

Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»
(Университет ИТМО)

Факультет программной инженерии и компьютерной техники

Образовательная программа: Системное и прикладное программное обеспечение

Направление подготовки (специальность): 09.03.04, Системное программное обеспечение

ОТЧЕТ

о производственной, преддипломной практике

Тема задания: *Оформление результатов по разработке автоматизированного трекера для настольных игр*

Обучающийся: *Конаныхина Антонина Александровна, Р34102*

Руководитель практики: *Пинкевич Василий Юрьевич, доцент факультета программной инженерии и компьютерной техники по сов-ву*

Руководитель практики от университета: *Маркина Татьяна Анатольевна, старший преподаватель факультета программной инженерии и компьютерной техники*

Дата: 25.05.2024

Санкт-Петербург

2024

ОГЛАВЛЕНИЕ

1	ВВЕДЕНИЕ.....	3
2	ОСНОВНАЯ ЧАСТЬ	5
	2.1 Написание текста ВКР	5
	2.2 Проверка текста ВКР в системе Антиплагиат	13
	2.3 Предзащита ВКР	13
3	ЗАКЛЮЧЕНИЕ	15

1 ВВЕДЕНИЕ

Темой данной производственной практики является оформление результатов по разработке автоматизированного трекера для настольных игр.

Целью работы является улучшение качества игрового процесса в настольных ролевых играх за счет автоматизации. Демонстрация результатов достижения цели осуществляется посредством выполнения **индивидуального задания**. (Таблица 1).

Таблица 1 – Индивидуальное задание

Порядковый № этапа	Наименование этапа	Период этапа в датах	Задание этапа
1	Инструктаж обучающегося	29.04.2024 - 30.04.2024	Инструктаж обучающегося по ознакомлению с требованиями охраны труда, техники безопасности, пожарной безопасности, а также правилами внутреннего трудового распорядка
2	Написание текста ВКР	01.05.2024 - 14.05.2024	Подготовить текст ВКР в соответствии с инд. заданием. Текст ВКР представляется в электронном виде и включает введение, основную часть, заключение (не менее 40 м.п.с для бакалавров и не менее 60 м.п.с. для магистров, не более 80 м.п.с.). Во введении дается общая характеристика работы (актуальность темы, цель и задачи исследования, степень проработанности проблемы, методы исследования, объект и предмет исследования, положения, выносимые на защиту, научная новизна и практическая значимость работы, апробация и внедрение результатов исследования,

			<p>личный вклад). При составлении обзора необходимо использовать не менее 20 источников не старше 5 лет, в том числе не менее 20% научных статей в изданиях, рекомендованных ВАК РФ для диссертаций, и не менее 15% в изданиях, индексируемых в базах WoS/Scopus. В заключении приводится список основных результатов исследования, с обоснованием их научной и практической значимости. При формулировке результатов необходимо указывать количественные и качественные характеристики экспериментальных исследований, подтверждающих достижение поставленной цели. Обязательным пунктом заключения должны быть пути дальнейших направлений исследований по теме ВКР.</p>
3	Проверка текста ВКР в системе Антиплагиат	15.05.2024 - 21.05.2024	Сдать на проверку текст ВКР секретарю ГЭК и получить справку о прохождении проверки (не более 25% заимствований).
4	Предзащита ВКР	22.04.2024 - 23.04.2024	Подготовить электронную презентацию (не более 15 слайдов) с результатами работы и пройти предзащиту (готовность к защите не менее 60%).
5	Оформление отчёта	24.05.2024 - 30.05.2024	<p>Оформление отчётных документов по практике в соответствии с требованиями:</p> <ol style="list-style-type: none"> 1. Подробное описание выполнения задач по этапам. 2. Оформление отчёта в

			соответствии с методическим пособием (https://books.ifmo.ru/file/pdf/2622.pdf) 3. Структура документа в соответствии с методическим пособием.
--	--	--	--

2 ОСНОВНАЯ ЧАСТЬ

2.1 Написание текста ВКР

Работа над текстом ВКР началась с анализа рынка аксессуаров для настольных ролевых игр. В ходе этого анализа были прочитаны современные исследования про влияние изоляции во время пандемии на ментальное состояние игроков, про существующие решения использования автоматизации в сфере НРИ и статьи про разработку внешних устройств на базе микроконтроллеров. Были найдены ближайшие по функционалу разрабатываемого трекера аналоги. Этими аналогами являются деревянный трекер, простое и дешёвое решение без автоматизации, но не универсальное, и цифровой стол со встроенным экраном. Последнее решение универсально, но очень громоздкое и дорогое, недоступное широким массам игроков. Анализируя плюсы и минусы данных аналогов, были разработаны базовые предъявляемые требования разрабатываемого в рамках ВКР устройства. Результатом этой работы стала схема подключения устройства к ПК (Рисунок 1).

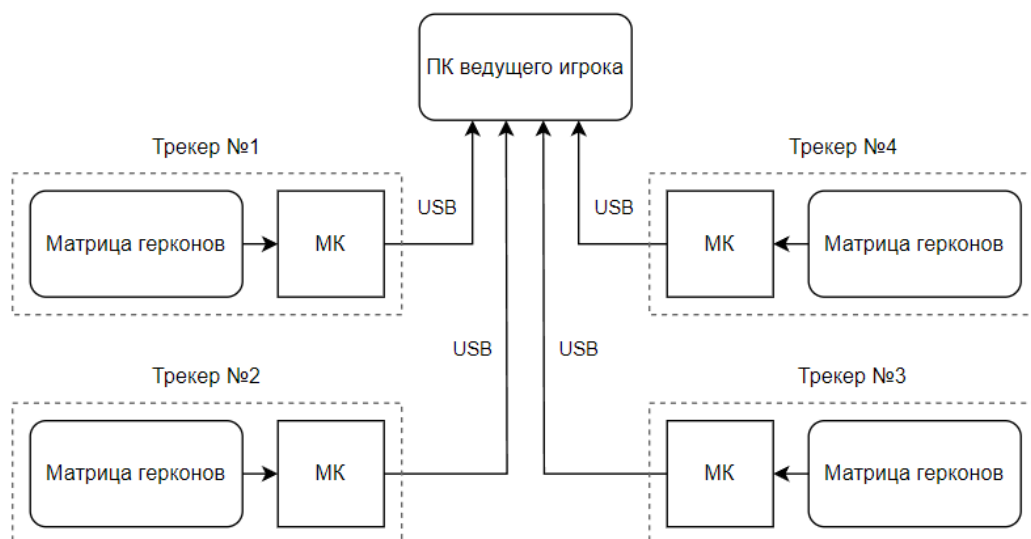


Рисунок 1. Схема работы системы из четырёх трекеров

Также были разработаны функциональные требования к разрабатываемому устройству:

1. Система должна собирать данные о наличии магнитных меток на устройстве, об их расположениях и количестве.
2. Система, основываясь на полученном расположении меток, должна находить численные значения характеристик, соответствующих текущему расположению меток. В зависимости от настроек конфигурации численные значения могут рассчитываться исходя из положения меток, их количества или взаимного расположения;
3. Собранная информация должна передаваться устройством на персональный компьютер, подключённый к нему по интерфейсу USB;
4. На персональном компьютере информация должна отображаться пользователю через графический интерфейс;
5. Программное обеспечение должно предоставлять пользователю возможность настройки отображения и интерпретации полученных данных.

После формулировки требований необходимо было приступить к работе над аппаратной частью устройства. Вообще идея использовать именно

герконы, а не кнопки и другие элементы, была позаимствована у некоторых аналоговых трекеров, которые просто использовали магниты без каких-либо датчиков, чтобы метки при сотрясании стола никуда не разлетались. Была разработана аппаратная составляющая устройства. В первую очередь, решалась проблема подключения герконов в матрицу, так как подобная задача имеет несколько вариантов решения. В рамках ВКР были рассмотрены три способа подключения. Первое – через подключение каждого геркона по отдельности к каждому входу микроконтроллера, второе – через подключение герконов по строкам и столбцам, и третье – через суммирование токов. Каждое из решений было проанализировано и просимулировано, были выявлены плюсы и минусы каждого подключения и была проанализирована возможность и удобства применения каждого решения для разрабатываемого устройства. Из всех вариантов подключения было выбрано подключение по столбцам и строкам, но в ходе тестирования было установлено, что следовало выбрать третье решение, с суммированием токов.

В качестве контроллера была выбрана плата STM32 NucleoF334, так как характеристики данного микроконтроллера хорошо подходят для решения поставленной задачи.

После выбора способа подключения герконов была разработана принципиальная электрическая схема устройства. Схема была разработана в среде EasyEDA. Итоговая принципиальная электрическая схема представлена ниже (Рисунок 2 и Рисунок 3).

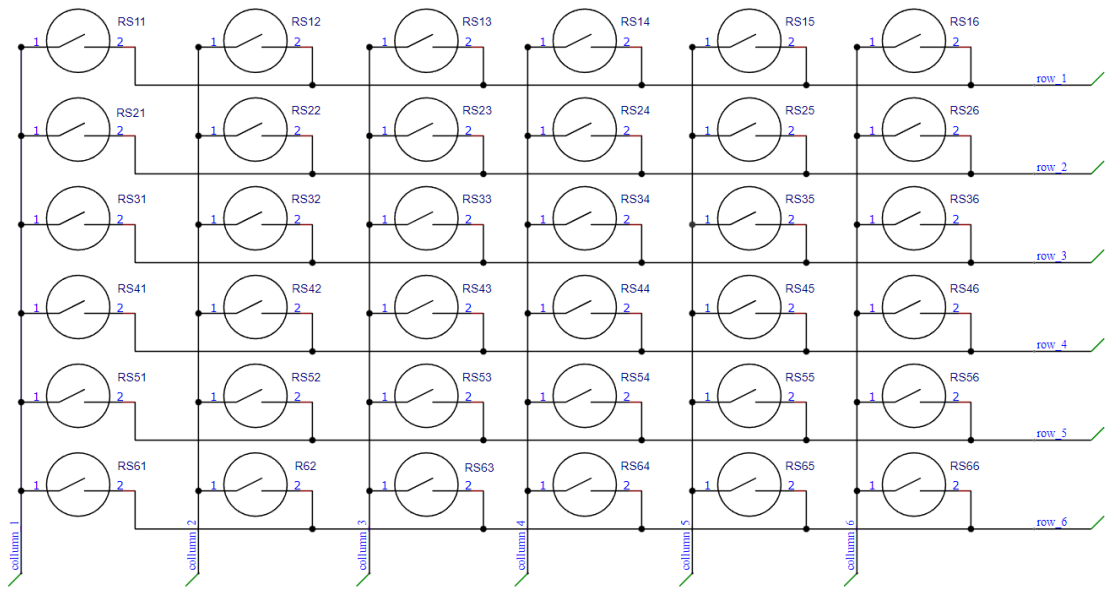


Рисунок 2. Принципиальная электрическая схема, часть с матрицей герконов

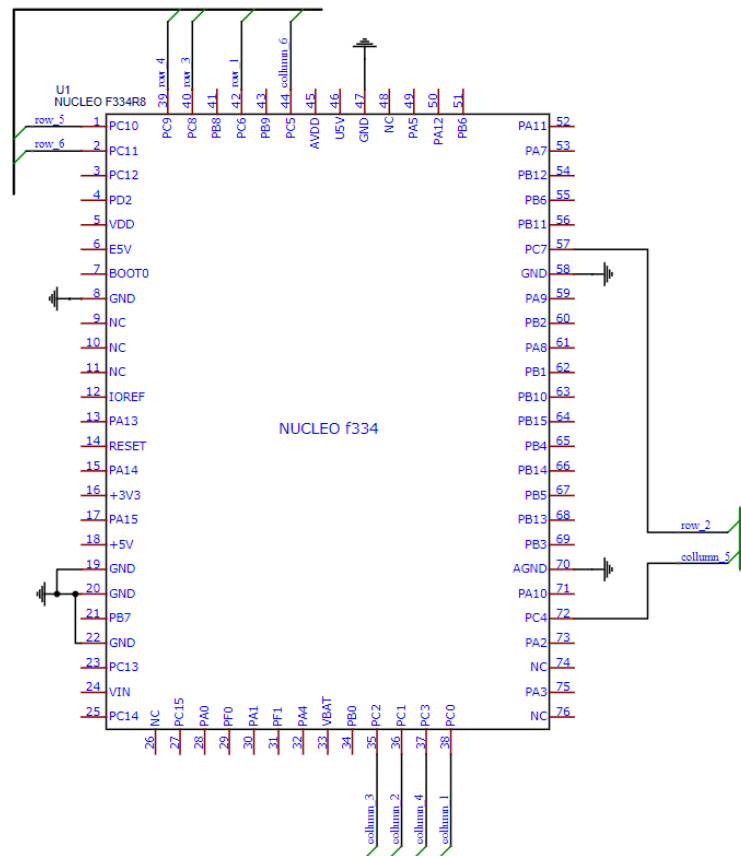


Рисунок 3. Принципиальная электрическая схема, часть с подключением микроконтроллера

На основе данной схемы было смонтировано тестовое устройство (Рисунок 4). Основной сложностью сборки оказалась хрупкость герконов, в

ходе монтажа несколько запасных датчиков случайно были сломаны, но в итоге всё удалось спаять.

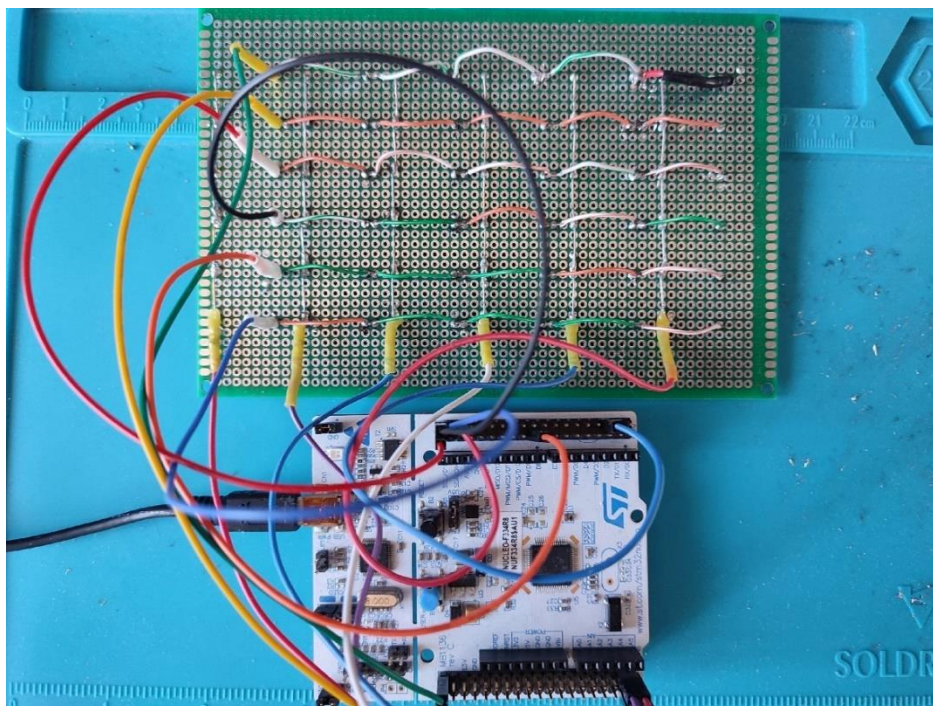


Рисунок 4. Тестовый стенд

В ходе работы над ВКР стояла задача разработать две программы. Первая, для платформы STM32, это программа самого устройства. Вторая для платформы ПК, это программа для пользовательского интерфейса.

В ходе разработки ПО для STM32 необходимо было разработать ПО, расширяемое новым функционалом без особых трудностей. Поэтому сначала была разработана UML диаграмма программы (Рисунок 5).

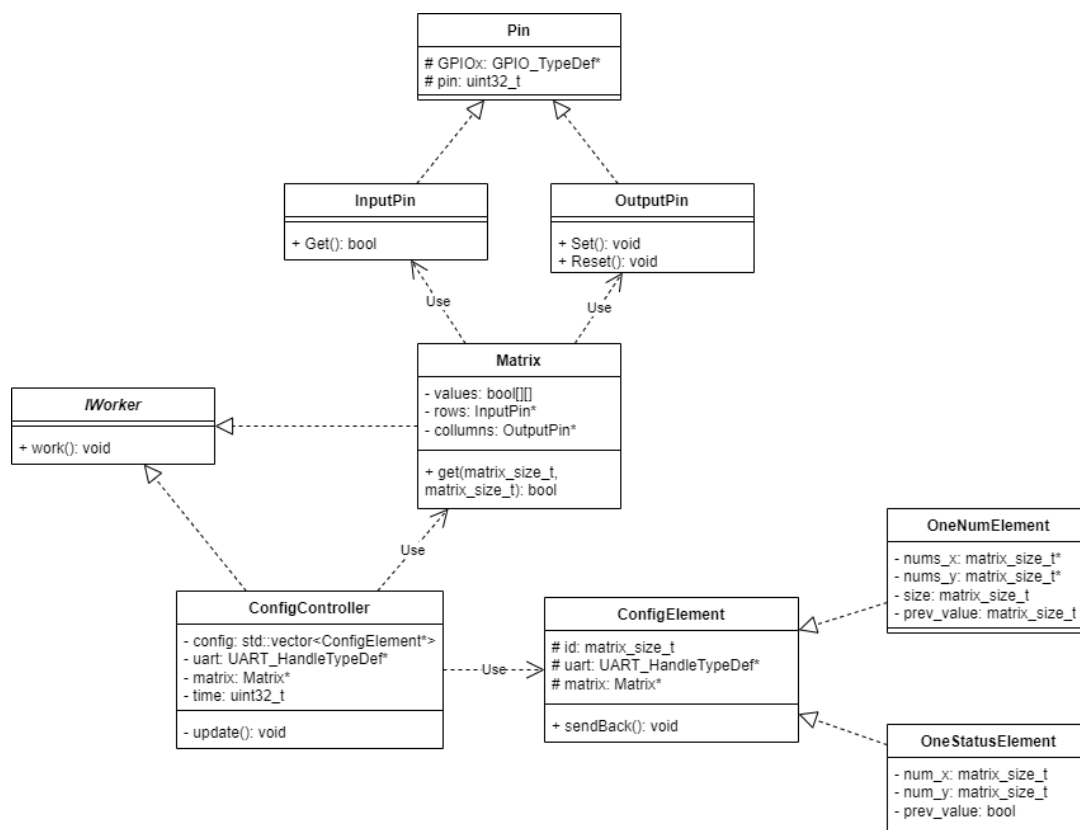


Рисунок 5. UML диаграмма программы для микроконтроллера

В диаграмме отражена связь классов между собой и, в первую очередь, связь между классами, описывающими конфигурацию. Это абстрактный класс ConfigElement и все его реализации. Такая организация кода позволяет добавить новый элемент, если понадобится добавить что-либо или поменять, так как учесть всех потребностей пользователей на этапе разработки прототипа невозможно. Также были реализованы базовые функции для работы с GPIO, для сбора данных с матрицы, и управление всем вышеперечисленным. Так как разработка велась в среде разработки STM32CubeIDE, после разработки диаграммы был использован инструмент конфигурации периферии через графический интерфейс. После чего на основе сгенерированного кода и диаграммы классов была написана сама программа. В ходе написания ПО были решены такие вопросы, как выбор baud rate интерфейса взаимодействия с ПК и разработка протокола, с помощью которого ПК и устройство обмениваются данными.

Вид пакета данных, отправляемого от STM32 к ПК:

8 bit	8 bit
ID	DATA

Вид пакета данных, отправляемого от ПК к STM32:

8 bit	8 bit	8 bit	8 bit
1	ID	X	Y

8 bit	8 bit	8 bit	8 bit	8 bit	8 bit	8 bit	...
0	ID	Size	X1	Y1	X2	Y2	...

Разработка ПО шла снизу вверх, начиная с базовых классов для взаимодействия с GPIO, затем создавался класс управления матрицей, затем уже конфигурация и взаимодействие с ПК. В ходе разработки промежуточные этапы тестировались с помощью отладчика.

Вторая половина ПО концептуально сильно проще и реализовывалась не в объектно-ориентированном стиле. Перед началом реализации ПО для ПК были разработаны макеты пользовательского интерфейса.

Выберите устройство:

☒

Выберите конфигурацию:

Рисунок 6. Макет первой страницы UI

Параметры:

Имя параметра: есть/нет

Имя параметра: число

...

Рисунок 7. Макет второй страницы UI

После разработки макетов был проведен поиск библиотек для реализации пользовательского интерфейса на языке Python. Была выбрана библиотека tkinter. С использованием данной библиотеки были реализованы разработанные ранее макеты. В ходе разработки были решены задачи хранения конфигураций на ПК. Для удобства редактирования конфигурация хранится в текстовом файле разработанного формата.

В ходе разработки тестирование происходило также с помощью отладчиков, но в отличие от прошлой программы уже двух – отладчика Python и отладчика STM32. После разработки UI было проведено тестирование и исправление ошибок. Например, была исправлена низкая производительность пользовательского интерфейса путём отправки только изменений состояний герконов, а не постоянной отправки состояния.

После того, как и устройство, и ПО были разработаны, было проведено тестирование стенда. В ходе тестирования устанавливались как физические, так и пользовательские параметры устройства. Были определены габариты устройства, токопотребление, результаты отражены в тексте ВКР. Из пользовательских параметров были оценены субъективная задержка и удобство использования трекера. Было установлено, что размер итогового устройства составляет 100 мм на 150 мм, питание устройства происходит за счёт питания от ПК ведущего игрока, получаемого по USB интерфейсу. Замеренное токопотребление в среднем составляет 0.14А. Также было установлено, что время отклика устройства незначительно и не заметно с точки зрения пользователей. Подключение устройства удобно. Единственная проблема – это составление пользовательских конфигураций, так как разработанный формат не очень интуитивен.

К сожалению, уже после сборки тестового прототипа в ходе тестирования была обнаружена проблема выбранного способа соединения герконов в матрицу – проблема фантомных нажатий.

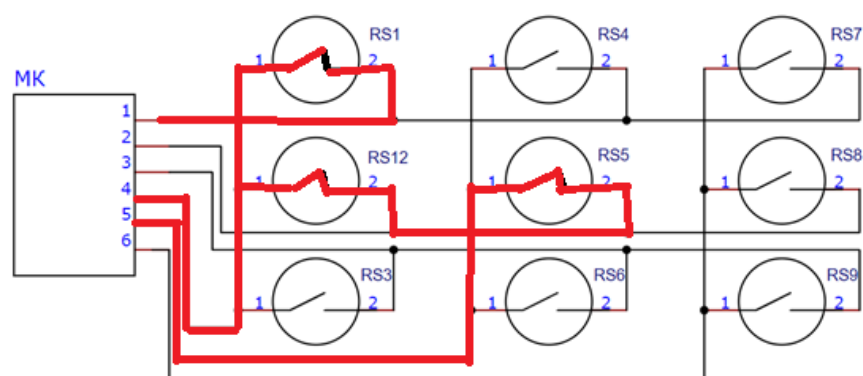


Рисунок 8. Схема возникновения фантомных нажатий

Проблема была подробно описана и проанализирована в тексте ВКР, также были рассмотрены основные способы её устранения – разработка редактора конфигураций, который будет уведомлять и предупреждать пользователя о возможности возникновения фантомных нажатий, и изменение схемы на другой из рассмотренных ранее вариантов подключения герконов.

2.2 Проверка текста ВКР в системе Антиплагиат

В ходе проверки текста ВКР в системе Антиплагиат никаких замечаний не было. Проверка была успешно пройдена с первой попытки с результатом в 87%, что выше порогового значения в 75%.

2.3 Предзащита ВКР

И в ходе первой мартовской предзащиты, и в ходе второй предзащиты в мае со стороны комиссий не было задано никаких вопросов. Комиссия хорошо оценила тему ВКР, содержание и презентацию к защите. Некоторые члены комиссии с интересом отреагировали на сам разрабатываемый трекер, говоря, что тоже увлекаются настольными играми и им было бы интересно подробнее почитать про трекер, что ещё раз говорит об актуальности проекта в профессиональных кругах.

После предзащиты и разговора с научным руководителем было решено всё же реализовать дополнительно вне рамок ВКР схему подключения с

суммированием токов на случай, если комиссия на защите будет спрашивать более подробно про отказ от этого варианта и то, насколько целесообразно или нет его вообще реализовывать на практике, лучше ли это подключение, чем подключение по столбцам и строкам.

3 ЗАКЛЮЧЕНИЕ

Таким образом, в ходе выполнения ВКР были выполнены поставленные задачи, а именно:

1) Проведён анализ существующих решений, выявлены причины их отсутствия. При решении этой задачи были рассмотрены два основных аналога, были отмечены плюсы и минусы данных решений, и, опираясь на это, были составлены требования к разрабатываемому устройству.

2) Исследованы работа и способы разводки принципиальных электрических схем на базе герконов. Было рассмотрено три способа подключения герконов, каждый в отдельности, оптимальным был выбран способ подключения по столбцам и строкам. Были проанализированы плюсы и минусы каждого решения, для каждого решения отмечена возможность использования в устройстве.

3) Разработана и изготовлена плата расширения для микроконтроллера STM32 Nucleo. В ходе решения данной задачи была разведена принципиальная электрическая схема устройства, после чего по схеме на макетной плате был собран прототип устройства.

4) Написано ПО для конфигурации и настройки трекера и UI для отображения обработанных данных. В ходе решения данной задачи были реализованы две программы, программа для STM32 на языке C++, для сбора данных с матрицы, и отправки на ПК, и программа на языке Python реализующая графический интерфейс.

5) Протестировано разработанное ПО на изготовленном тестовом стенде. Написанное ПО было протестировано на прототипе, собраны основные физические и пользовательские параметры устройства.

6) Оформлена документация. Был написан текст ВКР, прошедший проверку в системе Антиплагиат. А также успешно пройдена предзащита ВКР и получен допуск к защите.

Таким образом, можно сделать вывод, что ВКР реализована в полном объёме, все поставленные задачи выполнены, цель достигнута.