управляющий экспериментальной установкой компьютер (его называют Lab-сервером). С другой стороны, плата с LAN/WLAN шилдом может взаимодействовать с существующим Lab-сервером, который также может предоставлять доступ к удаленному эксперименту. Рис. 1 иллюстрирует общую схему архитектуры удаленного доступа к экспериментальной установке, реализованных с помощью плат Arduino и других устройств, используемых в экспериментальной установке.

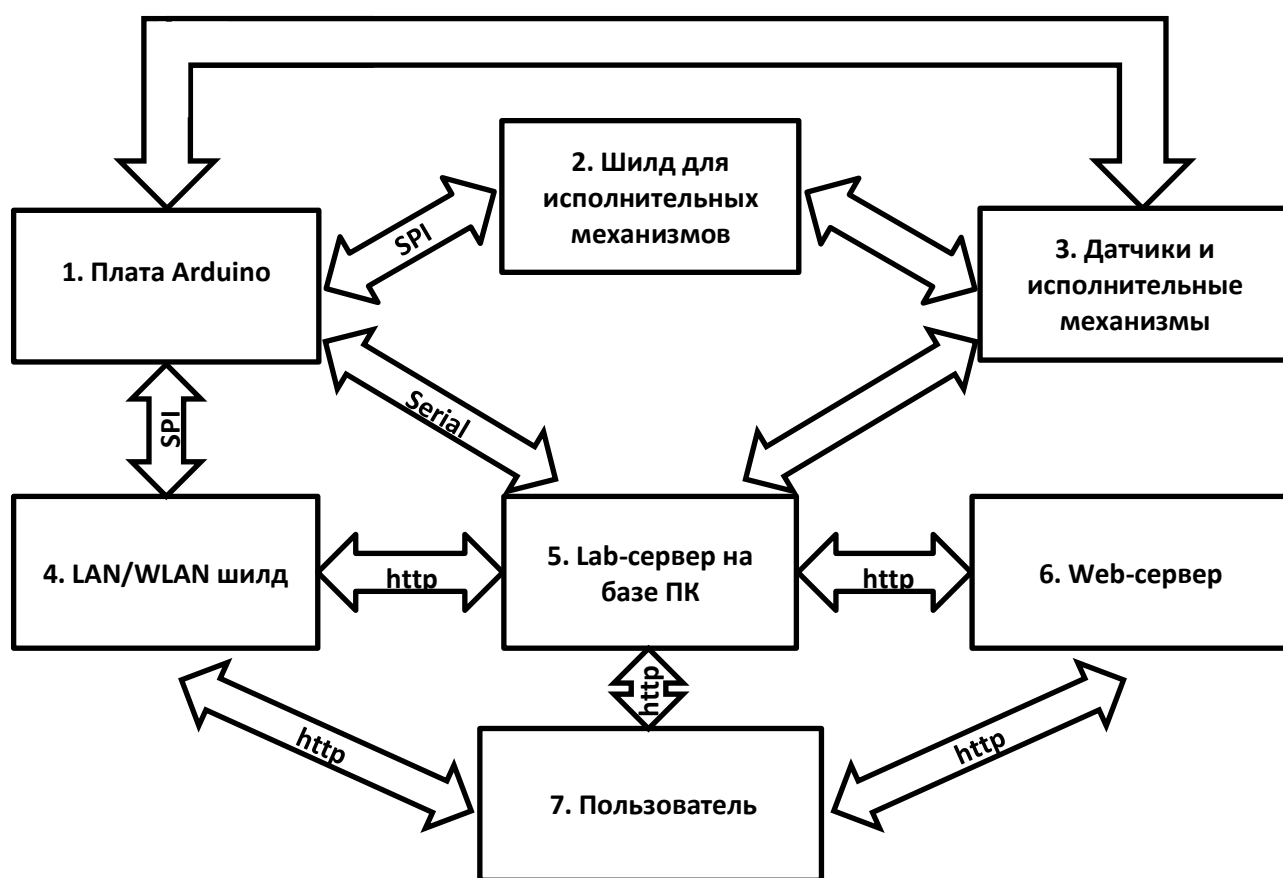


Рисунок – 1 Общая схема удаленного доступа к лабораторной установке на базе платформы Arduino

Не все соединения на рис. 1 в действительности всегда требуются, так как это зависит от требований конкретного эксперимента. Центральной частью рис. 1 является плата Arduino общего назначения (1), аналоговые и цифровые входы которой подключены к датчикам экспериментальной установки (3) непосредственно проводами. Цифровые выходы могут быть

подключены непосредственно или через соответствующий шилд (2) для исполнительных механизмов экспериментальной установки, если это необходимо. Плата общего назначения сообщается с шилдами главным образом через SPI (последовательный периферийный интерфейс).

Веб-доступ для удаленного пользователя (7) может быть обеспечен различными способами. Как показано на рис. 1, Ethernet или Wifi шилд (4) для связи по локальной проводной или беспроводной сети, можно использовать, запрограммировав его как простой веб-сервер, который может быть непосредственно доступен удаленным пользователем. В сложных проектах лучшим решением является использование отдельного Lab-сервера с программным клиентом, а также использование простого веб-сервера, обеспечиваемого шилдом, только для передачи данных. В этом случае удаленный пользователь обращается к веб-серверу, размещенному на плате Arduino или на каком-либо существующем Lab-сервере на базе ПК (5). Затем запросы пользователей передаются программному клиенту, который запрашивает данные с платы Arduino (1) с помощью Ethernet или Wifi Shield (4) и направляет полученные данные через сервер обратно удаленным пользователям. Другой способ заключается в соединении (5) посредством проводной последовательной связи с платой Arduino (1), которая управляет экспериментом, обеспечивая к ней веб-доступ. Проводная последовательная связь между двумя платами может быть последовательной (UART) или TWI (двухпроводный интерфейс), он же I2C.

Для управления экспериментом через глобальную сеть Интернет используется веб-сервер (6).

Общая схема на рис.1 может быть при необходимости расширена другими соединениями двух или более плат Arduino или сетевыми соединениями в различных иерархических конфигурациях [26].

Есть еще несколько возможностей, которые не представлены на рис. 1. Во-первых, это использование платы общего назначения с интегрированным доступом к локальной сети, такой как HWAYEN UNO R3. Функционально

это эквивалентно Arduino UNO R3 [27] с WiFi модулем ESP8266, но с несколько более низкой ценой. Во-вторых, вместо веб-сервера (6) для связи экспериментальной установки с удаленным пользователем по сети Интернет, можно использовать облачный сервис, например Firebase. В-третьих, как отмечалось, Ethernet или Wifi шилд, в случае HWAYEN UNO R3 – WiFi модуль ESP8266, можно использовать как простой веб-сервер и тем самым избавиться от Lab-сервера на базе ПК. Кроме того, модуль ESP8266 имеет 4МБ флеш-памяти, что освобождает нас от необходимости для хранения файлов веб-страниц использовать дополнительную память. С учетом этих возможностей, предполагая также, что никакие исполнительные механизмы в экспериментальной установке не используются, в данной работе была разработана новая схема удаленного доступа (рис. 2).

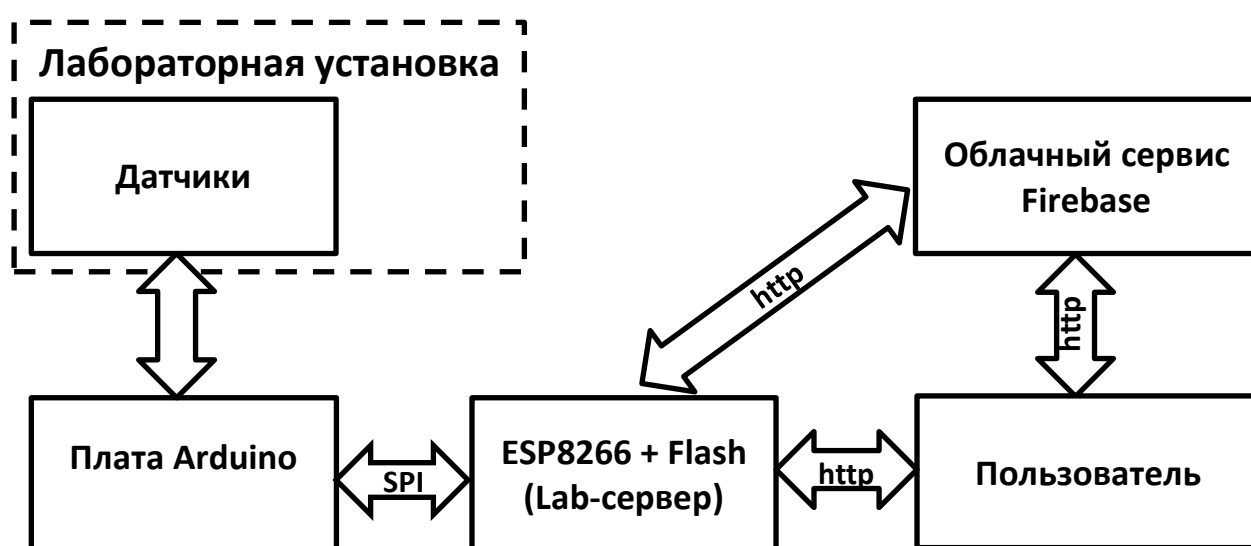


Рисунок – 2 Схема удаленного доступа к лабораторной установке на базе платформы Arduino

Использование облачного сервиса Firebase аргументируется тем, что он упрощает задачу администрирования пользователей, предоставляет хостинг и доступ к базе данных реального времени. Также существуют библиотеки Arduino, позволяющие запрограммировать ESP8266 для работы с API данного сервиса напрямую [28].

Данная схема удаленного доступа была применена при создании лабораторной установки для определения скорости звука в воздухе методом стоячих волн в трубе [4]. Краткие теоретические сведения об изучаемом в данной лабораторной работе физическом явлении приведены в следующем разделе.

1.3 Теоретические сведения

Звук - это распространяющееся в виде волн возмущение газообразной, жидкой или твердой среды посредством колебательного движения частиц. Звуковые волны являются механическими и представляют собой распространение деформации среды в виде взаимосвязанных колебаний плотности ρ , давления P , смещения частиц ξ и скорости v их движения. Область физики, изучающая звуковые явления, называется акустикой.

Если колебания (смещения) частиц происходят вдоль направления распространения волны, то волна называется продольной. Продольные волны наблюдаются в твердых телах, жидкостях и газах. Если частицы смещаются перпендикулярно направлению распространения волны, то это поперечная волна. Поперечные звуковые волны могут распространяться в различных средах, но в жидкостях и газах они очень быстро затухают, поэтому наблюдать поперечные волны легче всего в твердых телах.

Звуковая волна в газах может быть суммой колебаний с множеством различных частот. Чтобы выделить среди этих колебаний монохроматическую волну с определенной частотой используется резонатор в виде трубы длины L , заполненной воздухом. Воздух, ограниченный объемом трубы является колебательной системой, состоящей из очень большого числа частиц. Любая колебательная система, выведенная из положения равновесия и предоставленная сама себе, совершает свободные колебания, называемые собственными или нормальными колебаниями. Частоты, на которых система совершает нормальные колебания, называются собственными частотами или нормальными модами.

В дискретных системах, состоящих из N связанных колеблющихся элементов (осцилляторов), набор частот, на которых совершаются собственные колебания, равен N . В распределенных системах (струна, мембрана, резонатор) существует бесконечное множество собственных колебаний. Если на торце трубы установить мембрану, совершающую гармонические колебания, в трубе возникает плоская монохроматическая бегущая волна. Дойдя до края трубы часть волны, отразится и пойдет в обратную сторону, а часть, испытав преломление, будет распространяться в открытое пространство вне трубы. Взаимодействие бегущей и отраженной волн порождает стоячую волну.

Для определения фазовой скорости распространения звуковой волны c в воздухе используется связь этой величины с длиной волны λ и частотой звуковых колебаний f [29, с. 265]:

$$c = \lambda f \quad (1)$$

Звуковая волна, идущая от излучателя и волна, отраженная от другого конца трубы, ограниченного приемником звука (микрофоном), интерферирует в столбе воздуха, образуя стоячую волну.

Такую трубу, открытую с обеих сторон, можно считать акустическим резонатором, в котором в виде устойчивых стоячих волн существуют только те, которые удовлетворяют условию [29, с. 285]:

$$L = m\lambda/2, \quad (2)$$

Когда на длине трубы укладывается половина звуковой волны, что происходит при совпадении частоты внешнего воздействия динамика с основной частотой данной акустической системы, или кратное число половин, наблюдается увеличение акустического отклика системы в силу образования стоячих волн. Если плавно изменять частоту звуковых колебаний при неизменном значении длины трубы, будет наблюдаться последовательный ряд таких частот звуковых колебаний, при которых длина воздушного столба равна целому числу полуволн и воздушный столб в трубе звучит наиболее интенсивно.

Явление, при котором, наблюдается увеличение амплитуды колебаний системы при совпадении частоты внешнего воздействия с собственной частотой, называется резонансом.

Определение скорости звука в настоящей работе основано на измерении разностей двух соседних частот, для которых имеет место усиление сигнала на приемнике звука. Согласно (1) и (2), соответствующая расчетная формула имеет вид

$$c = 2L\overline{\Delta f}, \quad (3)$$

где c - фазовая скорость звука, L - длина трубы, $\overline{\Delta f}$ - среднее значение разностей соседних частот, на которых наблюдается резонансное усиление звука.

Таким образом, определяя набор частот, на которых происходит усиление звуковых колебаний в некотором частотном диапазоне и, зная длину трубы, по формуле (3), можно определить скорость звука.

2 ИНСТРУМЕНТЫ И МЕТОДЫ РЕШЕНИЯ ЗАДАЧИ

2.1 Лабораторная установка

Лабораторная установка для определения скорости звука методом стоячих волн в трубе представляет собой горизонтальную пластиковую трубку, с одной стороны которой прикреплен пьезодинамик, а с другой — датчик звука (рис. 3).

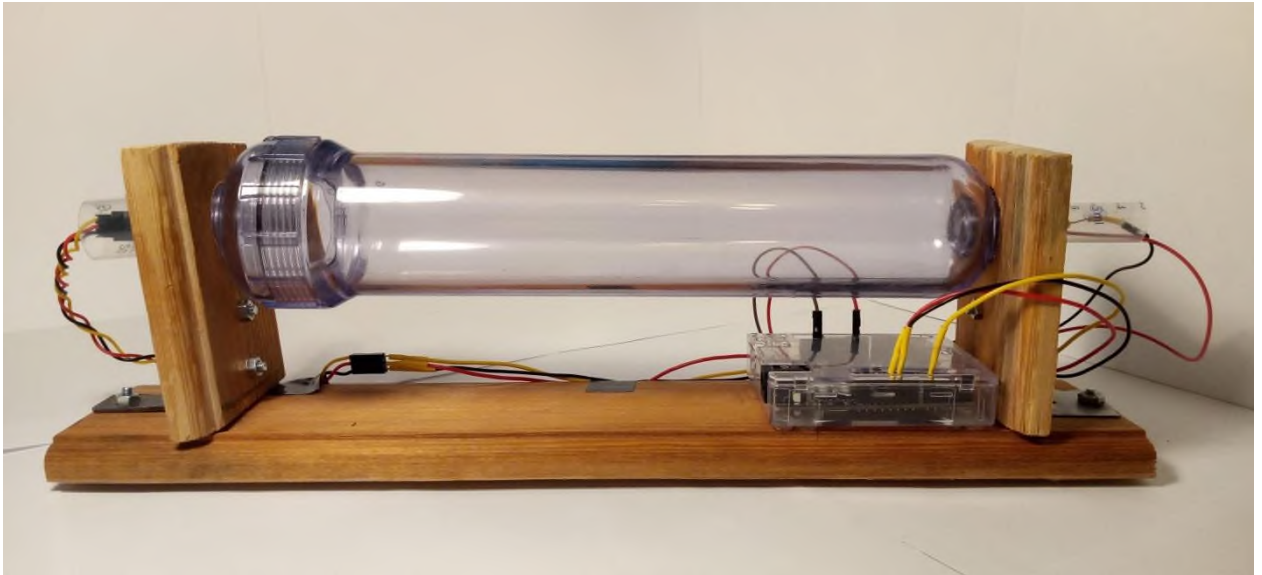


Рисунок – 3 Лабораторная установка для определения скорости звука методом стоячих волн в трубе

Схема установки изображена на рис. 4. Открытая с обоих концов труба (1) используется как акустический резонатор. Приемником звука служит модуль OSTOPUS EF04008 (2), размещенный у одного края трубы и соединенный с платой Arduino (3). Модуль подключается по схеме: вывод "V" модуля – к питанию +5 вольт, вывод "G" – к GND, вывод "S" – к аналоговому выводу "A0" платы Arduino. Пьезодинамик (4) размещен у другого края трубы и имеет два вывода. Положительный вывод подключен к цифровому выводу 9 платы Arduino с функцией ШИМ (PWM), в то время как отрицательный вывод заземлен на выводе GND Arduino через резистор на 100 Ом.

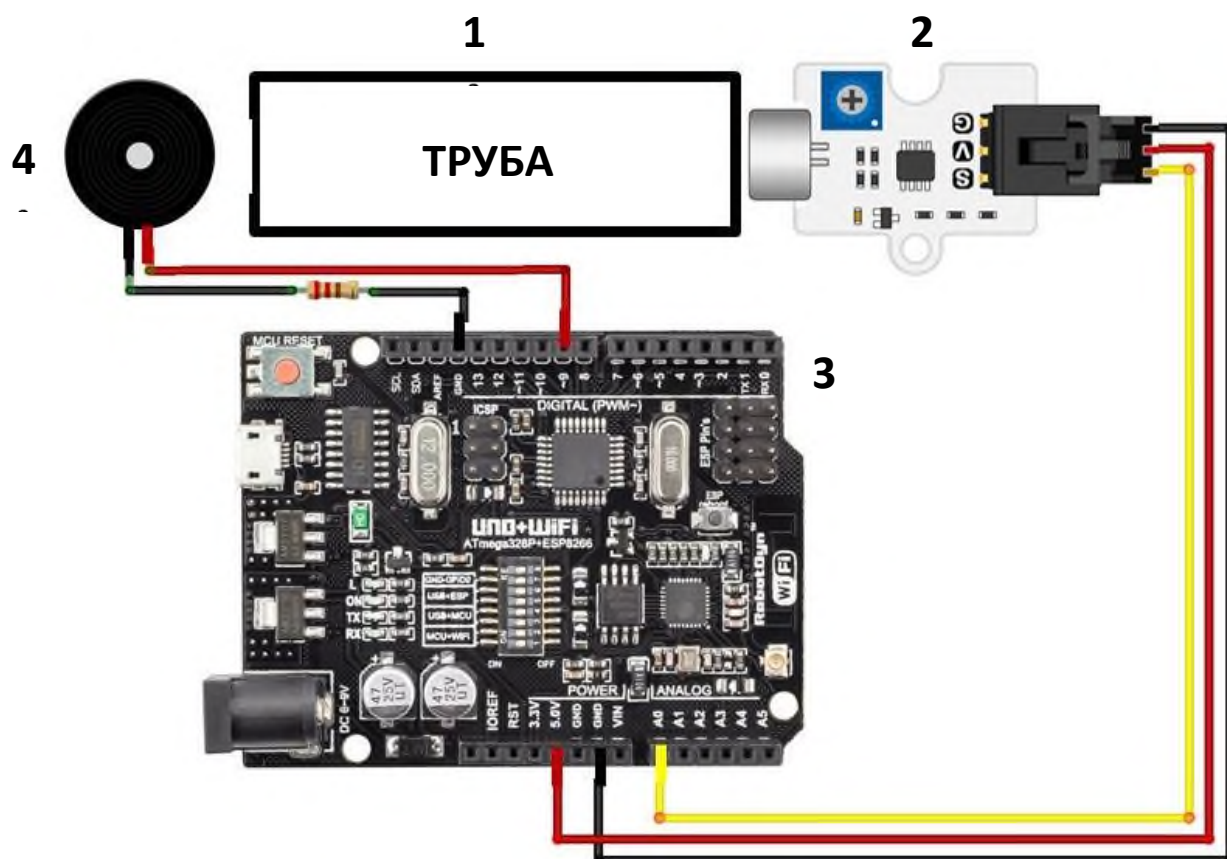


Рисунок – 4 Схема лабораторной установки

2.2 Используемые аппаратные компоненты

2.2.1 Плата Arduino

Для создания лабораторной установки была использована печатная плата HWAYEN UNO R3 от китайского производителя. Это версия классической платы ARDUINO UNO R3, в которой реализована полная интеграция микроконтроллера Atmel ATmega328 и IC Wi-Fi ESP8266 с флэш-памятью 4 МБ и конвертером USB-TTL CH340G на одной плате. Функционально это эквивалентно Arduino UNO R3 с Wi-Fi Shield и с SD Card Module, но с намного более низкой ценой. В данном пункте, ввиду только упомянутых отличий платы HWAYEN UNO R3 от оригинала, будет рассмотрена плата Arduino UNO R3, а особенности работы с используемой платой будут рассмотрены при описании практической части работы.

Плата Arduino UNO R3 — открытая программно-аппаратная платформа для разработки «умной» электроники на базе микроконтроллера ATmega328p, работающего на частоте 16 МГц, обладающего постоянной памятью на 32 кБ и оперативной памятью 2 кБ (Таблица 1). Плата оснащена 14 контактами цифрового ввода-вывода (GPIO) с функционалом UART (Serial), ШИМ (PWM), внешнее прерывание (interrupt), 6 контактами аналогового входа (АЦП), контактами вывода питания на 3,3 В, 5 В и Vin (прямой вывод с поданного на плату питания), а так же контактами шины I2C и интерфейса ICSP. Все контактные колодки продублированы контактными площадками на плате. Также на плате установлены светодиодные индикаторы подачи питания, активности приема и передачи по UART и работы программы.

Таблица – 1 Характеристики платы Arduino Uno

Микроконтроллер	ATmega328p
Рабочее напряжение	5 В
Входное напряжение (рекомендуемое)	7–12 В
Входное напряжение (предельное)	6–20 В
Цифровые входы/выходы	14 (6 из которых могут использоваться как выходы ШИМ)
Аналоговые входы	6
Постоянный ток через вход/выход	40 мА
Постоянный ток для вывода 3,3 В	50 мА
Флеш-память	32 Кбайт, при этом 0,5 Кбайт используются для загрузчика
ОЗУ	2 Кбайт
EEPROM	1 Кбайт
Тактовая частота	16 МГц

Arduino Uno может быть запитан от USB либо от внешнего источника питания - тип источника выбирается автоматически [30]. В качестве внешнего источника питания (не USB) может использоваться сетевой AC/DC-адаптер или аккумулятор/батарея. Штекер адаптера (диаметр – 2,1 мм, центральный контакт - положительный) необходимо вставить в соответствующий разъем питания на плате. В случае питания от аккумулятора/батареи, ее провода необходимо подсоединить к выводам Gnd и Vin разъема POWER.

Напряжение внешнего источника питания может быть в пределах от 6 до 20 В. Однако, уменьшение напряжения питания ниже 7 В приводит к уменьшению напряжения на выводе 5V, что может стать причиной нестабильной работы устройства. Использование напряжения больше 12 В может приводить к перегреву стабилизатора напряжения и выходу платы из строя. С учетом этого, рекомендуется использовать источник питания с напряжением в диапазоне от 7 до 12В.

Ниже перечислены выводы питания, расположенные на плате:

VIN. Напряжение, поступающее в Arduino непосредственно от внешнего источника питания (не связано с 5 В от USB или другим стабилизированным напряжением). Через этот вывод можно как подавать внешнее питание, так и потреблять ток, когда устройство запитано от внешнего адаптера.

5V. На вывод поступает напряжение 5 В от стабилизатора напряжения на плате, вне зависимости от того, как запитано устройство: от адаптера (7 – 12 В), от USB (5 В) или через вывод VIN (7 – 12 В). Запитывать устройство через выводы 5V или 3V3 не рекомендуется, поскольку в этом случае не используется стабилизатор напряжения, что может привести к выходу платы из строя.

3V3. 3,3 В, поступающие от стабилизатора напряжения на плате. Максимальный ток, потребляемый от этого вывода, составляет 50 мА.

GND. Выводы земли.

IOREF. Этот вывод предоставляет платам расширения информацию о рабочем напряжении микроконтроллера Arduino. В зависимости от напряжения, считанного с вывода IOREF, плата расширения может переключиться на соответствующий источник питания либо задействовать преобразователи уровней, что позволит ей работать как с 5 В, так и с 3,3В-устройствами.

2.2.2 Датчик звука

В данной работе в качестве датчика звука используется модуль OSTOPUS EF04008 (может быть использован любой другой Arduino-совместимый модуль). Датчик срабатывает при звуках, громкостью выше установленного порога. Модуль датчика звука применяется в приборах, содержащих аналоговую или цифровую электронику, в том числе и электронику на базе микроконтроллеров.

Модуль OSTOPUS EF04008 содержит сам датчик в виде электретного микрофона, которому необходимо питание от 3 до 10 вольт, индикаторы питания, индикатор срабатывания и микросхему компаратора, с помощью которого происходит определение превышения порога громкости звука.

Настройка порога срабатывания выполняется переменным резистором (потенциометром), также находящимся на плате модуля. Если порог громкости звука превышен, то на выходе «A0», к которому подключается вывод «S» модуля (см. разд. 2.1.), появляется высокий уровень напряжения, который соответствует уровню громкости окружающих шумов.

2.2.3 Пьезоизлучатель звука

В качестве источника звука в данной работе используется пассивный пьезоизлучатель звука или пьезодинамик. Пьезодинамик переводит переменное напряжение в колебание мембраны, которая в свою очередь создаёт звуковую волну. Иначе говоря, пьезодинамик — это конденсатор,

который звучит при зарядке и разрядке. Частоту звука можно регулировать, задавая соответствующие параметры программно.

2.3 Используемые программные средства

В данном разделе приводятся используемые в работе программные средства и их краткое описание. Программистская работа заключается в написании программ для модулей Arduino и разработке сайта для дистанционной работы с установкой.

Программирование платы HWAYEN UNO R3 производится через последовательный интерфейс (Serial, UART) посредством встроенного конвертера USB-TTL через разъем USB Type C с использованием специальной среды разработки Arduino IDE на языке Wiring (упрощенная версия C++ для работы с микроконтроллерами) [31].

Для Arduino существует множество программных библиотек позволяющих реализовать самый различный функционал и взаимодействовать с внешней электроникой (датчики, сернсоры, моторы, реле, шины передачи данных и много другое). Получить дополнительную информацию по работе с платой можно на сайте производителя <https://www.arduino.cc/en/Guide/ArduinoUno> (на английском языке) или на множестве справочных сайтов и сайтов с примерами программ, например <http://arduino.ru/Reference>.

В начале программы, как и в программах на C задаются переменные.

Программа, написанная в среде Arduino, называется «скетч» и состоит из двух функций, `setup()` и `loop()`.

`setup()` — функция `setup()` автоматически выполняется в самом начале программы Arduino. Внутри этой функции инициализируют переменные, пины (выводы платы) и любые библиотеки, которые используются в программе. Функция `setup()` запускается только один раз во время выполнения скетча Arduino, прямо при включении питания или сбросе.

loop() — функция loop() содержит основную часть программы. Эта функция выполняется после завершения setup(). Arduino будет выполнять команды внутри цикла снова и снова, пока плата не выключится.

После написания программы ее необходимо скомпилировать и загрузить на устройство с помощью среды разработки Arduino. После успешной загрузки устройство нужно перезагрузить.

Разработка сайта осуществлялась в редакторе кода Visual Studio Code с помощью технологий HTML, CSS и JavaScript специально для выбранной лаборатории удалённого доступа. Также используется облачный сервис Firebase, который предоставляет базу данных реального времени и позволяет обходиться без фактического написания серверной части.

2.3.1 База данных Firebase Realtime

База данных Firebase Realtime - это облачная NoSQL база данных компании Firebase. Данные хранятся в формате JSON и синхронизируются в режиме реального времени с каждым подключенным клиентом. Поддержаны особенности интеграции с приложениями под операционные системы Android и iOS, реализовано API для приложений на JavaScript, Java, Objective-C и Node.js, также возможно работать напрямую с базой данных в стиле REST из ряда JavaScript-фреймворков, включая AngularJS, React, Vue.js, Ember.js и Backbone.js. Предусмотрено API для шифрования данных. При создании кроссплатформенных приложения с помощью SDK для iOS, Android и JavaScript, все клиенты совместно используют один экземпляр базы данных в реальном времени и автоматически получают обновления с самыми новыми данными [32].

Ключевые особенности Firebase Realtime:

- Работа в реальном времени. Вместо типичных HTTP-запросов база данных Firebase Realtime использует синхронизацию данных - каждый раз, когда данные меняются, любое подключенное устройство получает это обновление в течение миллисекунд.

- **Оффлайн работа.** База данных Firebase Realtime позволяет создавать многофункциональные приложения для совместной работы, обеспечивая безопасный доступ к базе данных непосредственно из кода на стороне клиента. Данные сохраняются локально, и даже в автономном режиме события в реальном времени продолжают срабатывать, предоставляя пользователю полноценный отклик. Когда устройство восстанавливает соединение, база данных реального времени синхронизирует локальные изменения данных с удаленными обновлениями, которые произошли, когда клиент находился в автономном режиме, автоматически объединяя любые изменения.

- **Доступность с устройств клиентов.** Доступ к базе данных Firebase Realtime можно получить непосредственно с мобильного устройства или веб-браузера; нет необходимости в сервере приложений.

- **Масштабирование по нескольким базам данных.** С помощью базы данных Firebase Realtime можно масштабировать потребности приложений в данных, разбивая данные на несколько экземпляров базы данных в одном проекте Firebase. Есть возможность оптимизировать аутентификацию с помощью Firebase Authentication для конкретного проекта и проверять подлинность пользователей во всех экземплярах базы данных. Управлять доступом к данным в каждой базе данных можно с помощью пользовательских правил базы данных Firebase Realtime для каждого экземпляра базы данных.

База данных реального времени предоставляет гибкий язык правил на основе выражений, называемый правилами безопасности баз данных в реальном времени Firebase, для определения того, как данные должны быть структурированы и когда данные могут считываться или записываться. При интеграции с Аутентификацией Firebase разработчики могут определить, кто имеет доступ к каким данным и как они могут получить к ним доступ.

База данных реального времени является базой данных NoSQL и, поэтому имеет иные виды оптимизации и функциональность по сравнению с

реляционной базой данных. API-интерфейс базы данных реального времени позволяет выполнять только те операции, которые могут быть выполнены быстро. Это позволяет производить обработку данных в реальном времени, обслуживая миллионы пользователей без ущерба для скорости отклика. В связи с этим важно продумать то, как пользователи должны получать доступ к данным, и затем соответствующим образом структурировать их [33].

2.4 Описание практической части

2.4.1 Программирование Arduino

Для того чтобы разобраться в особенностях программирования используемой платы HWAYEH UNO R3 рассмотрим ее структурную схему (рис. 5).

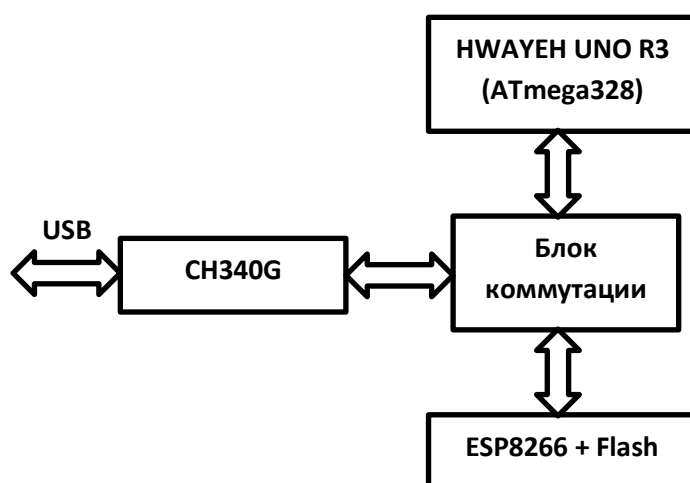


Рисунок –5 Структурная схема платы HWAYEH UNO R3

На одной плате интегрированы микропроцессорные модули Atmel ATmega328 и IC Wi-Fi ESP8266 с флэш-памятью 4 МБ и конвертер USB-TTL CH340G. Все модули могут работать вместе или независимо. Блок коммутации представляет собой набор из 8 DIP переключателей и чип USB порта CH340G, который может поочередно подключается к каждому из модулей. Схематическое представление возможных режимов работы платы изображено на рис. 6. Каждому режиму работы платы соответствует определенное положение переключателей (Таблица 2). Так, например, для того чтобы запрограммировать модуль ESP8266, необходимо выставить положения переключателей 5, 6 и 7 в положение «ON», а 1, 2, 3, 4 и 8 в положение «OFF». Аналогично делается для модуля ATmega328, только в этом случае в положение «ON» нужно выставить только 1-й и 2-й переключатели. Для того чтобы модули начали работать вместе необходимо

в очередной раз изменить положения переключателей и выставить их таким образом, чтобы два чипа были соединены между собой. При этом передача данных между модулями происходит через последовательный порт Serial (также известный как UART или USART).

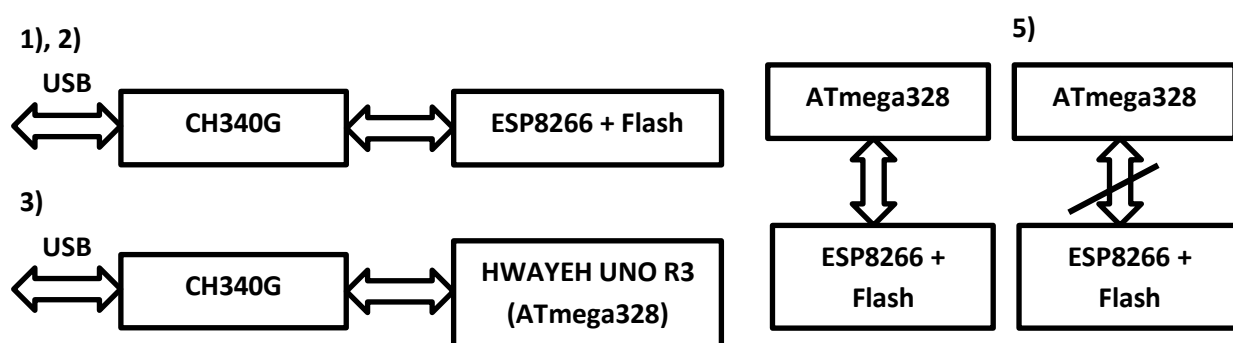


Рисунок 6 – Схематическое представление режимов работы HWAYEH UNO R3

Таблица – 2 Состояние переключателей и режимы работы платы

	Режим работы	1	2	3	4	5	6	7	8
1.	CH340 подключен к ESP8266 (загрузка скетча)	OFF	OFF	OFF	OFF	ON	ON	ON	NoUSE
2.	CH340 подключен к ESP8266 (отладка)	OFF	OFF	OFF	OFF	ON	ON	OFF	NoUSE
3.	CH340 подключен к ATmega328 (загрузка скетча, отладка)	OFF	OFF	ON	ON	OFF	OFF	OFF	NoUSE
4.	ATMega328+ESP8266	ON	ON	OFF	OFF	OFF	OFF	OFF	NoUSE
5.	Все модули работают независимо	OFF	OFF	OFF	OFF	OFF	OFF	OFF	NoUSE

Особенности использования данной платы:

1. Наличие двух микроконтроллеров, необходимость создания скетчей для каждого.
2. Для совместной работы модулей необходимо разработать протокол обмена данными между модулями.
3. Для программирования модулей необходимо каждый раз менять положение переключателей.
4. В режиме совместной работы модулей нет возможности через USB порт выводить отладочную информацию.

Работа по программированию модулей Arduino заключается в написании кода для основного модуля платы (ATmega328) и кода для WiFi модуля ESP8266.

1. Написание кода для основного модуля платы.

В данном скетче при получении численного значения частоты сигнала на порт основного модуля генерируется сигнал этой частоты, вычисляется относительная амплитуда сигнала и передается обратно.

В функции `setup()` с помощью функции `Serial.begin()` иницилируем последовательное соединение и задаем скорость передачи данных 9600 бит/с (бод).

В функции `loop()` с помощью конструкции `if` и условия `Serial.available()` проверяется наличие байт (символов) доступных для считывания из буфера последовательного порта. (Под символами понимаются данные, которые уже приняты и хранятся в последовательном приемном буфере, который может хранить максимум 64 байта). Далее, с помощью функции `Serial.readString()` считываем данные из буфера последовательного порта в строку, и, в зависимости от принятого значения, либо меняем частоту сигнала, генерируемого пьезодинамиком, либо подаем на вывод прямоугольный сигнал заданной частоты (с коэффициентом заполнения 50%) – функция `tone()`. С помощью функции `analogRead()` считываем значение с указанного

аналогового входа. Большинство плат Arduino имеют 6 каналов (8 каналов у платы Mini и Nano, 16 у Mega) с 10-битным аналого-цифровым преобразователем (АЦП). Напряжение, поданное на аналоговый вход, обычно от 0 до 5 В будет преобразовано в значение от 0 до 1023, это 1024 шага с разрешением 0,0049 В. Считывание значения с аналогового входа занимает примерно 100 микросекунд (0,0001 с), т.е. максимальная частота считывания приблизительно 10000 раз в секунду. Далее, ищем максимальное и минимальное значение амплитуды, вызываем функцию `noTone()`, которая прекращает генерирование прямоугольного сигнала после использования функции `tone()`. С помощью функции `Serial.println()` на порт посылаем строку символов – значение размаха амплитуды сигнала. Код скетча приведен в Приложении Б.

После написания программы ее необходимо скомпилировать и загрузить на устройство с помощью среды разработки Arduino IDE, выбрав в настройках плату «Arduino Uno». Далее, необходимо изменить положение DIP переключателей на плате, задав соответствующий режим работы. После успешной загрузки, устройство нужно перезагрузить.

2. Написание кода для WiFi модуля ESP8266.

Модуль ESP8266 программируется как простой веб-сервер, включающий обработчик событий Firebase. Для работы с API сервиса Firebase была использована Arduino библиотека Firebase-ESP8266 [34]. Кроме того, в скетче используются библиотеки: ESP8266WiFi, WiFiClient, ESP8266WebServer. Примеры скетчей для каждой из библиотек можно посмотреть, выбрав нужную библиотеку в пункте меню **Файл | Примеры в Arduino IDE** или на сайте разработчика.

В начале программы необходимо инициализировать переменные для установки связи по WiFi и с облачным сервисом Firebase (листинг 1).

```
#define FIREBASE_HOST
"YOUR_FIREBASE_PROJECT.firebaseio.com" //Without
http:// or https:// schemes
#define FIREBASE_AUTH
"YOUR_FIREBASE_DATABASE_SECRET"
#define WIFI_SSID "YOUR_WIFI_AP"
#define WIFI_PASSWORD "YOUR_WIFI_PASSWORD"
```

Листинг 1 – Задание переменных проекта

Значение переменных WIFI_SSID и WIFI_PASSWORD задаются в зависимости от локальной сети, в которой работает лабораторная установка.

Значения переменных FIREBASE_HOST и FIREBASE_AUTH неизменны и зависят от конкретного проекта Firebase. Их можно посмотреть в консоли разработчика Firebase.

Для работы по локальной сети скетч содержит следующие функции:

- is_authenticated() – проверка авторизации через Cookie.
- handleLogin() – для проверки введенных пользователем авторизационных данных.
- handleRoot() – функция для отправки страницы лаборатории на запрос пользователя.
- handleNotFound() – функция для ответа на запрос несуществующей страницы.
- getStrength() – функция для информирования пользователя о вводе неверных авторизационных данных.

В функции setup() осуществляется настройка сети WiFi и связь с облачным сервисом Firebase.

В функции loop() запускается веб-сервер и обработчики событий базы данных Firebase. Полный код скетча находится в GitHub репозитории по адресу: <https://github.com/CodeImm/ArduinoLab>.

После написания программы ее необходимо скомпилировать и загрузить на устройство с помощью среды разработки Arduino IDE, выбрав в настройках плату «Generic ESP8266 Module» и предварительно изменив

положение DIP переключателей на плате, задав соответствующий режим работы. После успешной загрузки, устройство нужно перезагрузить.

Т. к. программный код скетча выполняется последовательно и зациклен, работать с установкой можно через локальную сеть и облачный сервис одновременно. Однако, ввиду того, что сбор и передача данных сама по себе занимает некоторое время, есть некоторые ограничения по одновременному обращению к установке нескольких пользователей. Наиболее оптимальным представляется обеспечение доступа одному пользователю по сети интернет, и одному-двум пользователям по локальной сети или наоборот (большее количество пользователей может вызывать значительные задержки ответов по времени при последовательной обработке запросов).

Выводы по работе с платой HWAYEN UNO R3:

1. Очень удобная техническая реализация беспроводной связи.
2. Отладка и процесс написания кода очень неудобный и требует большого количества манипуляций с DIP переключателями.
3. Альтернативой данной платы может быть комплект из двух различных плат, что значительно менее удобно и дороже.

2.4.2 Разработка сайта

Для работы пользователя с установкой по сети Интернет в данной работе вместо веб-сервера используется облачный сервис Firebase. Взаимодействие пользователя с лабораторной установкой осуществляется через веб-интерфейс. Веб-интерфейс для функционирования АЛП УД должен, с одной стороны, обслуживать в интерактивном режиме диалог удалённого пользователя с облачным сервисом при настройке условий эксперимента, а с другой — реализовать заданный режим на лабораторной установке и трансляцию результатов его выполнения. Кроме того, разрабатываемый для данной АЛП УД сайт должен обеспечивать методическую поддержку лабораторной работы, т.е. содержать описание

лабораторного стенда, методику измерений, различные справочные материалы и т.п.

Созданный сайт АЛП УД включает в себя систему авторизации пользователей, реализованную с помощью встроенных служб Firebase, методическую поддержку лабораторной работы в виде отдельной страницы с методическим пособием и страницу с веб-интерфейсом для работы с лабораторной установкой и возможностью сохранять результаты работы. Загрузка сайта осуществляется с помощью стандартного Интернет-обозревателя по протоколу HTTP.

Взаимодействие с лабораторной установкой осуществляется через NoSQL базу данных (БД) реального времени Firebase с помощью ее слушателей. Асинхронный слушатель базы данных сообщает пользователю о хранящихся данных. Этот привязанный к списку ожидающих уведомлений слушатель активируется один раз в начале для первичного набора данных и потом срабатывает каждый раз при изменении данных.

Рассмотрим пример работы программного кода при снятии измерений с лабораторной установки (листинг 2). Когда пользователь вводит в специальное поле формы частоту сигнала и нажимает на кнопку генерации (HTML-элемент «addData»), срабатывает обработчик событий, в котором, при условии, что введенное пользователем значение частоты валидно (функция `valid()`), эта кнопка блокируется (функции `prop()` и `addClass()`). Далее, объявляется переменная «frequency», которой присваивается введенное значение из поля, объявляется переменная ссылки на узел БД «status», в котором хранятся атрибуты, характеризующие состояние установки (рис. 7). После чего, вызывается функция `update()` для обновления значения узла «frequency» на введенную частоту и «bpState» с `false` на `true`, что говорит о том, что необходимо сгенерировать сигнал на удаленной лабораторной установке. На установке выполняется слушатель узла «bpState», считывается значение частоты, генерируется сигнал, значение «bpState» в БД меняется на `false` и в БД записываются результаты измерения.

```
document.getElementById('addData').addEventListener('click', function () {
  if ($("#freq").valid()) {
    $("#addData").prop("disabled", true);
    $("#addData").addClass("button--disable");
    var frequency = $("#freq").val();
    const freqRef = firebase.database().ref('status');
    freqRef.update({
      frequency: frequency,
      bpState: true,
    });
  }
});
```

Листинг – 2 Обработчик событий, срабатывающий при нажатии на кнопку генерации сигнала



Рисунок – 7 Информация о состоянии лабораторной установки в базе данных Firebase

После того, как в узел БД записались результаты измерения, на сайте автоматически срабатывает слушатель, привязанный к этому узлу. Полученные данные разбираются и отображаются на странице. Кнопка генерации сигнала снова становится активной (листинг 3). Полный код проекта находится в GitHub репозитории по адресу: <https://github.com/CodeImm/ArduinoLabWebSite>.

```

firebase.auth().onAuthStateChanged(function (user) {
  if (user) {
    // User is signed in.
    var ref = firebase.database().ref('status/chartId');
    ref.once("value").then(function (snap) {
      if (snap.val() != "null") {
        var refStr = 'users/' + user.uid + '/charts/' + snap.val();
        var refChart = firebase.database().ref(refStr);
        refChart.on('child_added', snapshot => {
          var addedData = snapshot.val();
          config.data.datasets[0].data.push(parseInt(addedData.y));
          config.data.labels.push(parseInt(addedData.x));
          window.myLine.update();
          $("#addData").prop("disabled", false);
          $("#addData").removeClass("button--disable");
        });
      }
    });
  }
});

```

Листинг – 3 Слушатель узла БД, в который записываются результаты измерений

В результате работы были написаны и протестированы программы для модулей Arduino, входящих в лабораторную установку. Также разработан сайт для работы с установкой, который реализует следующий функционал:

- регистрация на сервисе;
- вход под своей учетной записью на сервис;
- методическая поддержка лабораторной работы;
- управление удаленной лабораторной установкой;
- сохранение результатов работы.

3 МЕТОДИКА ИСПОЛЬЗОВАНИЯ ЛАБОРАТОРИИ

3.1 Общие положения

При использовании АЛП УД через сеть Интернет предполагается, что учащийся перед проведением удалённого эксперимента принимал участие в более простых лабораторных работах, проводимых в своём образовательном учреждении, и имеет элементарные понятия о теме проводимого практикума, а также необходимые практические навыки для его проведения [35, 36, 37].

Для проведения лабораторного практикума учащиеся направляют заявки на выполнение лабораторной работы преподавателю. Чаще всего переписка для согласования времени проведения работы ведётся по электронной почте, но могут быть проведены и прямые аудиовизуальные переговоры с диспетчером (или системным программистом) объекта исследования, например, с использованием сред Zoom или Skype.

Сайт практикума находится по адресу <https://phys-labs.web.app/>.

Взаимодействие с установкой осуществляется с помощью веб-интерфейса через базу данных реального времени. Для работы с установкой сервер запрашивает авторизационные данные и регистрацию на сайте.

3.2 Взаимодействие с установкой

Для входа на сервер АЛП УД, учащемуся необходимо зарегистрировать свою учетную запись (рис. 8). После входа по созданной учетной записи, учащийся переходит на страницу с методическим пособием (рис. 9) для ознакомления с теоретической частью, описанием оборудования и методикой проведения эксперимента.

PhysLab Login Registration

Registration

Email

Password

Confirm Password

Send

Do you have account?
Enter Here

Рисунок 8 – Страница регистрации

Далее типовой сценарий проведения удалённого учебного эксперимента выглядит следующим образом. Учащийся нажимает на кнопку «Начать», осуществляя вход в монопольный режим управления удаленной лабораторной установкой, переходит на страницу с лабораторной работой (рис. 10), содержащую панель управления. Слева на странице находится рабочее поле, справа – поле ввода частоты сигнала и кнопка генерации сигнала. Внизу расположены кнопки для очистки графика, удаления последней отмеченной точки и кнопки для сохранения результатов работы в форматах CSV и PNG.

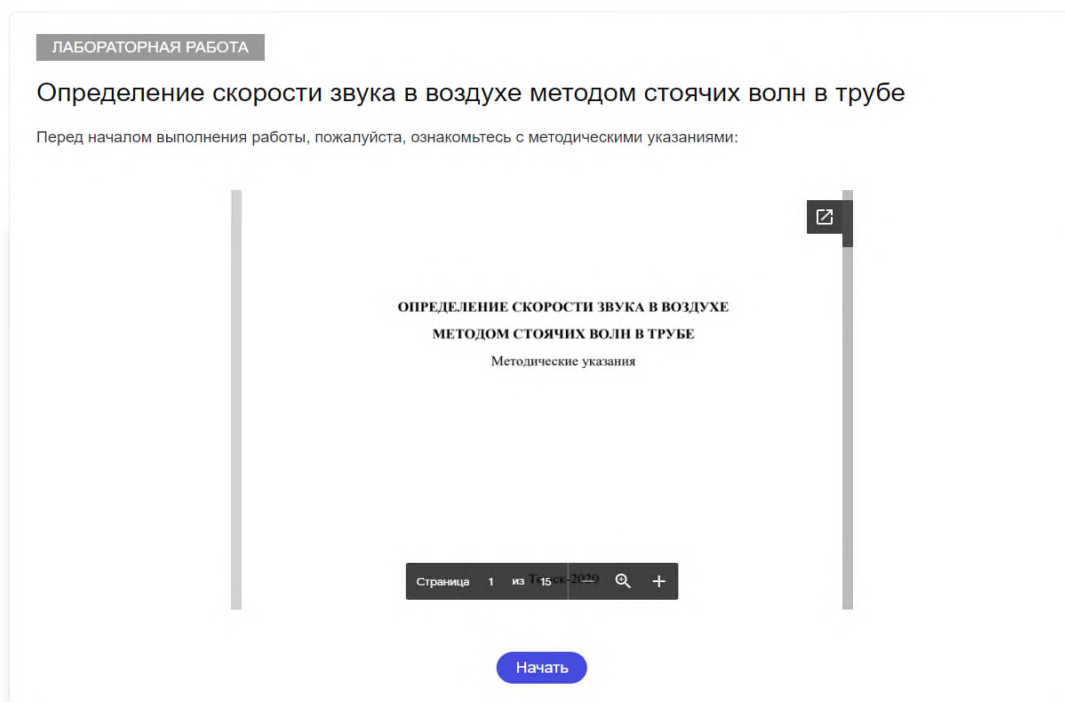


Рисунок 9 – Страница с методическим пособием

По окончании работы учащийся сохраняет полученные им результаты в табличном и/или графическом виде для обработки полученных данных и выходит из монопольного режима управления. Далее составляет отчет, сохраняет его в одном из форматов для хранения электронных документов и отправляет его преподавателю для проверки.

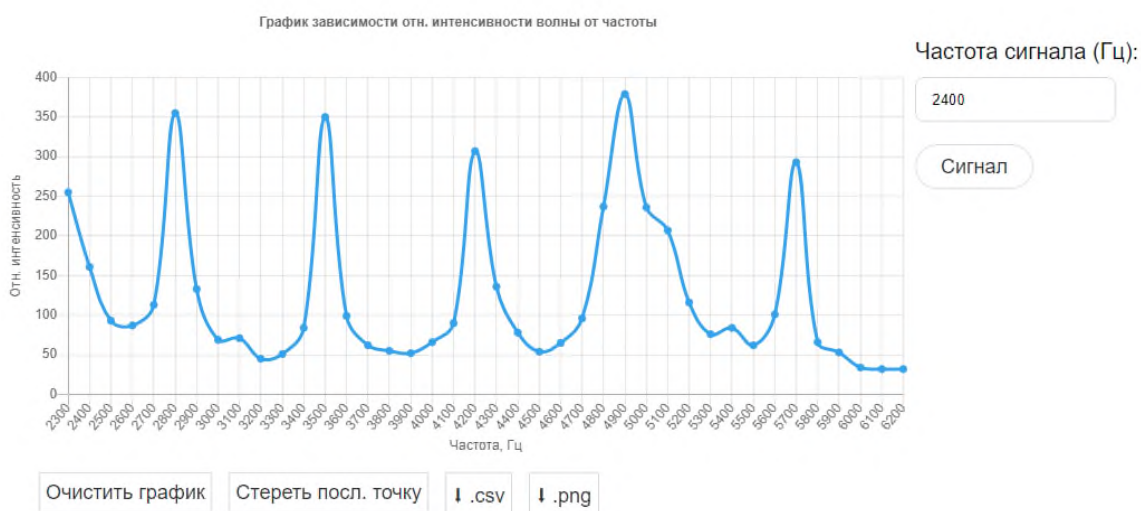


Рисунок 10 – Веб-интерфейс лабораторной работы с графиком

ЗАКЛЮЧЕНИЕ

В результате работы на основе обзора существующих решений построения лабораторных систем был создан бюджетный автоматизированный лабораторный практикум с удаленным доступом на базе платформы Arduino. Практикум включает в себя одну лабораторную работу для определения скорости звука в воздухе методом стоячих волн в трубе.

Была разработана схема удаленного доступа к лабораторной установке с использованием облачных технологий, требующая минимального количества ресурсов.

Были написаны и протестированы программы для модулей Arduino, входящих в лабораторную установку, разработан сайт для работы с установкой удаленно.

Созданный практикум на базе платформы Arduino был успешно апробирован в образовательном процессе на кафедре общей и экспериментальной физики ТГУ.

Использование практикума возможно не только в рамках дистанционного и открытого образования, но и при проведении традиционных очных учебных занятий.

Полученные результаты были представлены на XVII Всероссийской конференции Студенческих научно-исследовательских инкубаторов (СНИИ – 2020), которая состоялась 11-15 мая 2020 года, г. Томск. Публикация с участием автора «Акустический интерферометр Квинке и его использование для определения скорости звука» – [4].

В целом, семейство плат Arduino может быть довольно успешно использовано для решения различных задач дистанционного управления и измерений, которые не требуют производительности, превышающей возможности плат Arduino. Невысокая цена, низкое потребление, небольшие габариты, наличие всех необходимых функциональных возможностей делают Arduino и бренды подобных плат идеальными кандидатами для их применения в разработке бюджетных лабораторий по физике с удаленным доступом.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ

1. Журавлева О.Б., Крук Б.И. Соломина Е.Г. Управление Интернет-обучением в высшей школе// под ред. Б.И. Крука. – Новосибирск: «Веди», 2005. –255 с.
2. Норенков И.П. Системные вопросы дистанционного обучения // Информационные технологии. – 2001. - №3. – С.17-21.
3. Образование и XXI век: Информационные и коммуникационные технологии / Под ред. В.Г. Кинелева. – М.: Наука, 1999. – 191 с.
4. Горчаков Л.В., Колесников Д.А., Печерицин А.А. Акустический интерферометр Квинке и его использование для определения скорости звука//Проблемы учебного физического эксперимента. – Вып.31.-М.-ИСПО РАО, – 2020. – С.54-56.
5. Отраслевой стандарт 9.2-98. Системы автоматизированного лабораторного практикума. Основные положения. М.: Росстандарт, 1998.
6. Шауэрман А.А. Лаборатория с удаленным доступом как альтернатива обучающему моделированию [Электронный ресурс]. URL: <http://www.labfor.ru/articles/education/philosophy> (дата обращения: 10.03.2020).
7. Задков В.Н., Пономарев Ю.В. Компьютер в эксперименте: Архитектура и программные средства автоматизации. – М.: Наука, 1988. – 376 с.
8. Новый подход к инженерному образованию: теория и практика открытого доступа к распределенным информационным и техническим ресурсам / Ю.В. Арбузов, В.Н. Леньшин, С.И. Маслов и др. - М.: Центр-Пресс, 2000. - 238 с.
9. Сопряжение датчиков и устройств ввода данных с компьютерами IBM PC / Под ред. У. Томпкинса и Дж. Уэбстера. – М.: Мир, 1992. – 592 с.
10. С. В. Каменский, А. Б. Жуков Проектирование устройств сопряжения с объектом. – Новосибирск: НГТУ, 2006. - 24 с
11. Зимин А.М. Диагностика плазмы с удаленным доступом через сеть Интернет// Актуальные проблемы приложений физики низкотемпературной

плазмы: Матер. Шк. молодых ученых (2-9 сентября, 2003 г.) [Электронный ресурс]. URL: <http://plasma.karelia.ru/pub/fntfp/> (дата обращения: 10.03.2020).

12. Костин И. В., Алексеенко А. Е. Физический практикум на базе модуля ЦАП-АЦП, сопряженного с персональным компьютером // Вестник Балтийского федерального университета им. И. Канта, № 3, 2017. С. 74-83.

13. Сейтмуханова А. С. Использование современных информационно-измерительных технологий при совершенствовании комплексных лабораторных работ / А. С. Сейтмуханова // Информационно-измерительная техника и технологии : материалы V научно-практической конференции, Томск, 19-23 мая 2014 г. — Томск : Изд-во ТПУ, 2014. — С. 162-167.

14. Новый подход к инженерному образованию: теория и практика открытого доступа к распределенным информационным и техническим ресурсам / Ю.В. Арбузов, В.Н. Леньшин, С.И. Маслов и др. М.: Центр-Пресс, 2000; Зимин А.М. Лаборатории удаленного доступа в практической подготовке инженеров XXI века. М.: Изд-во МГТУ им. Н.Э. Баумана, 2007.

15. Выволокина Е.В. Разработка учебно-методического комплекса «Промышленные датчики расхода» / Е. В. Выволокина, В. А. Гольцев // Творческое наследие В. Е. Грум-Гржимайло: история, современное состояние, будущее: сборник докладов III Всероссийской научно-практической конференции студентов, аспирантов и молодых ученых с международным участием, посвященной 150-летию со дня рождения Владимира Ефимовича Грум-Гржимайло (Екатеринбург, 27–29 марта 2014 г.). — Екатеринбург: УрФУ, 2014. — Ч. 2. — С. 165-170.

16. Berntzen R., Strandman J.O., Fjeldly T.A., Shur M.S. Advanced Solutions for Performing Real Experiments over the Internet // International Conference on Engineering Education. – Oslo: Norway. – August 6-10, 2001. – P. 21-26.

17. Chen S.H., Chen R., Ramakrishnan S.Y.Hu, Zhuang Y., Ko C.C., Chen M. Development of Remote Laboratory Experimentation through Internet// Proceedings of the 1999 IEEE Hong Kong Symposium on Robotics and Control. – Hong Kong. – 1999 – P. 756-760.

18. Автоматизированный лабораторный практикум по механике деформируемого твердого тела с удаленным доступом / Б.В. Букеткин, О.В. Довыденко, А.М. Зимин и др. // Труды Всерос. науч. конф. «Научный сервис в сети Интернет». - М.: Изд-во МГУ, 2002. – С.117-119.
19. Интернет-лаборатория по механике деформируемого твердого тела / Б.В. Букеткин, О.В. Довыденко, А.М. Зимин и др. // Труды X Всерос. научн.-методич. конф. «Телематика'2003». СПб.: ГосНИИ «Информика», – 2003. – Т.2. – С.352-353.
20. Выдрин Д.Ф., Махнёва А.О., Мавлютов А.Р. Платформа Ардуино: преимущества // Academy – 2017. – № 1 (16). – С. 9–12.
21. Мартынюк А.С. Методические и технологические аспекты подготовки будущих учителей физики к использованию средств микроэлектроники в экспериментально-исследовательской работе //Фундаментальные исследования. – 2013. – №8. – С. 450-454.
22. Волков Е.В. Применение микроконтроллеров в составе установок для лабораторных работ по физике. - Режим доступа: <http://l.120-bal.ru/doc/17151/index.html> (дата обращения: 13.12.2016).
23. Колосков, С. Ю. Использование цифрового измерительного комплекса на базе платформы Arduino в лабораторном практикуме по физике / С.Ю. Колосков, М.И. Старовиков, И.В. Старовикова// Открытое и дистанционное образование. – 2017. – № 2. – С. 37-43.
24. Севостьянов, Р.А. ; Попов, А.П. Опыт создания стендов удаленного доступа для проведения школьных лабораторных работ по физике. Современные информационные технологии и ИТ-образование. – 2014. – С. 792-798.
25. Евдокимов Ю.К., Кирсанов А.Ю., Салахова А.Ш. Дистанционные автоматизированные учебные лаборатории и технологии дистанционного учебного эксперимента в техническом вузе // Открытое образование. – № 5, 2009. – С. 101-116

26. V.M. Cvjetkovi, M.Matijevi // Overview of architectures with Arduino boards as building blocks for data acquisition and control systems, 13 th International Conference on Remote Engineering and Virtual Instrumentation (REV), – 2016. [Электронный ресурс]. URL: <https://doi.org/10.3991/ijoe.v12i07.5818> (дата обращения: 10.03.2020).
27. Официальный сайт Arduino UNO [Электронный ресурс]. URL: <https://www.arduino.cc/en/Guide/ArduinoUno> (дата обращения: 10.03.2020).
28. Firebase-ESP8266 GitHub: [Электронный ресурс]. URL: <https://github.com/mobizt/Firebase-ESP8266> (дата обращения: 10.03.2020).
29. Савельев И.В. Курс общей физики : Т. 1. – М. : Наука, 1970. – 517 с.
30. Аппаратные характеристики Arduino [Электронный ресурс]. URL: <https://doc.arduino.ua/ru/hardware/Uno> (дата обращения: 10.03.2020).
31. Петин В.А. Проекты с использованием контроллера Arduino. 2-е изд., перераб. и доп. СПб.: БХВ-Петербург, 2015.
32. Firebase RealTime Database // Firebase Guides. [Электронный ресурс]. URL: <https://firebase.google.com/docs/database?hl=ru> (дата обращения: 11.05.2020)
33. Introduction // Firebase Guides. [Электронный ресурс]. URL: <https://firebase.google.com/docs/database> (дата обращения: 11.05.2020)
34. Firebase-ESP8266 GitHub: [Электронный ресурс]. URL: <https://github.com/mobizt/Firebase-ESP8266> (дата обращения: 10.03.2020).
35. Зимин А.М. Лаборатории удаленного доступа в практической подготовке инженеров XXI века. М.: Изд-во МГТУ им. Н.Э. Баумана, 2007
36. Лабораторный практикум с удалённым доступом как средство практической подготовки специалистов в техническом университете // И.Б. Федоров, А.М. Зимин, С.В. Коршунов, Ю.М. Кузнецов / Индустрия образования/ Под ред. А.А. Полякова, Ю.М. Кузнецова, Г.Ф. Филаретова, М.Б. Дружининой. М.: МГИУ, 2002. Вып. 2. С. 312–319

37. Зимин А.М., Маслов С.И. Практическая подготовка специалистов на основе технологии удалённого доступа к экспериментальным стендам через глобальную сеть // Открытое образование. 2009. № 5. С. 94–100.

ПРИЛОЖЕНИЕ А

За последние несколько лет значительно расширился ассортимент как Arduino-совместимых плат, выпускаемых сторонними разработчиками, так и плат на микроконтроллерах иной архитектуры (ESP8266, STM32). Несмотря на разную архитектуру, их программирование возможно через Arduino IDE. Но есть одна проблема, которая состоит в том, что сторонние платы отсутствуют в стандартной среде разработки Arduino. Добавить платы возможно как в автоматическом, так и в ручном режиме, после чего появляется возможность не только легко их программировать, но и использовать многие библиотеки Arduino. В данном приложении приводится инструкция к настройке Arduino IDE для возможности работы с микроконтроллерами ESP8266.

Менеджер плат — это инструмент Arduino IDE для добавления управляющих платформ с разными архитектурами (рис. А.1).

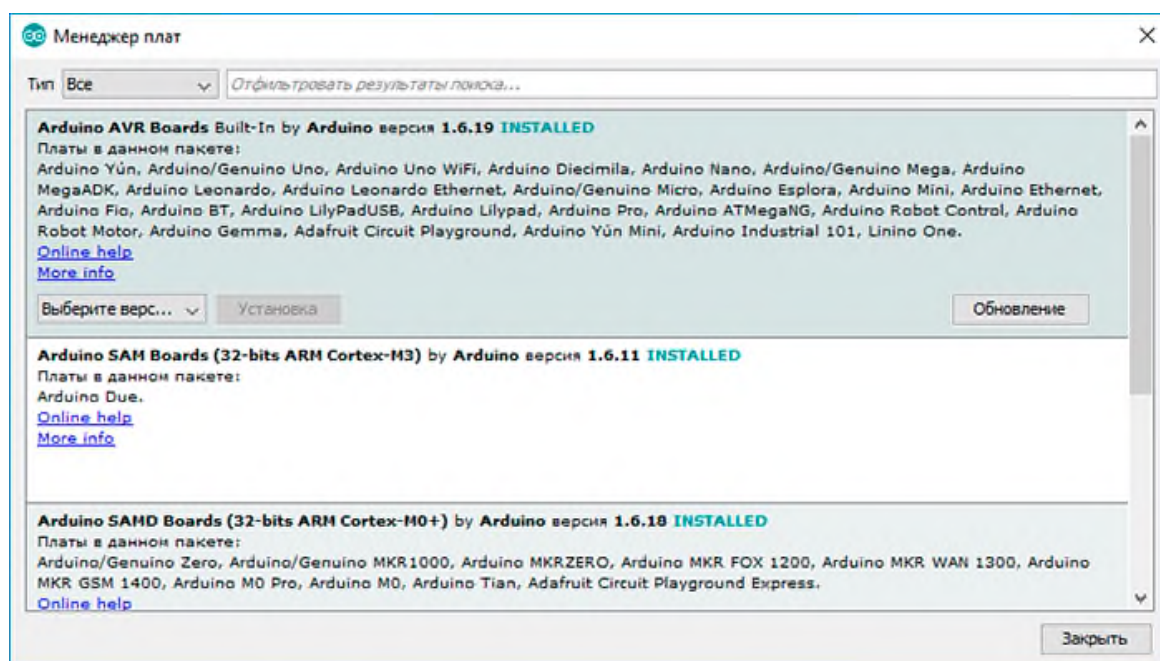


Рисунок А.1 – Менеджер плат Arduino IDE

Настройка

1. Открываем среду программирования Arduino IDE (рис. А.2).
2. Переходим в Настройки, для этого в пункте меню **Файл** выбираем **Настройки**.
3. В появившемся окне, в поле **Дополнительные ссылки для Менеджера плат (Additional Boards Manager URL)**, вводим адрес репозитория с необходимыми для платы пакетами и нажмиме «ОК» (рис. А.3). Стабильная версия библиотеки постоянно находится по ссылке: http://arduino.esp8266.com/stable/package_esp8266com_index.json (у других плат должна быть своя подобная ссылка, которую можно найти на сайте производителя или в документации)

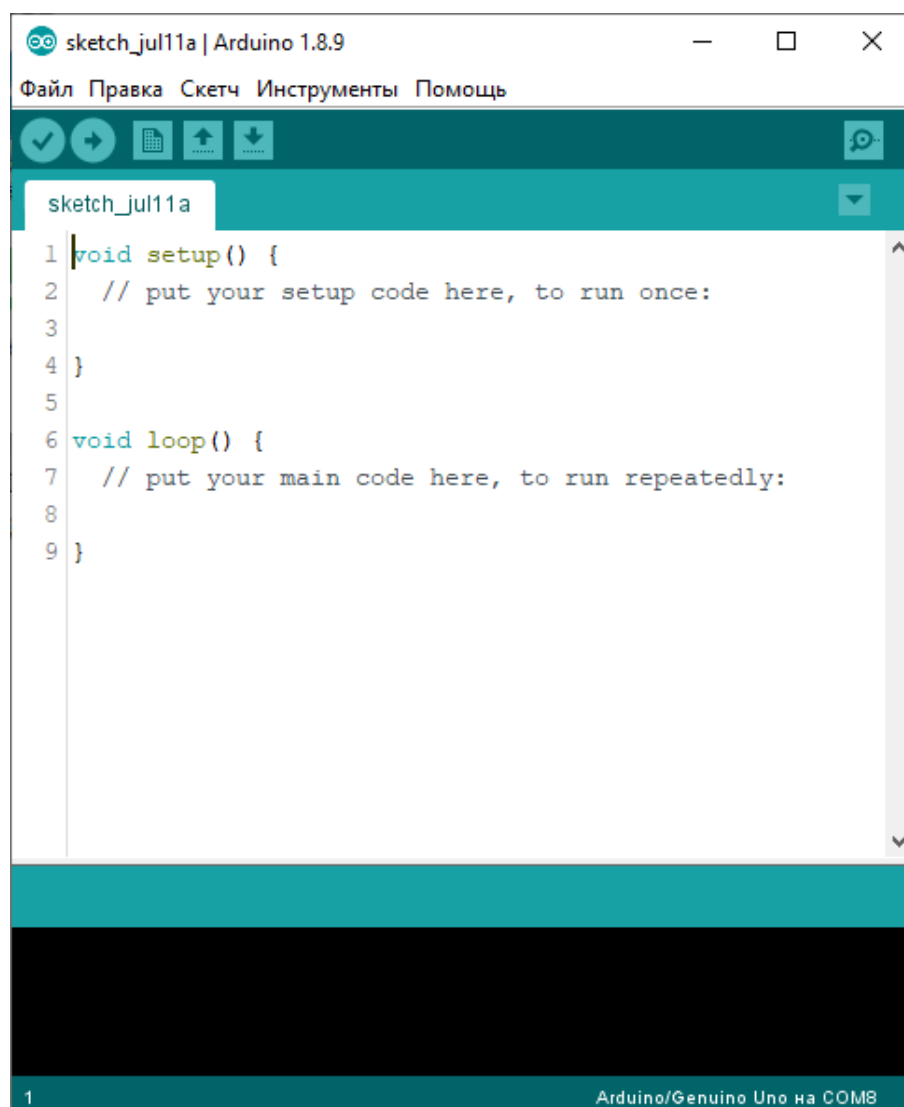


Рисунок А.2 – Среда Arduino IDE

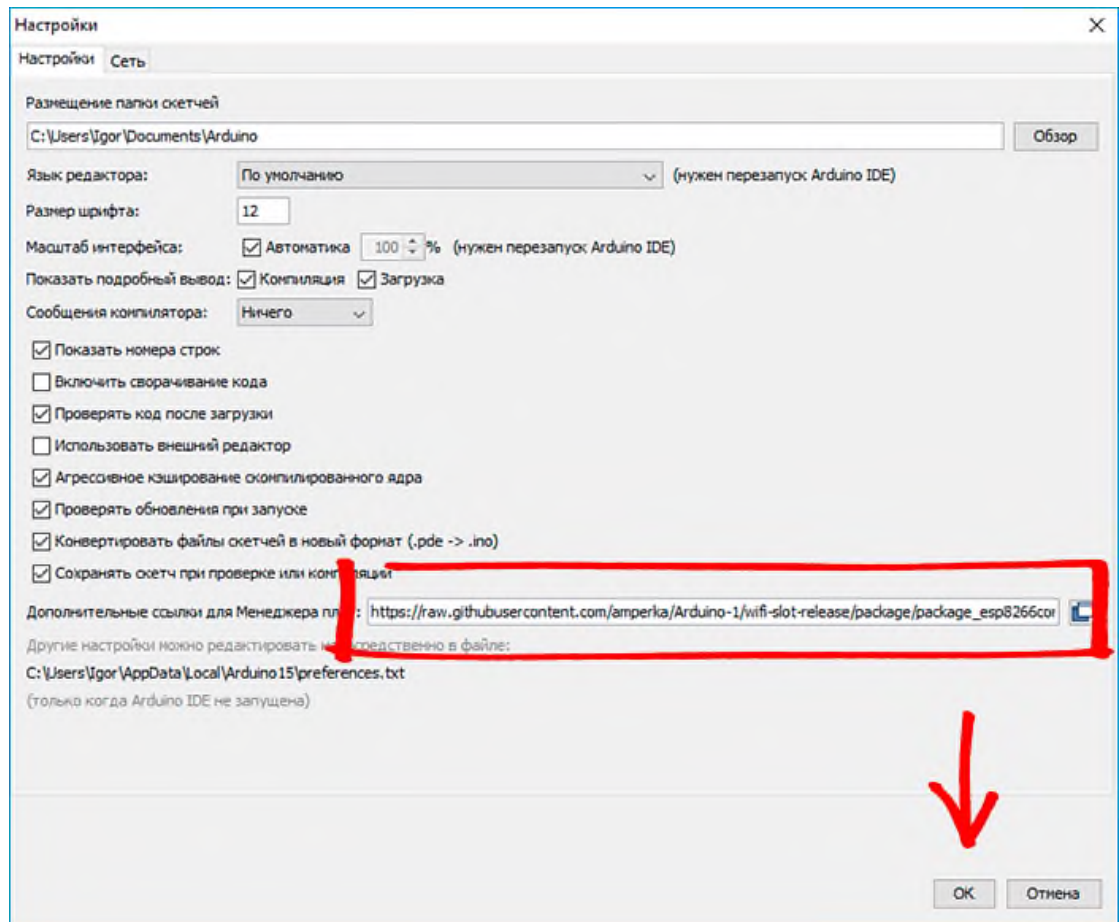


Рисунок А.3 – Окно настроек Arduino IDE

4. В пункте меню **Инструменты | Плата:** выбираем **Менеджер плат** (рис. А.4).

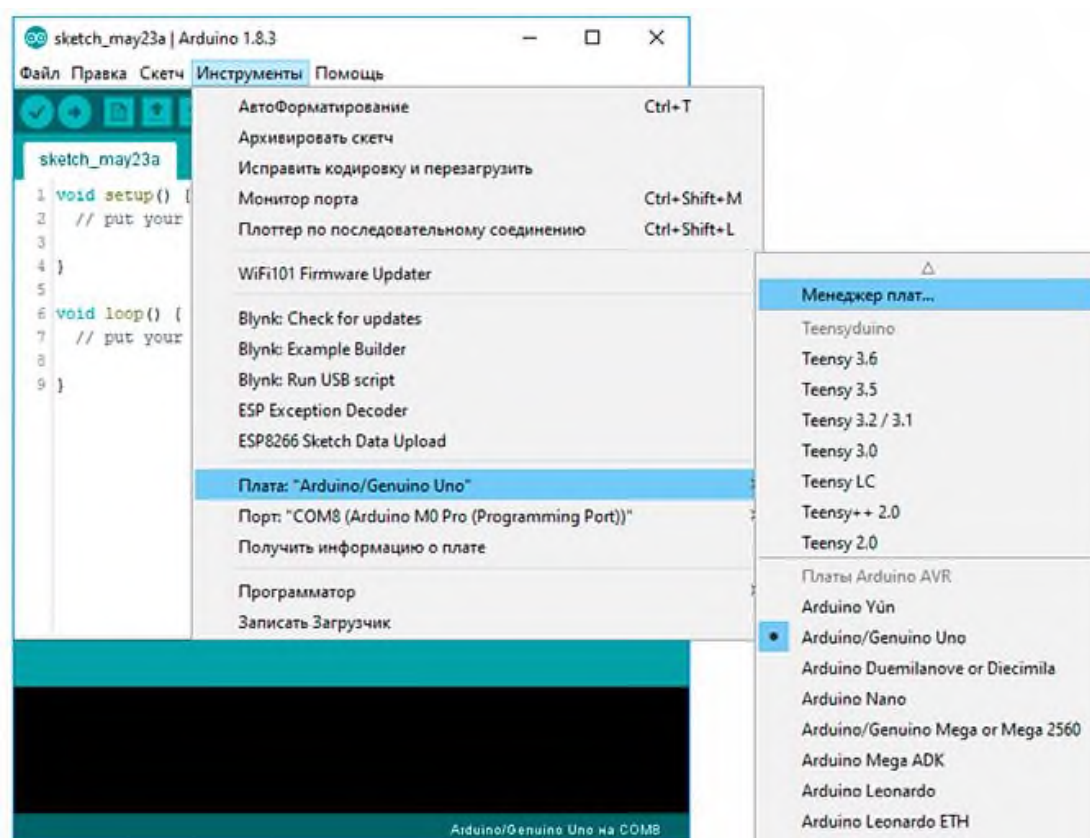


Рисунок А.4 – Открытие Менеджера плат

5. В списке менеджера плат находим плату «ESP8266», в правом нижнем углу выбираем необходимую версию библиотеки и нажимаем кнопку «Установка» (рис. А.5). Ждём окончания процесса установки необходимых пакетов. Надпись «INSTALLED» сообщает, что дополнения успешно установлены.

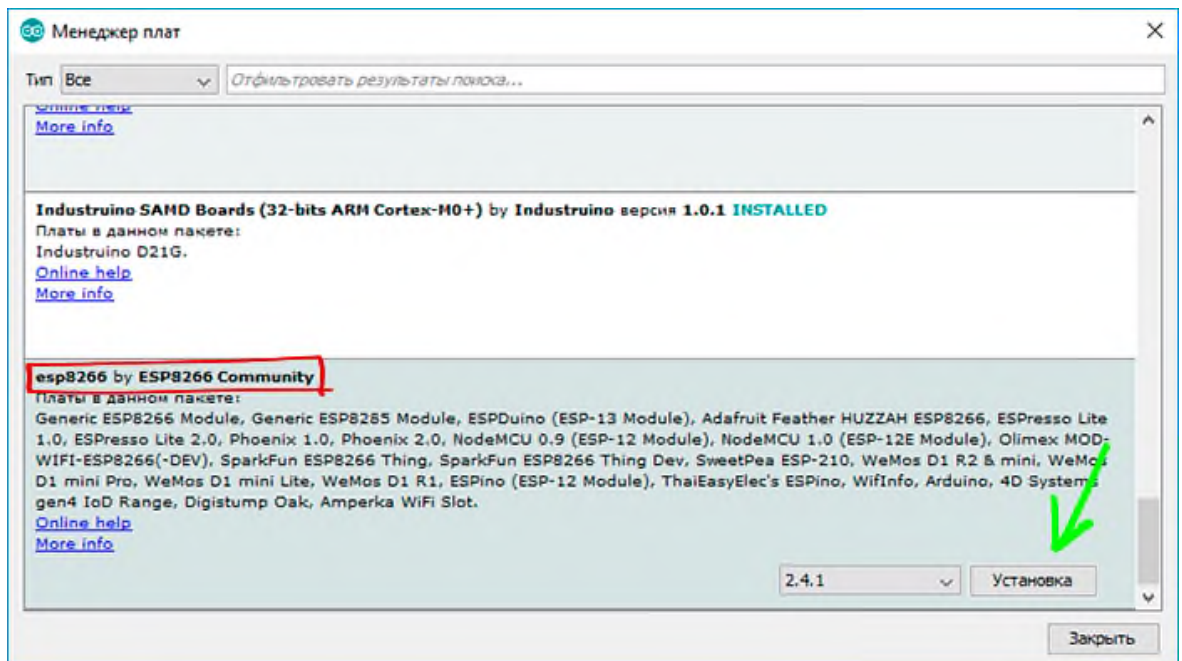


Рисунок А.5 – Библиотека esp8266 в списке Менеджера плат

6. Теперь нам доступны к программированию платформы с модулем ESP8266. Выбираем плату **Generic ESP8266 Module** (рис. А.6).

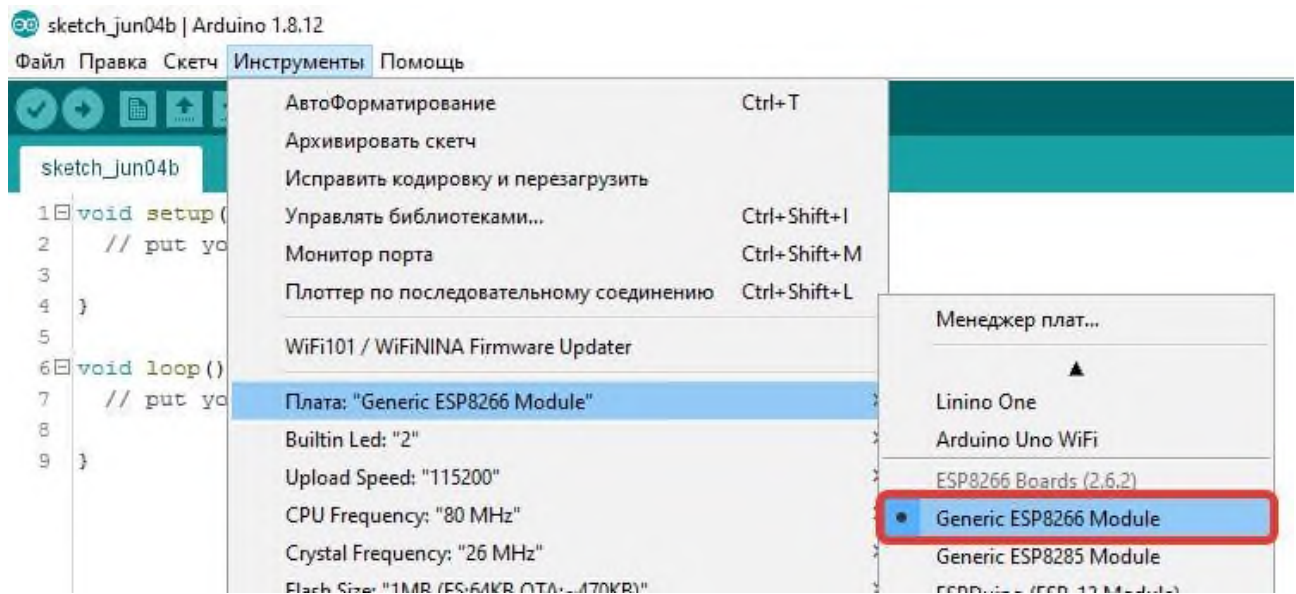


Рисунок А.6 – Выбор Arduino платы

Перед загрузкой скетча требуется задать необходимые параметры в меню **Инструменты | Upload Speed** и **Инструменты | Последовательный порт**. Здесь мы выбираем скорость загрузки **115200** и последовательный порт, к которому подключена плата.

После настройки IDE необходимо установить соответствующий режим работы платы с помощью DIP переключателей на самой плате.

ПРИЛОЖЕНИЕ Б

```
int microphonePin = A0;
const int SoundPin = 9;
int delaySound = 1000;
int microphoneValue = 0;
int val;
String inString = "";
int frequency = 5000;

void setup() {
    Serial.begin(9600);
}

void loop() {
    if (Serial.available()) { // если есть принятый
        СИМВОЛ, то читаем его
        inString = Serial.readString();
        val = inString.toInt();
        if (val > 32 && val < 10000) { // проверяем входит
            ли принятое значение частоты в допустимый диапазон
            // инициализируем переменные для определения
            размаха
            frequency = val;
            int mn = 1024;
            int mx = 0;
            tone(SoundPin, frequency); // запускаем генератор
            звука
            for (int i = 0; i < 1000; ++i) { // ищем макс и
            мин значения амплитуды
                int val = analogRead(microphonePin);
                mn = min(mn, val);
                mx = max(mx, val);
            }
            microphoneValue = mx - mn; // нашли размах
            амплитуды
            Serial.println(microphoneValue, DEC);
            delay(3000);
            noTone(SoundPin);
        }
    }
}
```


Отчет о проверке на заимствования №1



Автор: da.kolesnikov@mail.ru / ID: 5770935

Проверяющий: da.kolesnikov@mail.ru / ID: 5770935)

Отчет предоставлен сервисом «Антиплагиат»- <http://users.antiplagiat.ru>

ИНФОРМАЦИЯ О ДОКУМЕНТЕ

№ документа: 30
Начало загрузки: 16.06.2020 06:46:07
Длительность загрузки: 00:00:02
Имя исходного файла: Kolesnikov.pdf
Название документа: Kolesnikov
Размер текста: 1 кБ
Символов в тексте: 68016
Слов в тексте: 8238
Число предложений: 633

ИНФОРМАЦИЯ ОБ ОТЧЕТЕ

Последний готовый отчет (ред.)
Начало проверки: 16.06.2020 06:46:09
Длительность проверки: 00:00:02
Комментарии: не указано
Модули поиска: Модуль поиска Интернет



ЗАИМСТВОВАНИЯ

26,1%

САМОЦИТИРОВАНИЯ

0%

ЦИТИРОВАНИЯ

0%

ОРИГИНАЛЬНОСТЬ

73,9%

Заимствования — доля всех найденных текстовых пересечений, за исключением тех, которые система отнесла к цитированиям, по отношению к общему объему документа.
Самоцитирования — доля фрагментов текста проверяемого документа, совпадающий или почти совпадающий с фрагментом текста источника, автором или соавтором которого является автор проверяемого документа, по отношению к общему объему документа.

Цитирования — доля текстовых пересечений, которые не являются авторскими, но система посчитала их использование корректным, по отношению к общему объему документа. Сюда относятся оформленные по ГОСТу цитаты; общепотребительные выражения; фрагменты текста, найденные в источниках из коллекций нормативно-правовой документации.

Текстовое пересечение — фрагмент текста проверяемого документа, совпадающий или почти совпадающий с фрагментом текста источника.

Источник — документ, проиндексированный в системе и содержащийся в модуле поиска, по которому проводится проверка.

Оригинальность — доля фрагментов текста проверяемого документа, не обнаруженных ни в одном источнике, по которым шла проверка, по отношению к общему объему документа.

Заимствования, самоцитирования, цитирования и оригинальность являются отдельными показателями и в сумме дают 100%, что соответствует всему тексту проверяемого документа.

Обращаем Ваше внимание, что система находит текстовые пересечения проверяемого документа с проиндексированными в системе текстовыми источниками. При этом система является вспомогательным инструментом, определение корректности и правомерности заимствований или цитирований, а также авторства текстовых фрагментов проверяемого документа остается в компетенции проверяющего.

№	Доля в отчете	Доля в тексте	Источник	Ссылка	Актуален на	Модуль поиска	Блоков в отчете	Блоков в тексте
[01]	0%	8,04%	Пособие для системы повы...	http://rudocs.exdat.com	17 Ноя 2017	Модуль поиска Интернет	0	52
[02]	0%	8,04%	Пособие для системы повы...	http://rudocs.exdat.com	21 Фев 2019	Модуль поиска Интернет	0	52
[03]	6,39%	7,1%	Пособие для системы повы...	http://rudocs.exdat.com	25 Июн 2015	Модуль поиска Интернет	41	44

Еще источников: 17

Еще заимствований: 19,7%