Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

ПЕРМСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ (ПНИПУ)

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

УДК 004

ОТЧЁТ

О ТВОРЧЕСКОЙ РАБОТЕ

АВТОМАТИЗАЦИЯ РАБОЧЕГО МЕСТА РЫБАКА

Руководитель ТР,

доц. кафедры ИТАС \_\_\_\_\_\_\_\_\_\_ О.А. Полякова

Пермь, 2024 г.

СПИСОК ИСПОЛНИТЕЛЕЙ

Руководитель ТР,

доц. кафедры ИТАС \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ О.А. Полякова

подпись, дата

Исполнитель:

Студентка группы ИВТ-24-1б \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_А.И.Балтаева

подпись, дата

РЕФЕРАТ

Отчёт 11 с., 1 кн., 3 рис., 7 источн., 1 прил.

Оборудование:

1. Arduino Uno – контроллер для обработки сигналов.
2. Концевой выключатель (механический) – датчик поклевки, закрепленный на жерлице.
3. USB-кабель – для связи с ПК.
4. Подтягивающий резистор 10 кОм.
5. Макетная плата и провода – для подключения.
6. Платформа Arduino IDE – для прошивки контроллера.
7. Платформа Qt Creator – для создания интерфейса на ПК.
8. QSerialPort – библиотека для работы с COM-порто

Объектом исследования являются параметры, получение которых возможно при помощи датчика, установленного на место сбора данных.

Цель работы – разработка приложения, которое позволяет пользователю – рыбаку фиксировать момент поклевки рыбы в зимний период с помощью концевого выключателя

В процессе работы проводились экспериментальные исследования датчика, проверка работы передачи данные на компьютер в реальном времени через Arduino.

В результате работы было создано приложение, отображающее данные о состояния крючка, поймана ли рыба или нет. Передача данных реализуется при помощи специальных устройств, отправляющих значения параметров на плату, после чего при помощи usb-провода данные поступают в приложение. Данные обрабатываются в виде текста.

Эффективность работы приложения состоит в том, что приложение само автоматически определяет порт, подключает плату в нужное время и выводит данные о состоянии крючка.

**Содержание**

Введение 5

1 Как работает система 6

2 Датчик 6

3 Плата Arduino Uno 7

4 Разработка приложения 8

Заключение11

Список использованных источников13

**Введение**

Зимняя рыбалка – это экстремальный вид отдыха, требующий постоянного контроля снастей в условиях низких температур. Традиционно рыбакам приходится многократно проверять жерлицы (3-5 раз в день при -20°C и ниже), терпеть дискомфорт из-за длительного нахождения на морозе, рисковать здоровьем. Мой проект полностью устраняет эти проблемы, предлагая дистанционный мониторинг поклевок через компьютерное приложение.

На сегодняшний день решения для автоматизированного отслеживания поклевки в режиме реального времени практически отсутствуют на рынке рыболовных товаров. Существующие аналоги либо ограничены простыми звуковыми сигнализаторами, либо требуют постоянного визуального контроля со стороны рыбака. Данный проект направлен на создание системы мониторинга, которая обеспечит наблюдение за снастями без прямого участия человека, оповещение о поклевке через компьютерные устройства.

Задачи:

1. Разработка системы на базе Arduino с концевыми выключателями для фиксации поклевки
2. Создание приложения для просмотра данных.
3. Визуализация данных.

**1 Как работает система**

Жерлец — это рыболовная снасть для ловли хищных рыб. Принцип работы жерлицы для зимней рыбалки заключается в том, что на крючок насаживается живая приманка (живец) и опускается в воду. Голодный хищник захватывает наживку. В этот момент с катушки начинает активно разматываться леска, и срабатывает пружинка.

Для реализации данного проекта на жерлец был установлен механический морозостойкий концевой выключатель. Через него был проведена леска так, что когда рыба заглатывает наживку, леска стягивается и опускает конструкцию концевика. Концевик срабатывает и передает сигнал через dupon-провода на плату Arduino Uno. Плата же через usb-провод передает сигнал в компьютер. Пример работы представлен на рис.1

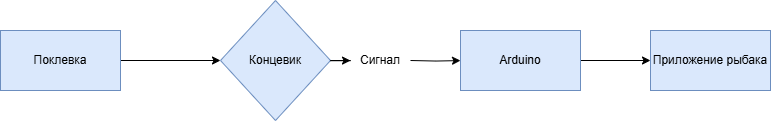


Рисунок 1

**2 Датчик**

Для системы используется механический концевой выключатель с нормально-разомкнутыми контактами . К датчику были припаяны три dupon-провода (п-п), первый – для подключения к питанию через com, второй- для подключения к GND через контакт NO, третий необходим для подключения резистора.

Физический принцип работы:

1. В исходном состоянии:
   * Леска жерлицы освобождает шток выключателя
   * Контакты разомкнуты (HIGH на пине Arduino)
2. При поклевке:
   * Леска давит шток
   * Пружина внутри выключателя замыкает контакты
   * Сигнал на пине Arduino: LOW

Схема подключения концевого выключателя к плате Arduino представлен на рис. 2.

Изображение выглядит как текст, диаграмма, снимок экрана, Параллельный

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 2

**3 Плата Arduino Uno**

Для тестирования была выбрана плата Arduino Uno 5v, так как она компактная, простая в использовании и прекрасно выполняет необходимые функции, низкая себестоимость. Подключение к концевому выключателю уже было рассмотрено раннее в разделе 1. Сама плата подлючена к компьютеру через  USB. Так плата получает питание, а компьютер сигнал от концевого выключателя. Далее необходимо вгрузить скетч на Arduino с помощью платформы Arduino IDE. Это позволит определять сигналы с датчика и передавать его нашему приложению.

Код написан на языке программирования С++(рис. 3).

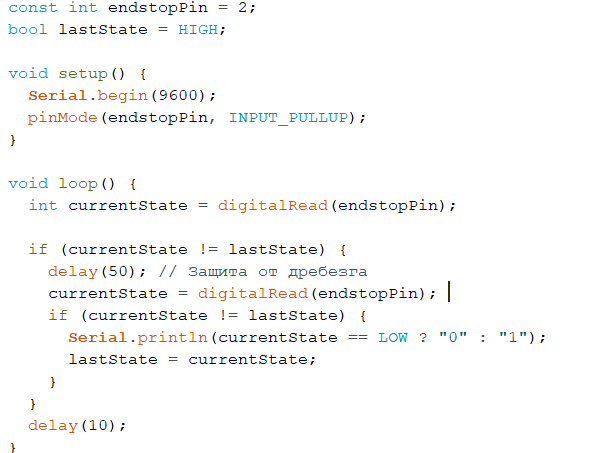


Рисунок 3

**4 Разработка приложения**

Приложение разработано при помощи платформы Qt (creator). Связь между Arduino и платформой Qt производилась с помощью библиотеки QSerialPort. Библиотека предоставляет полный API для работы с последовательными портами. Он поддерживает все стандартные параметры порта (скорость, биты данных и т.д.).

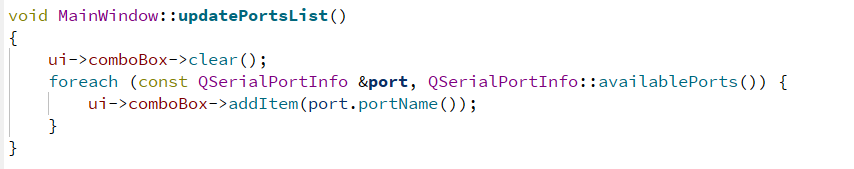
Конструктор MainWindow() инициализирует главное окно приложения и всех компонентов. В данной функции создется ui, инизиализируется порт Com-3, указывается скорость 9600 как и на коде Arduino.

Изображение выглядит как текст, снимок экрана, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 4

Функция updatePortsList() позволяет обновлять список доступных COM-портов автоматически.

Рисунок 5

Функция on\_connectButton\_clicked()выполняет ключевую роль в управлении подключением к Arduino, реализуя установку/разрыв соединения через COM-порт, обработку ошибок подключения. На рис.6 можно увидеть, что , как прописано в коде, создается кнопка подключения к плате, после которой приложение открывает порт или , если возникла ошибка с подключением, выводит ошибку.

Изображение выглядит как текст, снимок экрана, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 6

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Контент, сгенерированный ИИ, может содержать ошибки.Далее создана функция readSerialData(),которая является обработчиком сигнала readyRead() и отвечает за чтение сырых данных с последовательного порта, детектирование состояния поклевки ("1" – рыба поймана, "0" – улова нет)

Рисунок 7

И последняя функция  handleSerialError() отвечает за обработку ошибок последовательного порта. В случае ошибки он закрывает порт и возращает исходный текст.

На рис.8 представлена UML-диаграмма данного кода.

Изображение выглядит как текст, снимок экрана, диаграмма, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.Рисунок 8

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Контент, сгенерированный ИИ, может содержать ошибки.

**4 Перспективы**

Сейчас проект работает по проводу, но в будущем можно добавить радиомодуль (nRF24L01, LoRa или GSM), чтобы отслеживать поклёвки на расстоянии.

**Заключение**

Проект «Зимняя рыбалка» успешно решает важную задачу — автоматизацию контроля поклёвки при ловле хищной рыбы на жерлицы. Благодаря использованию механического концевого выключателя, Arduino и Qt-приложения, система своевременно оповещает рыбака о срабатывании снасти, избавляя от необходимости постоянно следить за лунками в морозную погоду.

Ключевые достижения:

1. Работоспособный прототип – система стабильно фиксирует поклёвку и передаёт данные на компьютер.
2. Простота и доступность – использованы недорогие и распространённые компоненты.
3. Удобный интерфейс – приложение позволяет легко подключиться к Arduino и отслеживать состояние жерлицы.

Этот проект — отличный пример того, как современные технологии могут упростить традиционные виды деятельности. В будущем систему можно дорабатывать, делая её ещё более удобной и функциональной.

Гордимся тем, что:

1. Создали практически полезное устройство для реальных условий.
2. Использовали минимальные ресурсы для достижения результата.
3. Заложили основу для более сложных модификаций.

**Список использованных источников**

1 Страуструп Б. Язык программирования С++. – 1997 - №1 – С.76.

2 Страуструп Б. Программирование. Принципы и практика использования C++. – 2011 - №1 – С. 279.

3 Кнут Д.Э. Искусство программирования. – 1968 - №1 – С. 98.

4 Керниган Б. Язык программирования С. – 1978 - №1 – С. 116.

# 5 Сложность алгоритмов и программ. –URL: https://cyberleninka.ru /article /n/ slozhnost-algoritmov-i-programm (дата обращения 15.05.2024).