

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое Акционерное Общество «Казахский Национальный Исследовательский
Технический Университет имени К.И.Сатпаева»



**SATBAYEV
UNIVERSITY**

Институт автоматики и информационных технологий

Кафедра «Робототехники и технических средств автоматики»

Кокумбеков Кенжебек Гизатуллаевич

Тема: «Разработка устройства регистрации и сбора данных»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
к дипломному проекту

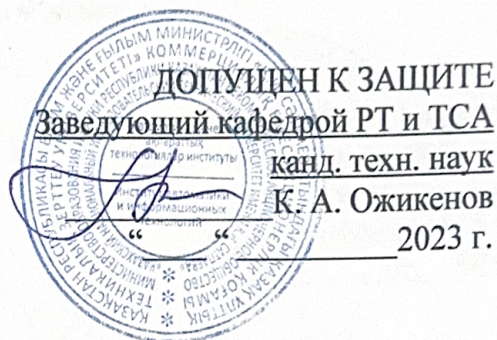
Специальность 6В07111 – Робототехника и мехатроника



SATBAYEV
UNIVERSITY

Институт автоматики и информационных технологий

Кафедра «Робототехники и технических средств автоматики»



ДИПЛОМНЫЙ ПРОЕКТ

На тему: « Разработка устройства регистрации и сбора данных»

по специальности 6В07111– Робототехника и мехатроника

Выполнил Кокумбеков К.Г.

Рецензент
PhD, ассоциированный профессор

Научный руководитель
к.т.н Ассоциированный
профессор

(ученая степень, звание)

 Балбаев Г. К.

 Тюлепбердинова Г.А

«26» май 2023 г.

«27» май 2023 г

Алматы 2023

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое Акционерное Общество «Казахский Национальный Исследовательский
Технический Университет имени К.И.Сатпаева»



SATBAYEV
UNIVERSITY

Институт автоматизации и информационных технологий

Кафедра «Робототехники и технических средств автоматизации»

6B07111 – Робототехника и мехатроника



ЗАДАНИЕ

на выполнение дипломного проекта

Обучающемуся Кокумбекову Кенжебеку Гизатуллаевичу

Тема: Разработка устройства регистрации и сбора данных

Утверждена приказом Ректора Университета № 408/п от «22» 11 2022 г

Срок сдачи законченной работы «30» мая 2023 г.

Исходные данные к дипломному проекту: Arduino Nano , Mega , HX711

Перечень подлежащих разработке вопросов в дипломном проекте:

а) изучение протокола i2c

б) анализ данных

в) изучение задачи

г) создание данных

Рекомендуемая основная литература: из 15 наименований 15

ГРАФИК подготовки дипломной работы (проекта)

Наименование разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю и консультантам	Примечание
Технологическая часть	13.03-03.04.2023 г.	Выполнено
Исследовательская часть	20.03-30.04.2023 г.	Выполнено
Практическая часть	24.03-28.04.2023 г.	Выполнено
Заключение	01.05-10.05.2023 г.	Выполнено

Подписи консультантов и нормконтролера на законченную дипломную работу (проект) с указанием относящихся к ним разделов работы (проекта)

Наименования разделов	Консультанты, И.О.Ф. (уч. степень, звание)	Дата подписания	Подпись
Нормоконтролер	Игембай Е.А. старший преподаватель	26.05.23	
Основная часть	Тюлепбердинова Г.А. Ассоциированный профессор	20.05.23	
Практическая часть	Тюлепбердинова Г.А. Ассоциированный профессор	21.05.23	

Научный руководитель

 Тюлепбердинова Г.А.

Задание принял к исполнению обучающийся

 Кокумбеков К.Г.

Дата

"29" май 2023г.

АННОТАЦИЯ

Цель дипломного проекта – разработка устройства регистрации и сбора данных. В работе рассматриваются основные принципы работы устройства, его функциональные возможности и технические характеристики. Описывается процесс проектирования и изготовления устройства, включая выбор компонентов, схемотехнику, разработку программного.

В работе рассматриваются возможности применения разработанного устройства в различных областях, в том числе в медицине, промышленности и других отраслях. Приводятся примеры конкретных задач, которые могут быть решены с помощью устройства.

В заключении делается вывод о том, что разработанное устройство является эффективным инструментом для регистрации и сбора данных, обладает широкими возможностями применения и может быть использовано в различных областях.

Кроме того, в работе также рассматриваются возможные пути дальнейшего улучшения устройства, в том числе с использованием новых технологий и материалов. Также описываются потенциальные проблемы и риски, которые могут возникнуть при использовании устройства, и предлагаются способы их предотвращения.

Важным аспектом работы является анализ конкурентных устройств на рынке и оценка конкурентоспособности разработанного устройства. В работе рассматриваются основные конкурентные преимущества и недостатки, а также потенциальные возможности для улучшения устройства и укрепления его позиций на рынке.

В целом, данная дипломная работа представляет собой исчерпывающий обзор разработки устройства регистрации и сбора данных, включая его функциональные возможности, технические характеристики, применение, конкурентоспособность и перспективы развития. Работа является важным вкладом в развитие современной техники и науки, а также может быть использована как руководство для будущих разработчиков устройств регистрации и сбора данных. В конструкторской части описана работа основных частей прототипа, составленная в виде макета. Рассчитаны параметры схем.

АҢДАТПА

Дипломдық жобаның мақсаты-деректерді тіркеу және жинау құрылғысын жасау. Жұмыста құрылғының негізгі принциптері, оның функционалдық мүмкіндіктері мен техникалық сипаттамалары қарастырылады. Құрылғыны жобалау және дайындау процесі сипатталады, оның ішінде компоненттерді тандау, схема, бағдарламалық жасақтама.

Жұмыста әзірленген құрылғыны әртүрлі салаларда, соның ішінде медицинада, өнеркәсіпте және басқа салаларда қолдану мүмкіндіктері қарастырылады. Құрылғының көмегімен шешуге болатын нақты тапсырмалардың мысалдары келтірілген.

Қорытындылай келе, әзірленген құрылғы деректерді тіркеу мен жинаудың тиімді құралы болып табылады, қолданудың кең мүмкіндіктері бар және оны әртүрлі салаларда қолдануға болады деген қорытындыға келеді.

Сонымен қатар, жұмыс сонымен қатар құрылғыны одан әрі жақсартудың мүмкін жолдарын, соның ішінде жаңа технологиялар мен материалдарды қолдануды қарастырады. Сондай-ақ, құрылғыны пайдалану кезінде туындауы мүмкін проблемалар мен қауіптер сипатталған және олардың алдын алу жолдары ұсынылған.

Жұмыстың маңызды аспектісі-нарықтағы бәсекеге қабілетті құрылғыларды талдау және дамыған құрылғының бәсекеге қабілеттілігін бағалау. Жұмыста негізгі бәсекелестік артықшылықтар мен кемшіліктер, сондай-ақ құрылғыны жақсарту және оның нарықтағы позициясын нығайту үшін әлеуетті мүмкіндіктер қарастырылады.

Тұтастай алғанда, бұл дипломдық жұмыс оның функционалдық мүмкіндіктерін, техникалық сипаттамаларын, қолданылуын, бәсекеге қабілеттілігін және даму перспективаларын қоса алғанда, тіркеу және деректерді жинау құрылғысын әзірлеуге жан-жақты шолу болып табылады. Жұмыс заманауи техника мен ғылымның дамуына маңызды үлес болып табылады және оны болашақ тіркеу және деректерді жинау құрылғыларын жасаушыларға нұсқаулық ретінде пайдалануға болады. Дизайн бөлімінде прототиптің негізгі бөліктерінің макет түрінде жасалған жұмысы сипатталған. Схемалардың параметрлері есептелген.

ANNOTATION

The purpose of the diploma project is to develop a device for recording and collecting data. The paper discusses the basic principles of the device, its functionality and technical characteristics. Describes the process of designing and manufacturing the device, including the selection of components, circuit design, software development.

The paper considers the possibilities of using the developed device in various fields, including medicine, industry and other industries. Examples of specific tasks that can be solved using the device are given.

In conclusion, it is concluded that the developed device is an effective tool for recording and collecting data, has wide application possibilities and can be used in various fields.

In addition, the paper also discusses possible ways to further improve the device, including using new technologies and materials. It also describes potential problems and risks that may arise when using the device, and suggests ways to prevent them.

An important aspect of the work is the analysis of competitive devices on the market and the assessment of the competitiveness of the developed device. The paper examines the main competitive advantages and disadvantages, as well as potential opportunities to improve the device and strengthen its position in the market.

In general, this thesis is an exhaustive review of the development of a device for recording and collecting data, including its functionality, technical characteristics, application, competitiveness and development prospects. The work is an important contribution to the development of modern technology and science, and can also be used as a guide for future developers of registration and data collection devices. The design part describes the work of the main parts of the prototype, compiled in the form of a layout. The parameters of the schemes are calculated.

СОДЕРЖАНИЕ

Введение	9
1 Теоретический анализ	10
1.1 Анализ существующих устройств регистрация и сбор данных на основе массы товаров	10
1.2 Алгоритм устройства регистрации и сбора данных	15
2 Конструкция оборудования	18
2.1 Формулировка созданного макета	18
2.2 Исполнительные механизмы	18
2.2.1 Аппаратная платформа Arduino Nano	19
2.2.2 Аналогово-цифровой преобразователь HX711	21
2.2.3 Тензометрический датчик веса	23
2.2.4 Схема подключения устройств к плате Arduino Nano	25
3 Программное обеспечение	27
3.1 Среда разработки Arduino IDE	27
3.2 Среда разработки PyCharm	28
Заключение	
Список использованных литературы	

ВВЕДЕНИЕ

Регистрация и сбор данных – важный процесс в многих областях, включая науку, технику, медицину, промышленность и многие другие. Современные технологии и развитие науки требуют постоянного развития устройств для сбора и анализа информации. Для выполнения этой задачи необходимо оборудование, которое может собирать и хранить данные в режиме реального времени. Целью данной дипломной работы является разработка устройства, предназначенного для регистрации и сбора данных о количестве медицинских товаров в режиме реального времени. Это дает нам возможность создавать простые и эффективные решения для учета товаров в аптеках, складах и других местах.

В данной работе мы будем использовать платформу Arduino для создания устройства, которое будет регистрировать и собирать данные товаров на основе их массы. Arduino - это открытая электронная платформа с открытым исходным кодом, которая позволяет создавать устройства и контроллеры на основе микроконтроллеров. Она очень популярна среди электронных инженеров и энтузиастов, так как позволяет создавать простые и эффективные решения для многих задач.

Программа для устройства будет написана на языке C++, в Arduino и будет работать следующим образом:

- Считывание данных с тензометрического датчика веса.
- Определение количество товаров на основе их массы.
- Отправка данных на компьютер через последовательный порт для дальнейшей обработки в учетной системе.
- Для регистрации и сбора данных мы будем использовать следующие компоненты:
 - Плата Arduino Nano.
 - Тензометрический датчик веса и АЦП HX711.
 - Дополнительный компонент для подключения к компьютеру.

1 Теоретический анализ

1.1 Анализ существующих устройств регистрация и сбор данных на основе массы товаров

Существуют различные устройства для регистрации и сбора данных на основе массы товаров, используемые в различных отраслях, таких как промышленность, логистика и торговля.

Мобильные устройства для взвешивания могут иметь различные характеристики в зависимости от конкретных потребностей. Вот некоторые из них:

Диапазон массы. Мобильные устройства для взвешивания могут иметь различные диапазоны массы, начиная от небольших весов, предназначенных для взвешивания легких предметов, до более крупных устройств, способных взвешивать грузы большой массы.

Размеры. Размеры мобильных устройств для взвешивания могут варьироваться от компактных портативных устройств до более крупных систем, в зависимости от предназначения и места использования. Некоторые устройства могут быть легко переносимыми, чтобы использовать их в разных местах, в то время как другие могут быть установлены на стационарных площадках.

Точность. Мобильные устройства для взвешивания могут иметь различные уровни точности в зависимости от требуемой точности взвешивания. В некоторых случаях может потребоваться высокая точность, особенно при взвешивании товаров, которые имеют небольшую массу.

Функциональность. Некоторые мобильные устройства для взвешивания могут иметь дополнительные функции, такие как подсчет количества взвешиваемых предметов, хранение данных, подключение к компьютеру или сети для передачи результатов взвешивания и другие функции, которые могут быть полезны в конкретной ситуации.

Ниже рассмотрены некоторые из наиболее распространенных устройств для регистрации и сбора данных на основе массы товаров:

1) **Весы с автоматическим взвешиванием:** это устройства, которые автоматически взвешивают товары и регистрируют их массу. Они могут быть использованы в различных отраслях, таких как производство, логистика и торговля. Весы с автоматическим взвешиванием могут быть как стационарными, так и портативными.



Рисунок – 1.1 Весы с автоматическим взвешиванием

2) Ленточные весы: это устройства, которые используются для взвешивания продуктов, которые движутся по конвейеру. Ленточные весы позволяют автоматизировать процесс взвешивания и уменьшить время на обработку товаров. Они широко используются в производственных и логистических отраслях.

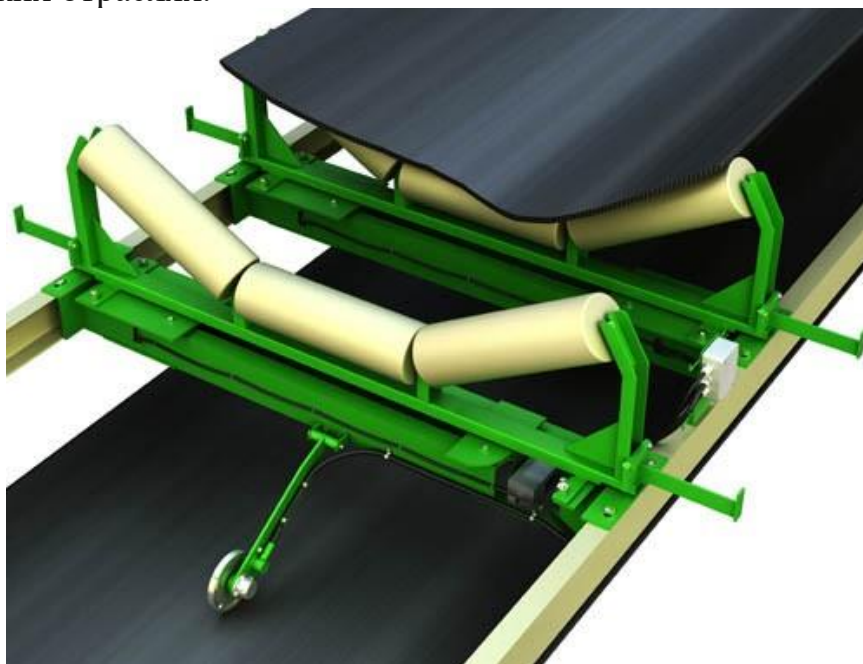


Рисунок – 1.2 Весы конвейерные (ленточные весы) СВЕДА ВК-230

3) Платформенные весы: это устройства, которые используются для взвешивания крупных товаров, таких как паллеты. Они могут быть использованы в промышленности, логистике и транспорте для взвешивания крупных грузов. Платформенные весы могут иметь различные размеры и грузоподъемность.



Рисунок – 1.3 Платформенные весы

4) Весовые сенсоры: это устройства, которые устанавливаются на различных объектах, таких как конвейеры, паллеты или транспортные средства, и позволяют регистрировать массу товаров, перемещаемых через них. Они могут быть использованы для автоматического сбора данных о массе товаров и контроля производственного процесса.

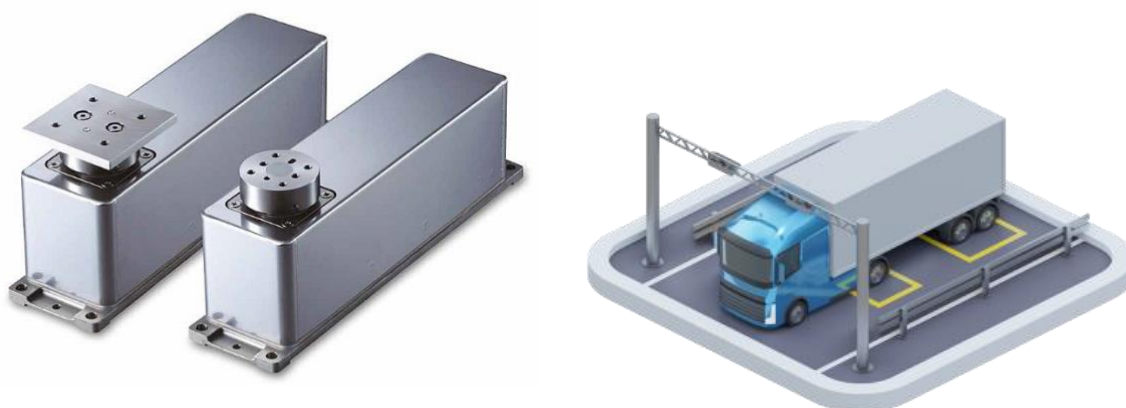


Рисунок – 1.4 Весовые сенсоры

5) Мобильные устройства для взвешивания: это устройства, которые могут быть использованы для взвешивания товаров на месте, например, на торговых точках или на производственных линиях. Они могут иметь различный диапазон массы, размеры и точность.



Рисунок – 1.5 Мобильные устройства для взвешивания

6) Устройства для автоматического сбора данных о массе товаров: это устройства, которые автоматически считывают данные о массе товаров и сохраняют их в базе данных. Они могут быть использованы для автоматической регистрации и сбора данных о массе товаров в различных отраслях, таких как логистика, производство и торговля. Устройства для автоматического сбора данных о массе товаров могут использоваться в сочетании с другими устройствами, такими как сканеры штрих-кодов и RFID-считыватели, для автоматического сбора данных о товарах и их идентификации.



Рисунок – 1.6 Устройства для автоматического сбора данных о массе товаров

7) Системы автоматического управления складом: это устройства, которые используются для автоматического управления складом и включают в себя различные компоненты, такие как весовые датчики, RFID-считыватели и системы управления запасами. Они могут быть использованы для автоматического сбора данных о массе товаров и управления их распределением на складе.



Рисунок – 1.7 Системы автоматического управления складом

8) Системы автоматической сортировки: это устройства, которые используются для автоматической сортировки товаров на основе их массы и других характеристик. Они могут быть использованы в логистике и транспорте для автоматической сортировки товаров в зависимости от их массы и назначения.



Рисунок – 1.8 Системы автоматической сортировки

1.2 Алгоритм устройства регистрации и сбора данных

Алгоритм проекта - это последовательность действий, которые необходимо выполнить для достижения цели проекта. Он описывает, какие шаги нужно предпринять, какие ресурсы использовать и как оценивать результаты, чтобы успешно завершить проект.

Чтобы разработать алгоритм устройства регистрации, нам необходимо определить, какие данные вы хотите регистрировать и каким образом вы планируете их получать. Вот общий шаг за шагом алгоритм для устройства регистрации:

Определить цель регистрации. Что именно вы хотите зарегистрировать? Например, это может быть информация о пользователях, событиях, медицинских препаратах или других типах данных.

Определить источник данных. Где и как вы будете получать данные для регистрации? Это может быть ввод сенсоров, ввод пользователя или данные из другого источника.

Установить параметры регистрации. Определить, какие параметры данных мы хотим регистрировать. Например, если мы регистрируем данные пользователя, мы можем сохранить имя, адрес, адрес электронной почты и так далее.

Создать структуру данных для хранения регистрационной информации. Определить, как мы будем хранить зарегистрированные данные. Это может быть база данных, файловая система или другое хранилище данных.

Разработать алгоритм регистрации. Определить последовательность действий, которую устройство будет выполнять для регистрации данных. Это может включать ввод данных, проверку их корректности, сохранение в структуру данных и любые другие необходимые операции.

Обрабатывать и сохранять данные: После получения данных выполнить необходимую обработку.

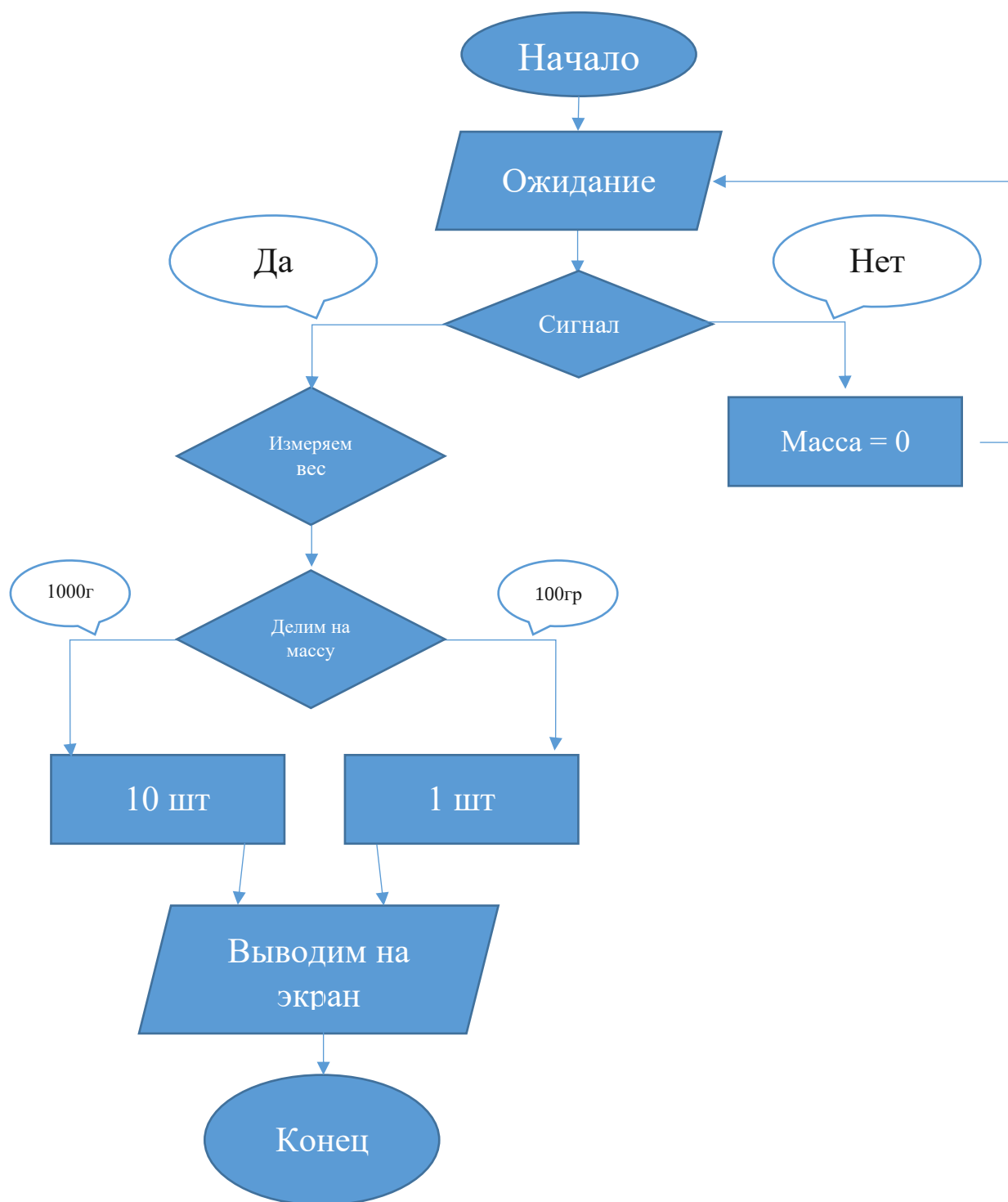


Рисунок – 1.9 Алгоритм устройства регистрации и сбора данных

Задача этого алгоритма - облегчить и ускорить процесс подсчета количества медицинских товаров, основываясь на их физических характеристиках. Это может быть особенно полезно в медицинских учреждениях, аптеках или других местах, где важно точно определить количество товаров без необходимости их отдельного пересчета.

Прежде чем перейти к алгоритму, необходимо иметь две величины: массу одного медицинского товара и массу его упаковки. Предположим, у нас

есть медицинские товары, каждый из которых имеет массу X грамм и упаковку с массой Y грамм.

Вот как работает алгоритм:

- Ввод данных. Пользователь вводит значения массы медицинского товара (X) и массы его упаковки (Y) в граммах.

- Расчет количества товаров. Алгоритм использует формулу для определения количества медицинских товаров (N) на основе их массы и массы упаковки: $N = (X + Y) / X$

- Вывод результата. Алгоритм выводит полученное значение количества товаров (N). Это число представляет собой приближенную оценку количества медицинских товаров на основе их массы и массы упаковки.

Важно отметить, что этот алгоритм предполагает, что масса упаковки однородна для всех медицинских товаров. Если упаковки имеют разную массу, то результат может быть несколько неточным. Также стоит отметить, что алгоритм не учитывает другие факторы, такие как объем или форма медицинского товара.

Для определения количества товаров на основе их массы необходимо знать массу каждого товара и его упаковки, а также массу упаковки без товара. Если известна масса одного товара и его упаковки, можно разделить общую массу на массу одного товара с упаковкой, чтобы определить количество товаров. Например, если общая масса 1000 грамм, а масса одного товара в упаковке составляет 100 грамм, то количество товаров равно 10.

2 Конструкция оборудования

2.1 Формулировка созданного макета

Созданный макет представляет собой устройство регистрации и сбора данных на основе микроконтроллера Arduino, тензометрического датчика веса и HX711. Устройство позволяет измерять вес товаров и собирать данные о весе для дальнейшей обработки и анализа. Оно имеет возможность передачи данных на компьютер или другое устройство через интерфейс USB. Макет является прототипом готового устройства и может быть доработан или модифицирован в соответствии с конкретными потребностями и требованиями проекта.

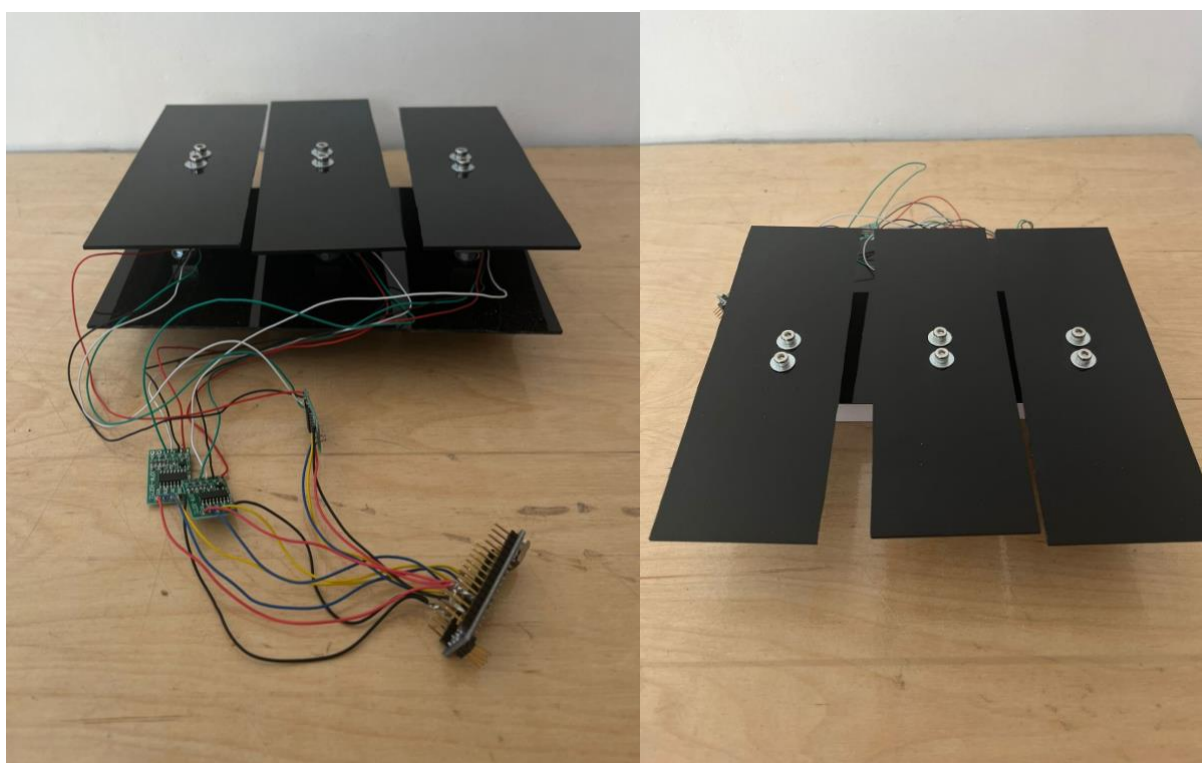


Рисунок – 2.1 Созданный макет устройства регистрации и сбора данных

2.2 Исполнительные механизмы

Ардуино Нано - это компактная версия платформы Arduino, которая предоставляет возможность программирования и управления электронными компонентами. Она идеально подходит для проектов, где ограничено пространство, но требуется мощность и гибкость Arduino. Ардуино Нано обладает достаточной производительностью и функциональностью для наших нужд.

HX711 - это высокоточный 24-битный аналого-цифровой преобразователь (АЦП), который специально разработан для измерения веса с

использованием тензодатчиков. Он может работать с различными типами тензодатчиков, включая полумостовые и мостовые конфигурации. HX711 обеспечивает высокую точность и стабильность измерений в широком диапазоне весов.

Тензодатчик веса - это датчик, который изменяет свое сопротивление или напряжение при деформации. Он состоит из тонкой пластины, на которой находятся тензорезисторы. Когда на тензодатчик действует вес объекта, пластина деформируется, что приводит к изменению электрических характеристик тензорезисторов. Измеряя эти изменения, мы можем определить вес объекта.

Теперь рассмотрим, как все это работает вместе. Ардуино Нано выполняет роль микроконтроллера, который управляет процессом сбора и обработки данных. HX711 подключается к Ардуино Нано через интерфейс SPI (Serial Peripheral Interface) и обеспечивает прецизионное измерение веса с использованием тензодатчика.

Подключение тензодатчика к HX711 довольно простое. Тензодатчик имеет четыре вывода: VCC (питание), GND (земля), OUT+ и OUT-. VCC и GND подключаются к соответствующим пинам на Ардуино Нано для питания. OUT+ и OUT- подключаются к соответствующим пинам на HX711.

HX711, в свою очередь, подключается к Ардуино Нано следующим образом: VCC к питанию (обычно 5 В), GND к земле, DT (Data) к пину данных Ардуино, SCK (Clock) к пину тактирования Ардуино. Для управления HX711 мы используем библиотеку HX711, которую можно легко найти и загрузить из интернета.

После настройки соединений мы можем начать сбор и обработку данных. Ардуино Нано отправляет команды HX711 для начала измерений, а затем получает результаты измерений. Мы можем использовать Ардуино для дополнительной обработки данных, например, для калибровки или вывода на дисплей.

Преимущество такой системы состоит в ее гибкости и расширяемости. Мы можем использовать Ардуино для добавления дополнительных функций, таких как сохранение данных, отправка результатов через интернет или управление другими исполнительными механизмами.

2.2.1 Аппаратная платформа Arduino Nano

Arduino Nano - это микроконтроллерная плата, созданная на базе Atmel AVR микроконтроллера ATmega328P. Она является одной из самых компактных и удобных для использования плат Arduino, поскольку имеет малый размер (18 мм x 45 мм) и множество встроенных возможностей.

На плате Arduino Nano имеются 14 цифровых входов/выходов (6 из которых могут быть использованы как ШИМ-выходы), 8 аналоговых входов, возможность работы с UART, I2C и SPI интерфейсами, встроенный кварц 16 МГц, USB-интерфейс и многое другое.

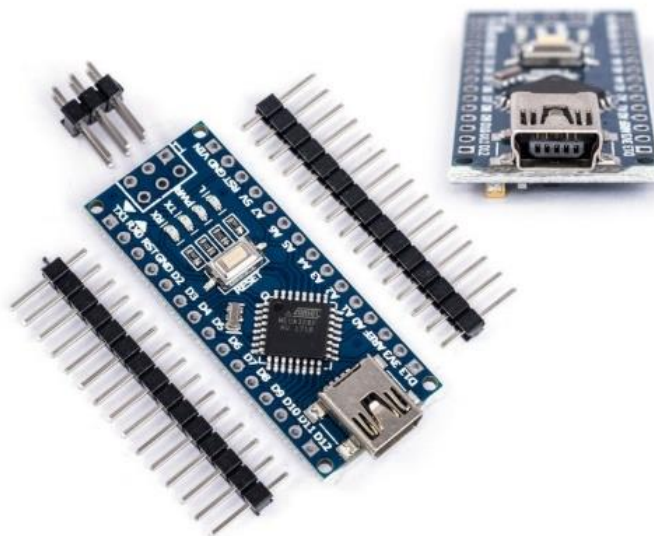


Рисунок – 2.2 Arduino Nano

Основные технические характеристики Arduino Nano:

- Микроконтроллер: ATmega328P
- Рабочая частота: 16 МГц
- Память: 32 Кб Flash-памяти, 2 Кб SRAM, 1 Кб EEPROM
- Число цифровых входов/выходов: 14 (6 из которых могут быть использованы как ШИМ-выходы)
- Число аналоговых входов: 8
- Напряжение питания: 5 В
- Потребляемый ток: до 200 мА

Arduino Nano поддерживает большое количество периферийных устройств и датчиков, что делает ее идеальным выбором для различных проектов, таких как устройства управления роботами, системы домашней автоматизации, сенсорные системы и т.д. Плата Arduino Nano используется во многих проектах, где требуется небольшой размер, но высокая производительность.

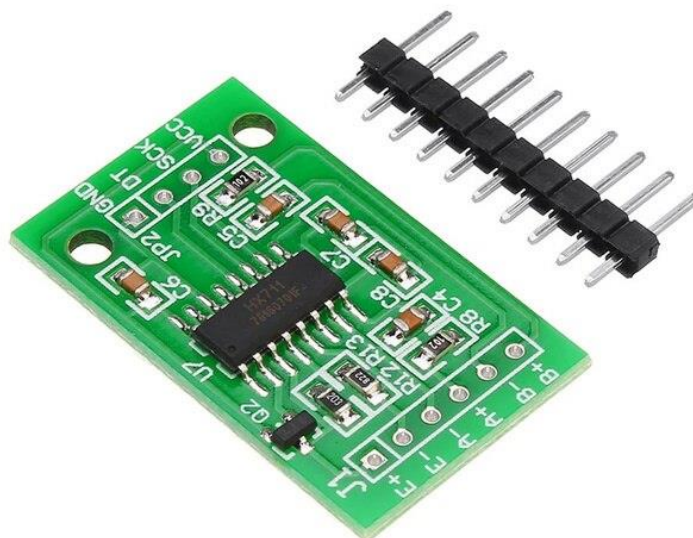


Рисунок – 2.4 Аналогово-цифровой преобразователь НХ711

Основные технические характеристики НХ711:

- Разрядность АЦП – 24 бит
- Усиление по входу А – 64 или 128
- Усиление по входу В – 32
- Частота измерений – 10 или 80 раз в секунду
- Питающее напряжение – 2,6-5,5 В
- Потребляемый ток – менее 10 мА
- Входное напряжение – ± 40 мВ
- На плате с АЦП имеются два разъёма – J1 и JP2, на которых имеются следующие обозначения:
 - J1:
 - E –, E + питание тензорного моста
 - A –, A + дифференциальный вход канала А
 - B –, B + дифференциальный вход канала В
 - JP2:
 - GND, VCC питание
 - DT, SCK – информационные шины

НХ711 подключается к Arduino через интерфейс I2C или SPI. Он имеет два канала входа для подключения тензодатчиков или других источников аналогового сигнала. Кроме того, он обеспечивает встроенное управление энергопотреблением, что позволяет существенно снизить потребление энергии в режиме ожидания.

Для использования НХ711 в проекте необходимо подключить его к Arduino и написать соответствующий код в Arduino IDE. Код должен включать

библиотеку HX711, которая обеспечивает удобный интерфейс для работы с HX711.

2.2.3 Тензометрический датчик веса

Тензометрический датчик веса (также называемый тензодатчиком веса) – это электронный датчик, который используется для измерения веса или силы, которая воздействует на него. Он основан на принципе деформации металлической пластины под воздействием внешней силы. Он состоит из тонкой металлической полоски (резистора), которая натянута между двумя болтами. Когда на датчик веса действует нагрузка, полоска растягивается, и это вызывает изменение сопротивления резистора. Это изменение сопротивления затем измеряется и преобразуется в соответствующее значение веса или силы. Тензодатчики веса широко используются в промышленности, торговле, научных и медицинских приборах, а также в домашней электронике.



Рисунок – 2.4 Тензометрический датчик веса

Принцип работы тензодатчика веса основан на эффекте пьезоэлектрической деформации материала. Тензодатчик состоит из тонкой металлической полоски, натянутой между двумя базами. Когда на тензодатчик действует вес, металлическая полоска деформируется, изменяя свое сопротивление электрическому току, который проходит через нее. Таким образом, изменение сопротивления тензодатчика пропорционально приложенному весу.

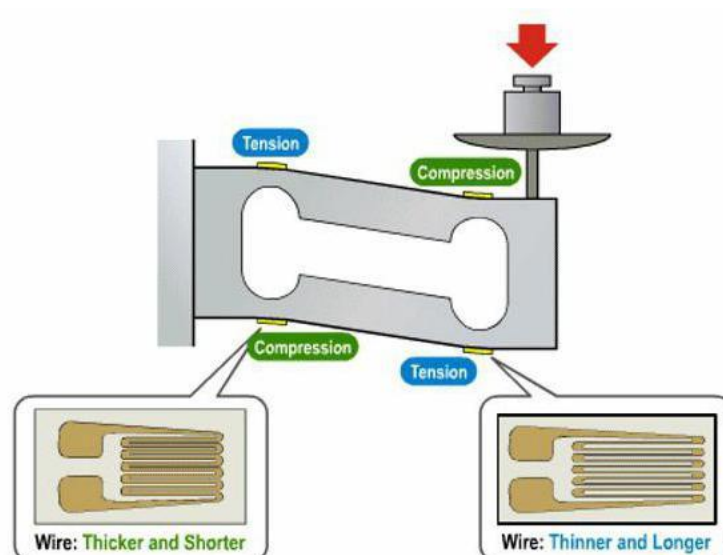


Рисунок – 2.4 Принцип работы тензодатчика веса

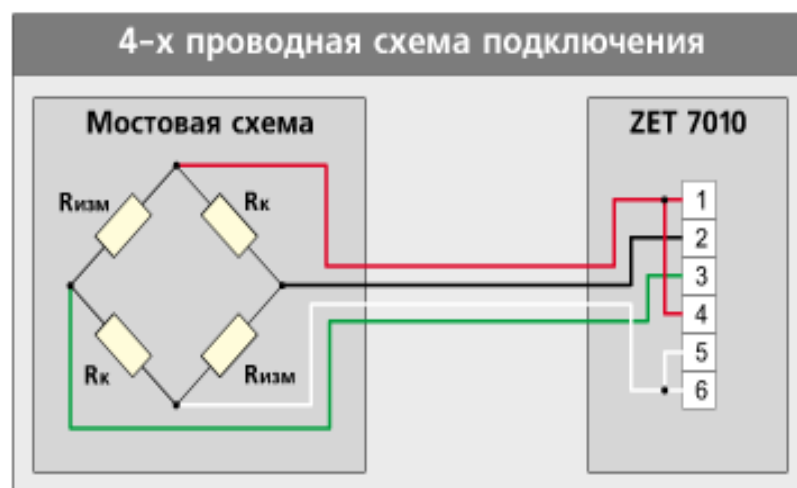


Рисунок – 2.6 Схема устройства и подключения мостового датчика

В данной конструкции используется датчик веса, выполненный на упругом резисторе и при изменении веса, меняется его сопротивление, а, следовательно, и напряжение, снимаемое с мостовой схемы.

Для измерения веса с помощью тензодатчика необходимо использовать усилитель сигнала, так как изменение сопротивления тензодатчика очень мало. НХ711 является таким усилителем, который может усилить сигнал от тензодатчика до уровня, который можно считать с помощью Ардуино.

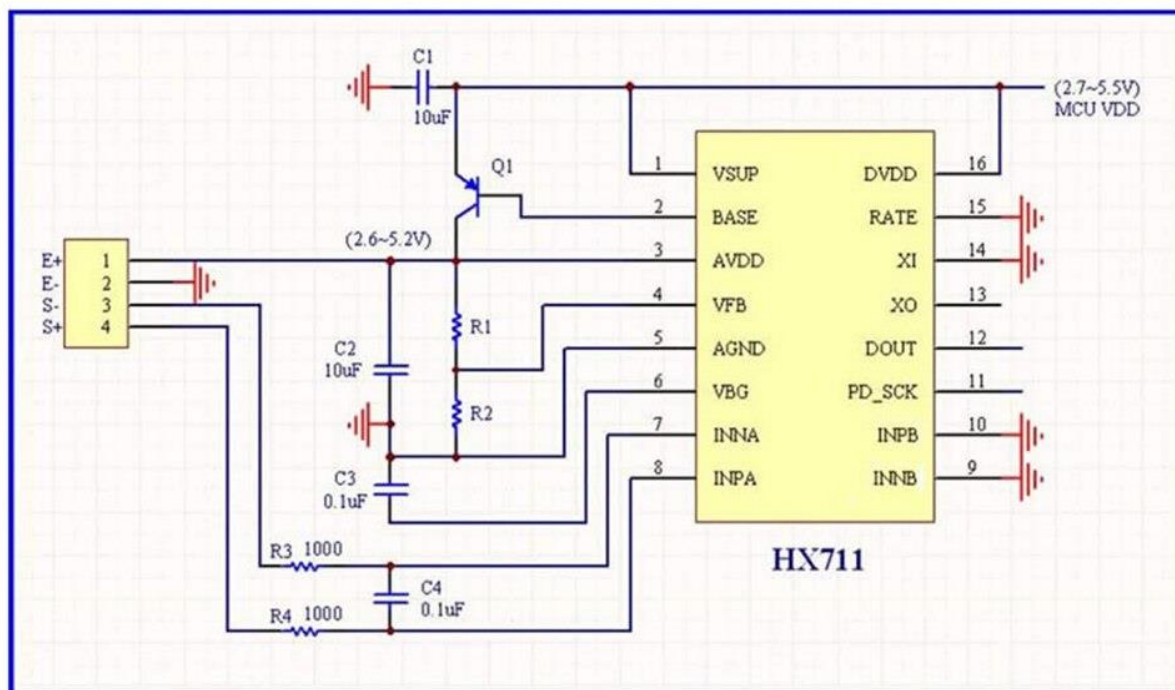


Рисунок – 2.6 Схема модуля HX711

2.2.4 Схема подключения устройств к плате Arduino Nano

HX711 поддерживает двухканальный дифференциальный вход, что позволяет измерять вес с двух нагрузочных датчиков одновременно. Он также имеет встроенный 24-битный аналого-цифровой преобразователь (ADC), который обеспечивает высокую точность и разрешение для измерений веса.

Для создания устройства были использованы Arduino Nano и HX711. Arduino Nano обеспечивает интерфейс для программирования и управления, а HX711 выполняет измерение веса с помощью подключенных весовых датчиков.

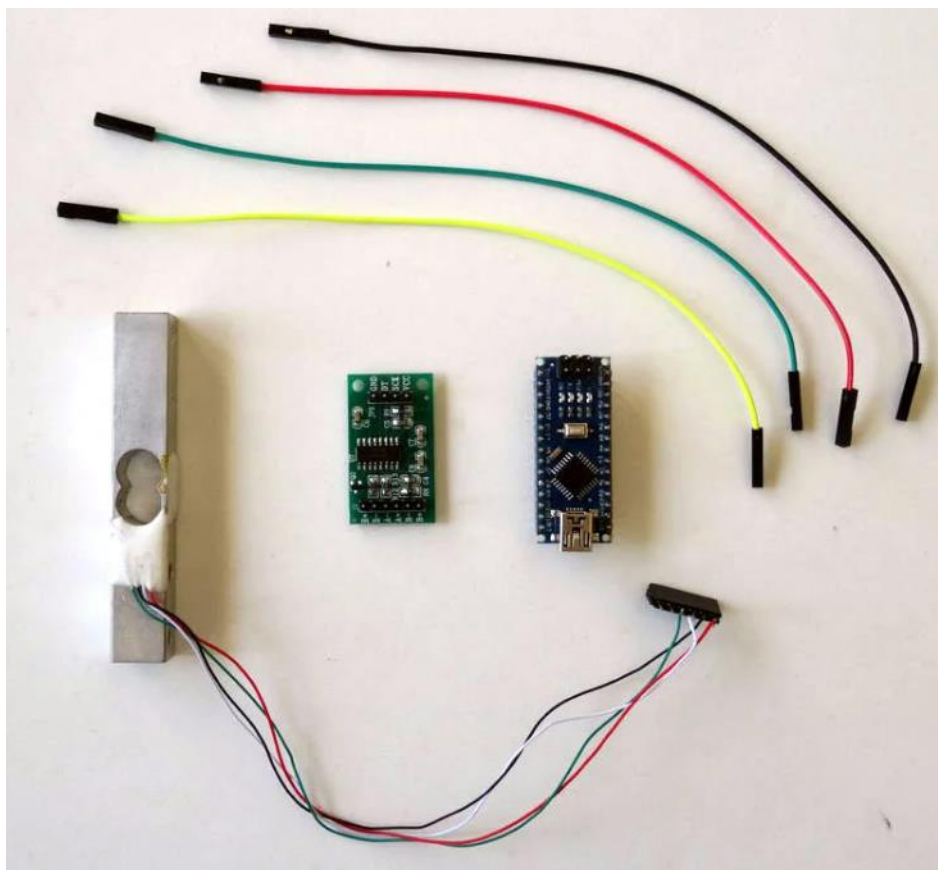


Рисунок – 2.7 Arduino Nano и HX711 с тензодатчиком веса.

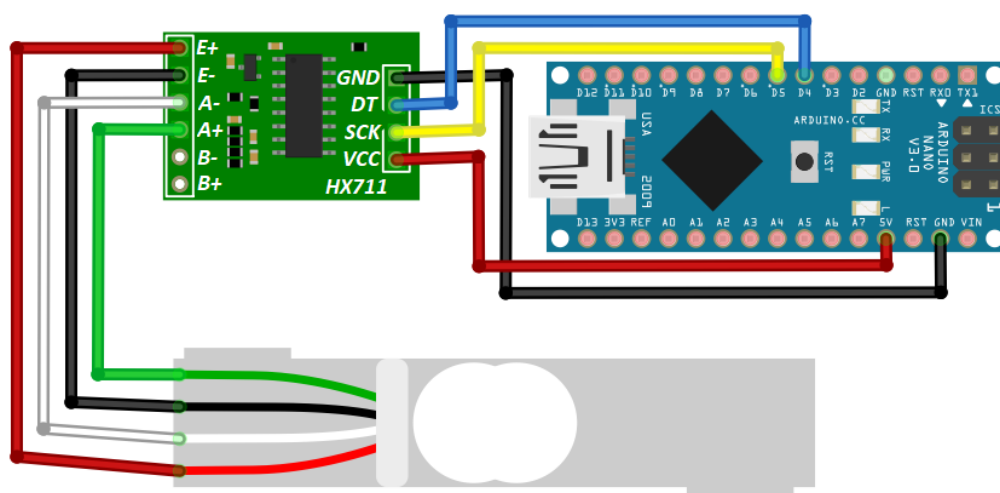


Рисунок – 2.7 Схема подключения Arduino Nano и HX711 с тензодатчиком веса.

3 Программное обеспечение

3.1 Среда разработки Arduino IDE

Arduino IDE - это среда разработки (IDE) для программирования микроконтроллеров на базе Arduino. Она позволяет разрабатывать, загружать и отлаживать программы для платформ Arduino. Среда Arduino IDE основана на языке программирования Wiring, который является простым языком для написания скетчей (небольших программ на Arduino). Она предоставляет удобный интерфейс для создания, редактирования и загрузки кода в плату Arduino, а также для управления библиотеками и скетчами.

Среда Arduino IDE позволяет разработчикам писать код на C или C++, а также использовать специальные библиотеки Arduino для управления аппаратными компонентами платформы. После написания кода, он может быть загружен в плату Arduino через USB-порт, используя встроенный загрузчик.

Arduino IDE также предоставляет широкий набор инструментов для отладки кода, включая монитор порта, который позволяет выводить данные на компьютер для анализа, и серийный монитор, который выводит данные из платформы на компьютер в реальном времени. Кроме того, существует большое сообщество разработчиков и пользователей Arduino, которые предоставляют множество примеров кода и библиотек для использования в своих проектах.

Среда разработки Arduino интегрирована в программную кодировку из редактора, области уведомлений, окна ввода текста (консоль), с панели приборов кнопками часто используемых команд и состоит из некоторых меню. Взаимодействие между загрузкой и разработкой программы для целей подключения Arduino подключается к аппаратной части.

Помните, что программа должна иметь две неотъемлемые функции необходимо сохранить: `setup()` и `loop()`. После именованной функции и после простых скобок выражение, в которое вставляется код, записывается в скобки.

Функция `Setup ()` включает каждый блок питания или Arduino после снятия платы активируется только один раз. Это функция инициализация переменных в «теле», работа цифровых портов производится установка порядка и т.п.

Функция `Loop ()` записана в ее «теле» в бесконечном цикле выполняет команды по порядку.

В дипломной работе используем библиотеку «HX711.h» для работы с тензодатчиком веса. В функции `setup()` мы настраиваем последовательный порт и указываем пины для подключения HX711.

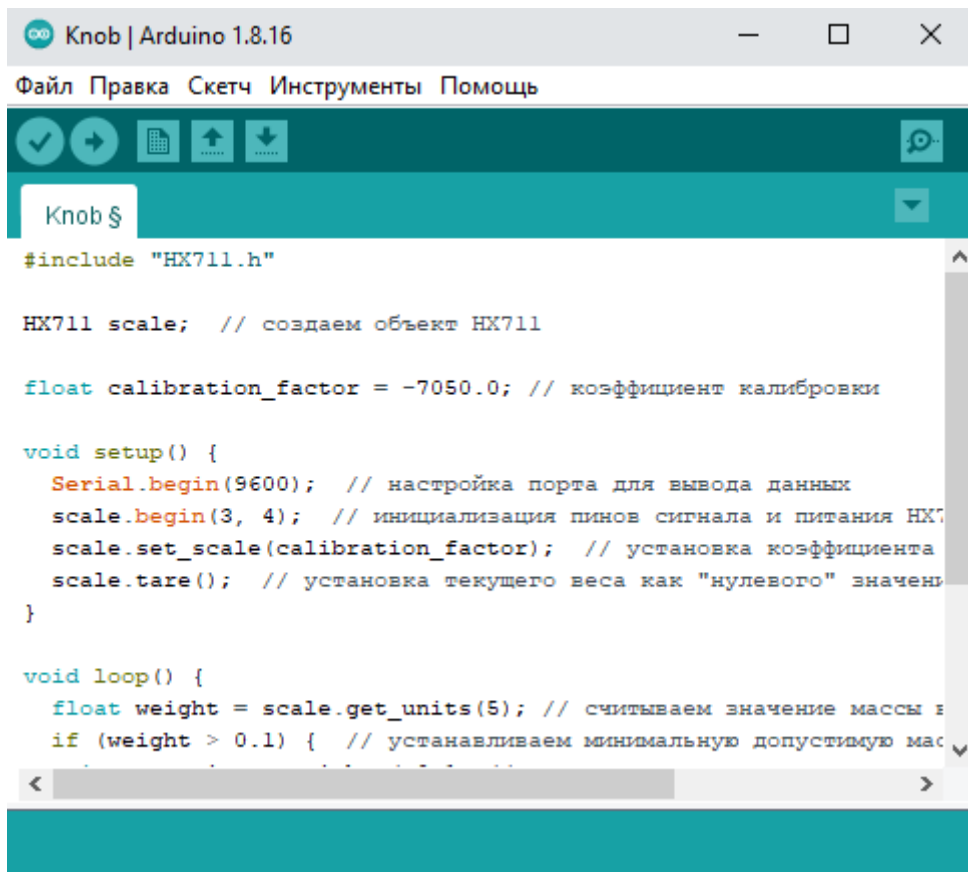


Рисунок – 3.1 Среда разработки Arduino IDE

3.2 Среда разработки PyCharm

PyCharm - это интегрированная среда разработки (IDE) для языка программирования Python, разработанная компанией JetBrains. Она предоставляет широкий набор инструментов и функций, которые помогают программистам повысить производительность и эффективность при разработке приложений на Python.

Для приема данных с Arduino в PyCharm вам потребуется настроить соответствующие порты связи и протокол передачи данных. Вот общий процесс:

- 1) Установим плагин "PySerial" в PyCharm. Этот плагин позволяет взаимодействовать с последовательным портом, который используется для связи с Arduino.
- 2) Подключаем Arduino к компьютеру с помощью USB-кабеля.

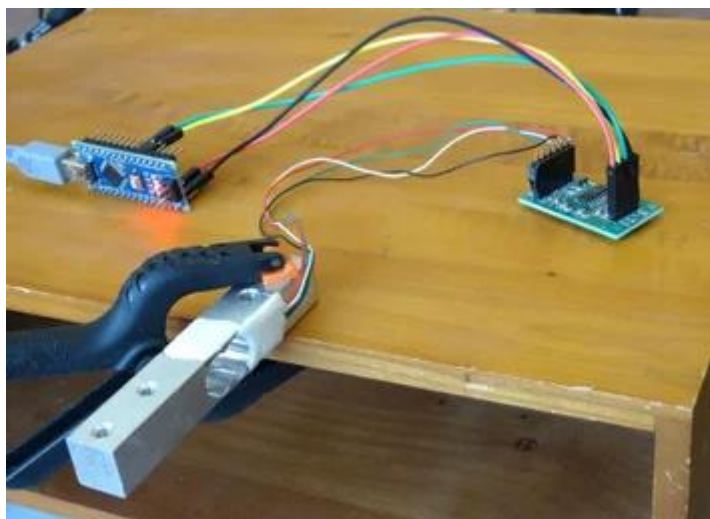


Рисунок – 3.2 Подключаем Arduino к компьютеру и отправляем данных

- 3) Откроем PyCharm и создаём новый проект Python.
- 4) Импортируем модуль serial из пакета serial:

```
python

import serial
```

Рисунок – 3.3 Импортируем модуль serial из пакета serial

- 5) Откроем последовательный порт, связанный с Arduino, с помощью функции Serial():

```
python

ser = serial.Serial('COMX', 9600)
```

Рисунок – 3.4 Последовательный порт, связанный с Arduino

- 6) Получаем данные, отправленные с Arduino, с помощью функции readline() или readlines():

```
python

data = ser.readline() # Чтение одной строки данных
# или
data = ser.readlines() # Чтение всех доступных строк данных
```

Рисунок – 3.5 Чтение полученных строки данных

7) Обрабатываем полученные данные в соответствии с вашими потребностями.

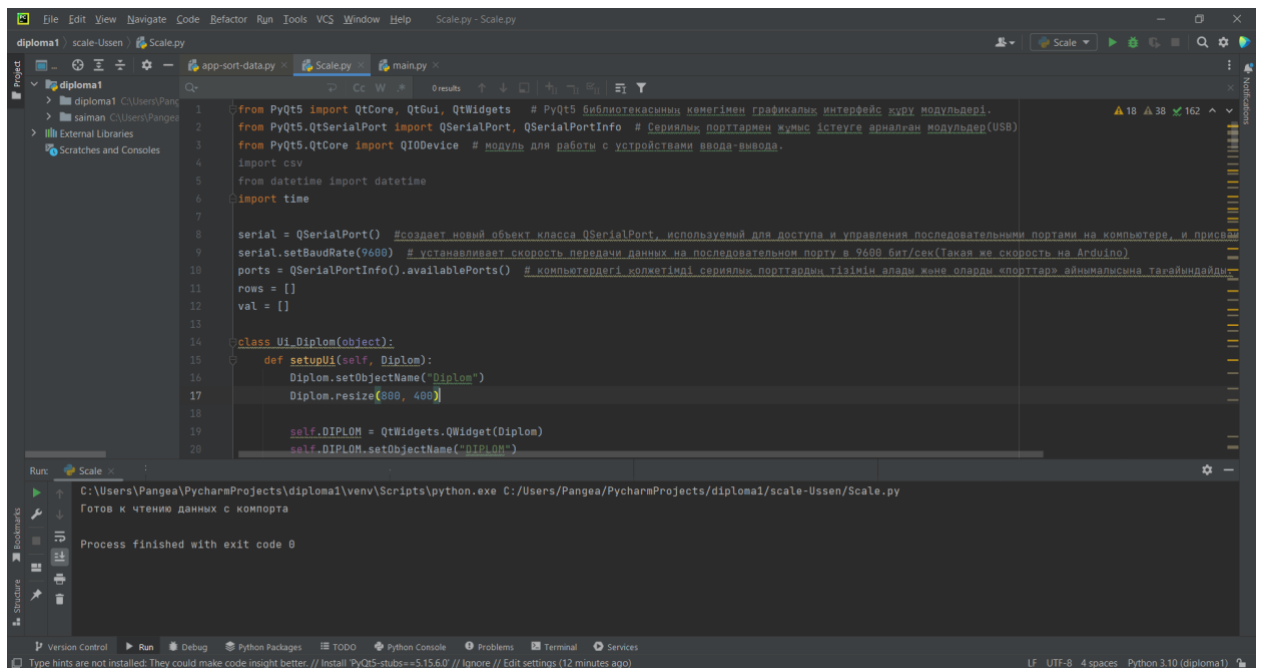


Рисунок – 3.6 Интерфейс PyCharm

Данный код реализует графический интерфейс с помощью библиотеки PyQt5. Он используется для создания приложения, которое взаимодействует с устройством через последовательный порт (USB).

Код импортирует необходимые модули и создает объект класса QSerialPort, который используется для работы с последовательным портом. Затем происходит настройка интерфейса, создание виджетов и размещение их на форме.

Основной класс Ui_Diplom определяет методы для настройки пользовательского интерфейса и обработки событий. Он содержит виджеты, такие как QLCDNumber для отображения чисел, QLabel для отображения текста и QPushButton для кнопок с изображениями. Также он определяет метки и расположение элементов на форме.

Класс Window наследует от QWidget и создает экземпляр класса Ui_Diplom. Он связывает сигнал readyRead объекта QSerialPort с функцией onRead, которая считывает данные из последовательного порта. Полученные данные отображаются на дисплеях и записываются в CSV-файл.

В блоке кода в конце происходит проверка, является ли данный скрипт основным, и если да, то создается объект QApplication и объект Window. Приложение отображается на экране и запускается цикл обработки событий.

Основная функция кода выполняет настройку интерфейса, запускает приложение и ожидает его завершения.

В целом, данный код создает графический интерфейс для взаимодействия с устройством через последовательный порт, отображает полученные данные и записывает их в файл.

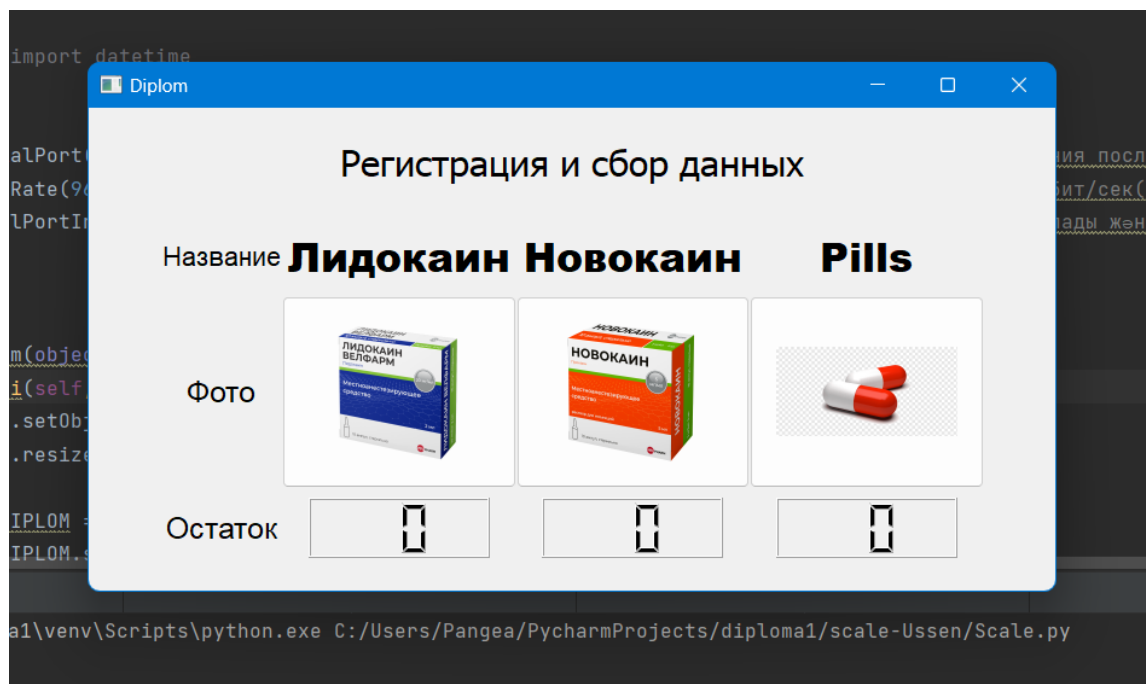


Рисунок – 3.7 Окно проекта

После выполнения данного кода открывается окно, в котором отображается графический интерфейс приложения. Это окно создается с использованием библиотеки PyQt5 и содержит следующие элементы:

- Три секционных индикатора (QLCDNumber): lcd1, lcd2 и lcd3. Они отображают числовые значения и предназначены для отображения данных.
- Три надписи (QLabel): label1, label2 и label3. Они содержат названия соответствующих секционных индикаторов.
- Три кнопки (QPushButton): photo1, photo2 и photo3. Они содержат изображения и предназначены для отображения фотографий.
- Три надписи (QLabel): label_name, label_photo и label_ost. Они содержат названия соответствующих секций (название, фото, остаток).
- Надпись (QLabel): Topic. Она содержит заголовок окна.

Окно настроено на размер 800x400 пикселей и содержит все перечисленные элементы в определенной компоновке с использованием QGridLayout.

Кроме того, код устанавливает соединение с последовательным портом, чтобы читать данные с него. Когда приходят данные, они считываются, обрабатываются и отображаются на секционных индикаторах. Также есть функциональность для записи данных в файл CSV.

После запуска приложения, окно отображается на экране, и пользователь может взаимодействовать с ним, например, видеть данные на индикаторах и фотографии, а также записывать данные в файл CSV.

Этот код устанавливает соединение с Arduino Nano через последовательный порт для чтения данных с него. Взаимодействие между компьютером и Arduino Nano осуществляется через USB-порт.

Чтобы подключить Arduino Nano к компьютеру, необходимо выполнить следующие шаги:

- Подключаем Arduino Nano к компьютеру с помощью USB-кабеля.
- Убеждаемся, что необходимое программное обеспечение установлено. Для работы с Arduino Nano вам понадобится Arduino IDE, который можно скачать с официального сайта Arduino.
- Открываем Arduino IDE и выбираем соответствующую плату и порт. Для Arduino Nano выбираем плату "Arduino Nano" или "Arduino Nano 328" в меню "Инструменты" (Tools), а затем выбираем правильный порт подключения в меню "Порты" (Ports).
- Загружаем программу на Arduino Nano. Убеждаемся, что в коде определены функции для чтения данных с порта, их обработки и вывода на секционные индикаторы и фотографии. Проверяем, что пины Arduino, используемые в коде, правильно подключены к секционным индикаторам и фотографиям.
- После загрузки программы на Arduino Nano она начнет выполнение, и данные будут отправляться через последовательный порт на компьютер.

В коде на компьютере используется библиотека PyQt5 для создания графического интерфейса и взаимодействия с Arduino Nano. Код устанавливает соединение с последовательным портом, считывает данные с порта с помощью функции `serial.readLine()`, а затем обрабатывает и отображает их на секционных индикаторах и фотографиях.

```
def onRead(self): # Функция для чтения из последовательного порта
    while serial.canReadLine(): # Пока есть данные для чтения с последовательного монитора
        rx = serial.readLine() # Чтение данных из последовательного монитора
        rxs = str(rx, 'utf-8').strip() # Декодирование в строку
        data = rxs.split(',') # Разделить строку символом ',' символ для получения отдельных значений
        print(data)
```

Рисунок – 3.8 Функция `serial.readLine()`

Таким образом, Arduino Nano и компьютер взаимодействуют через USB-порт, и код на компьютере может считывать данные с Arduino Nano и отображать их на графическом интерфейсе.

ЗАКЛЮЧЕНИЕ

В ходе работы было проведено исследование существующих устройств регистрации и сбора данных на основе массы товаров, проведен анализ и выбраны компоненты для создания прототипа устройства. Была использована аппаратная платформа Arduino Nano, тензодатчик веса и HX711 для измерения массы товаров.

Был разработан код для Ардуино, который позволяет работать с тензодатчиком веса и HX711 для регистрации и сбора данных на основе массы товаров и определения количества товара. Прототип устройства был создан и протестирован, показав хорошие результаты в точности и надежности измерений.

Таким образом, созданное устройство является эффективным инструментом для регистрации и сбора данных о количестве медицинских товаров на основе их массы. Оно может быть использовано в медицинских учреждениях и аптеках для автоматизации процесса учета товаров и улучшения качества обслуживания пациентов.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. <https://www.smartaids.ru/catalog/product/umnaya-trost-robin/>
2. <https://www.smartaids.ru/catalog/product/elektronnaya-trost-ray/>
3. http://dostupsreda.ru/store/dlya_invalidov_po_zreniyu/bytovaya_tekhnika/ultra_zvukovoy_fonar_sonar-5uf-v1/
4. <http://arduino.ru/Hardware/ArduinoBoardNano>
5. <http://arduino.ru/Reference/Else>
6. <http://arduino.ru/Reference/Setup>
7. <http://arduino.ru/Reference/Loop>
8. <https://doc.arduino.ua/ru/prog/SoftwareSerial>
9. <https://playground.arduino.cc/Code/NewPing/>
10. Сомая Мадакам (август 2015 г.) "интернет вещей: умные вещи" Международный журнал связи, Vol. 4, № 4
11. Карабанов Ю. Ф. Расчет Теплообменника: учебное пособие
12. Эштон, К. "это интернет вещей" RFID Journal, Vol.22, № 7, стр. 97-114
13. <https://yagazeta.com/stil-zhizni/umnye-ochki-esight-podaryat-zrenie-slepym/>
14. <https://style.rbc.ru/items/57163a149a79472acdb3590e>
15. http://od-os.ru/articles/426/?sphrase_id=570900
16. https://www.smartaids.ru/catalog/sighting_loss/
17. МСЭ Гершенфельд, Н.: Когда вещи начинают. Генри Холт и компания (1999).
18. Европейская комиссия: интернет вещей Европейский план действий. COM (2009)
19. <https://habr.com/ru/post/309746/>
20. <https://www.arduino.cc/>