

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СИСТЕМЫ ДИНАМИЧЕСКОЙ ТЕРМОРЕГУЛЯЦИИ ДЛЯ НОСИМОЙ ЭЛЕКТРОНИКИ

DEVELOPMENT OF SOFTWARE FOR DYNAMIC THERMOREGULATION SYSTEM FOR WEARABLE ELECTRONICS

**R. Manaev
I. Vasiliev**

Summary. The paper discusses the development of software that implements the control of the dynamic thermoregulation system, which provides the function of maintaining the microclimate for wearable electronics. The aim of the work is to develop software for the rapid prototyping of the hardware and software complex of the dynamic thermoregulation system. The developed software package has no analogs with open source code at the time of publication. The elements of the novelty of the presented solution are the growing popularity of wearable electronics in general, as well as the relatively recent definition of a new segment of the wearable electronics market — smart clothes, the first products of this segment presented to the public. The structure of the developed software is described. An overview of the program's graphical interface is given.

Keywords: smart clothes, wearable electronics, software, thermoregulation.

Манаев Руслан Гайратович

ФГБОУ ВО «Уфимский Государственный Авиационный
Технический Университет», г. Уфа
manavrion@gmail.com

Васильев Игорь Сергеевич

ГБПОУ «Уфимский Колледж Радиоэлектроники,
Телекоммуникаций и Безопасности» г. Уфа;
stormcreator@inbox.ru

Аннотация. В статье рассматривается разработка программного обеспечения, реализующая управление системой динамической терморегуляции, обеспечивающей функцию поддержания микроклимата для носимой электроники. Целью работы является разработка программного обеспечения для быстрого прототипирования программно-аппаратного комплекса системы динамической терморегуляции. Разработанный программный комплекс не имеет аналогов с открытым исходным кодом на момент публикации. Элементами новизны представленного решения являются возрастание популярности носимой электроники в целом, а также относительно недавнее определение нового сегмента рынка носимой электроники — умной одежды, первых представленных общественности продуктов этого сегмента. Описана структура разработанного программного обеспечения. Приведен обзор графического интерфейса программы.

Ключевые слова: умная одежда, носимая электроника, программное обеспечение, терморегуляция.

Введение

Повседневная верхняя одежда не обладает необходимой эффективностью в городских условиях, автоматически не адаптируется к среде с динамически меняющейся температурой, что исключает поддержку комфортного для пользователя микроклимата. Макросистема терморегуляции организма [1] очень сложна и имеет множество недостатков, связанных с индивидуальностью телесных характеристик каждого отдельного человека. Комфортная температура окружающей среды — один из важных факторов, обеспечивающих безопасность жизнедеятельности человека. Человечество достигло определенных успехов в создании стабильной и комфортной температуры внутри жилых помещений. Однако на улице всё по-прежнему зависит от погоды и развитости городских инфраструктур. В множестве городов современная инфраструктура не обеспечивает необходимого человеку комфортного температурного режима. Частый перепад температур является стрессом для организма человека, который способствует ухудшению самочувствия и потере общей

эффективности (во время осуществления человеком активной умственной и физической деятельности), развитию раздраженности и, как следствие, повышению вероятности проявления разного вида агрессии, а также увеличивает риск заболевания гриппом, ОРВИ, а также осложнения текущих заболеваний [2].

Необходимость разработки одежды нового поколения обусловлена, в том числе, возрастающим глобальным спросом на носимую электронику. Умная одежда, как разновидность носимой электроники, занимает не существенную долю рынка (не более 1,3% за 2019 год, что является 4,2 млн. штук, согласно данным IDC [3]), при этом оставаясь одним из наиболее перспективных направлений для развития стартапов.

Комплекс поддержки микроклимата (далее КПМ) — это программно-аппаратный комплекс индивидуально-пользования, с функцией регулировки температуры под верхней одеждой пользователя. В КПМ реализован сбор данных с температурных датчиков, расположенных достаточно близко, чтобы проводить мониторинг

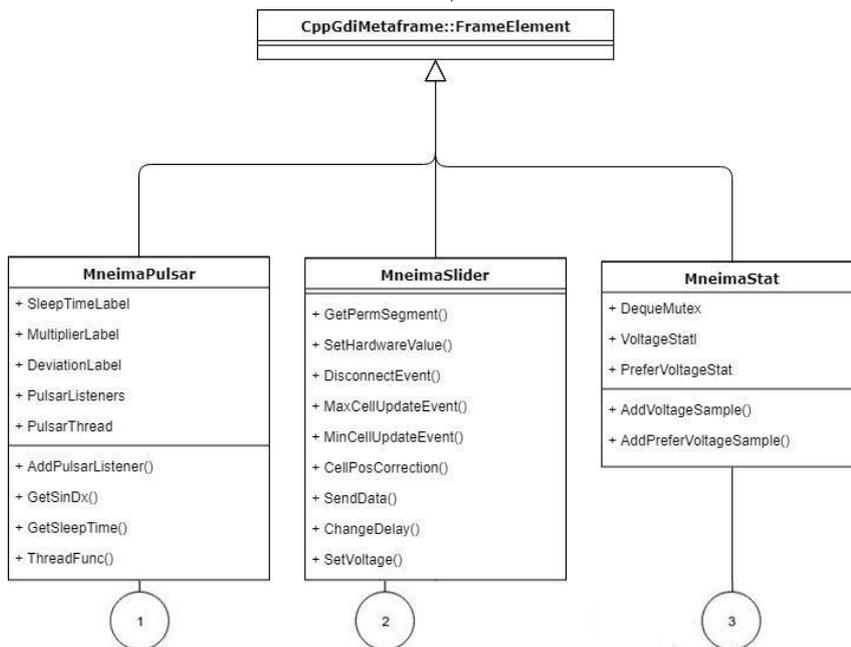


Рис. 1. UML диаграмма классов приложения (1/2).

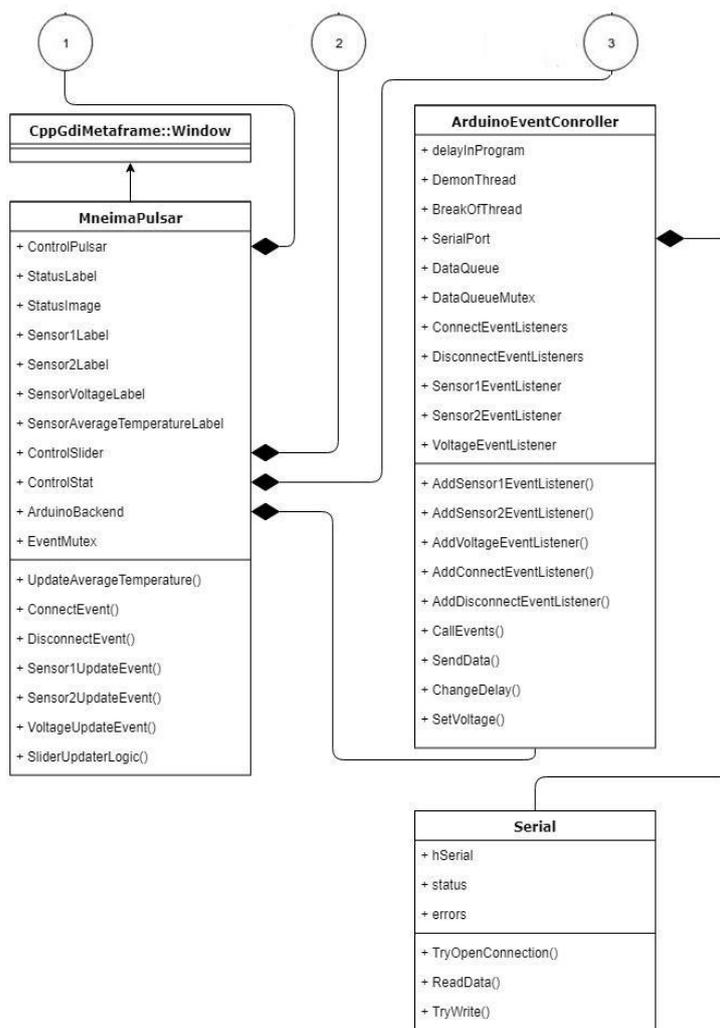


Рис. 2. UML диаграмма классов приложения (2/2).

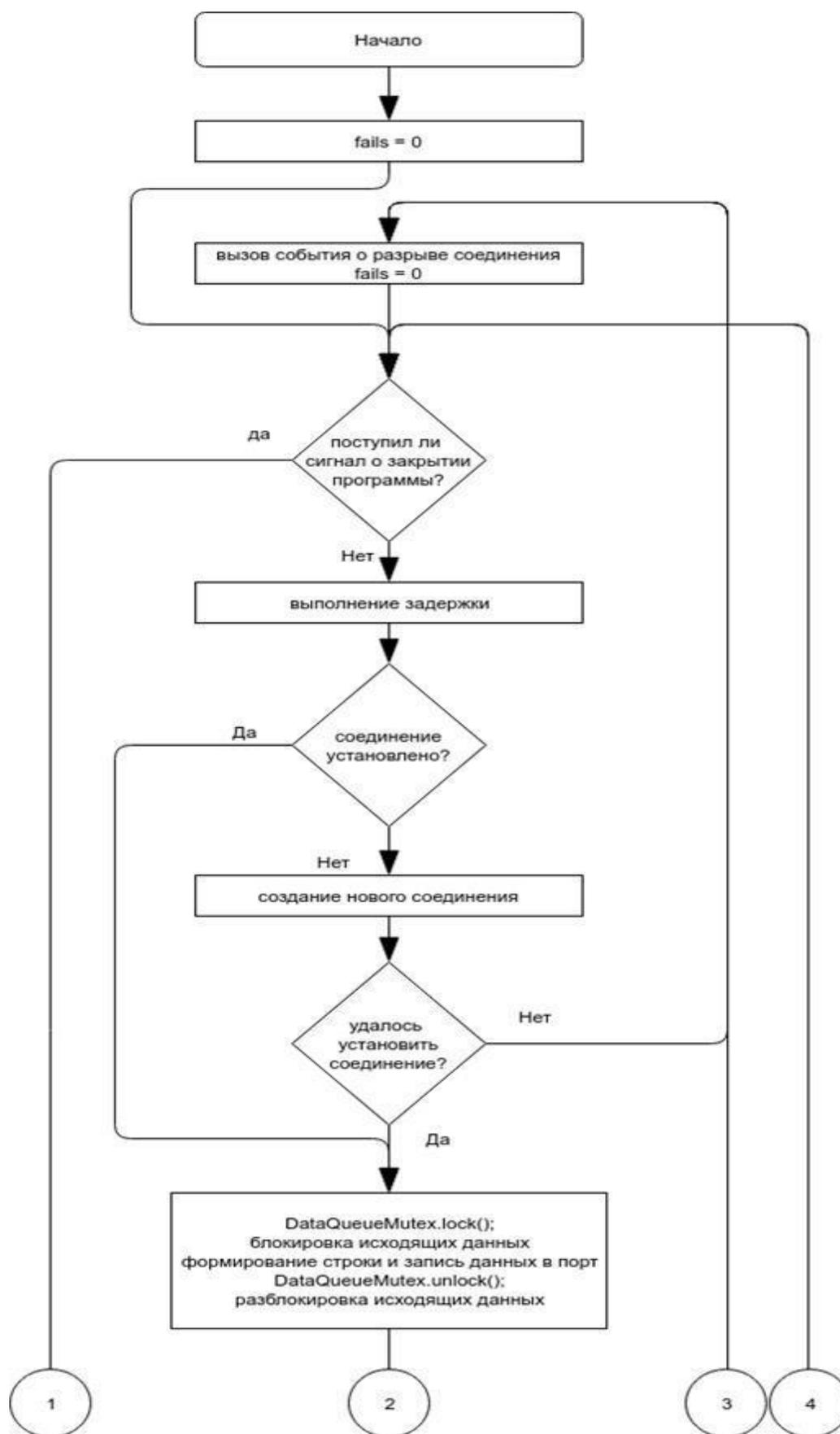


Рис. 3. Блок-схема взаимодействия приложения с микроконтроллером (1/4).

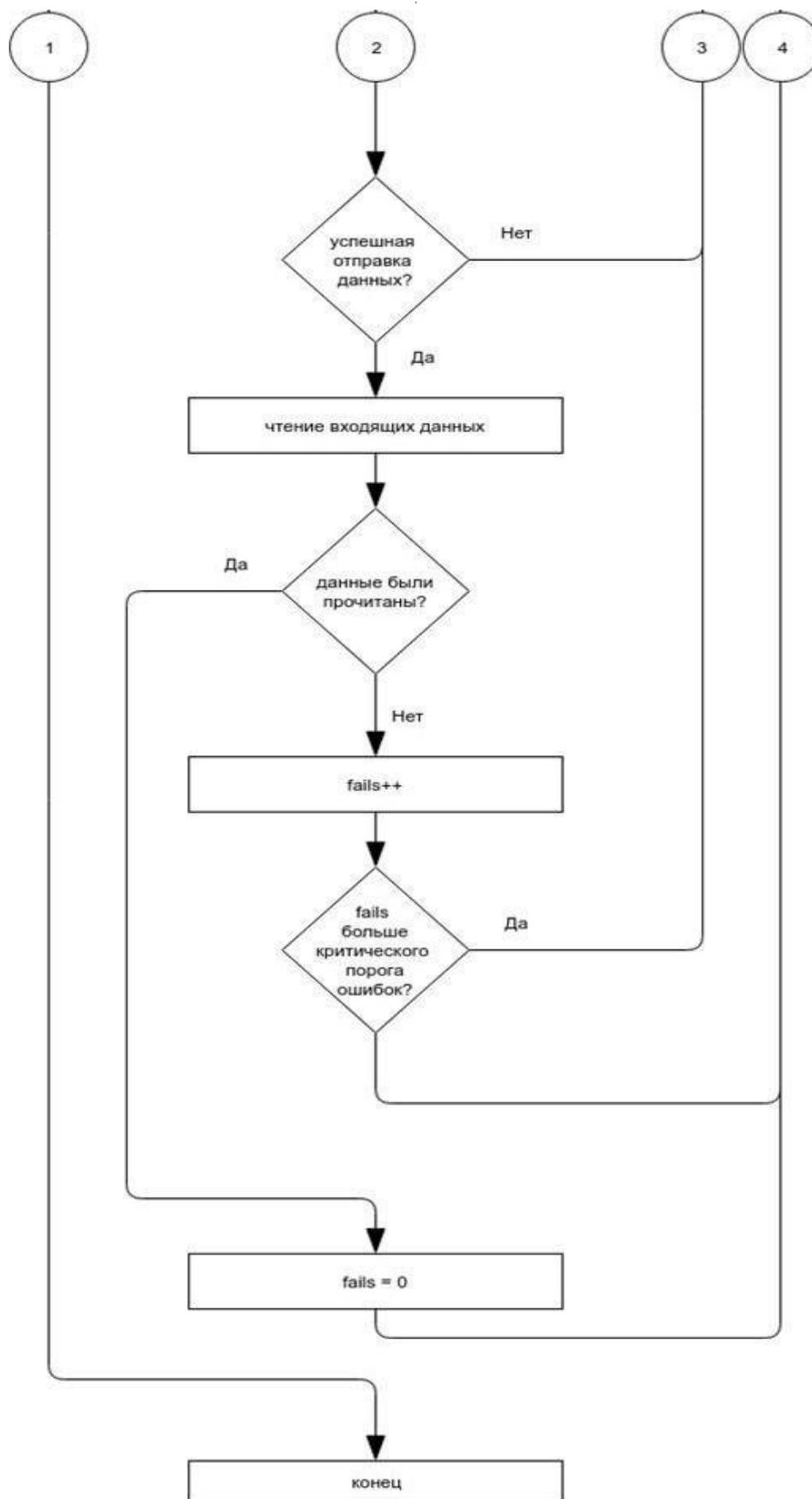


Рис. 4. Блок-схема взаимодействия приложения с микроконтроллером (2/4).

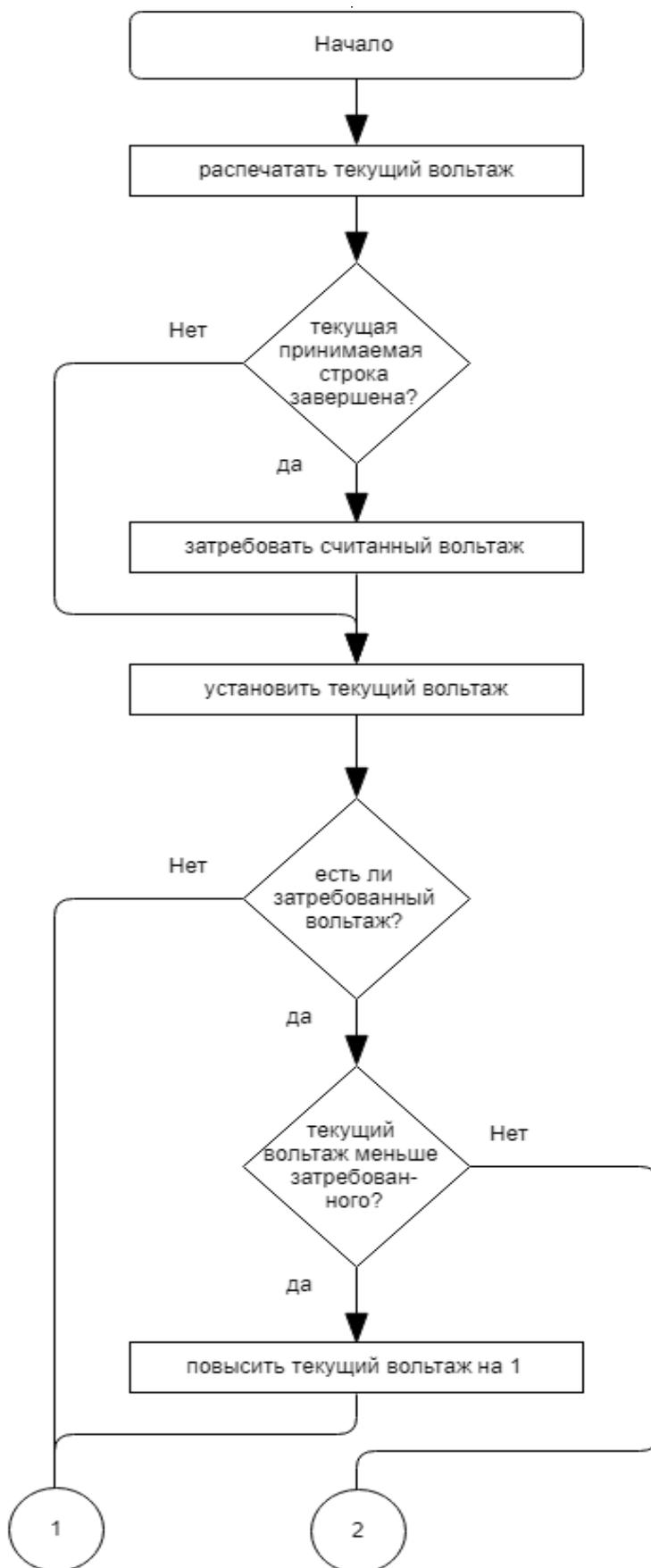


Рис. 5. Блок-схема взаимодействия приложения с микроконтроллером (3/4).

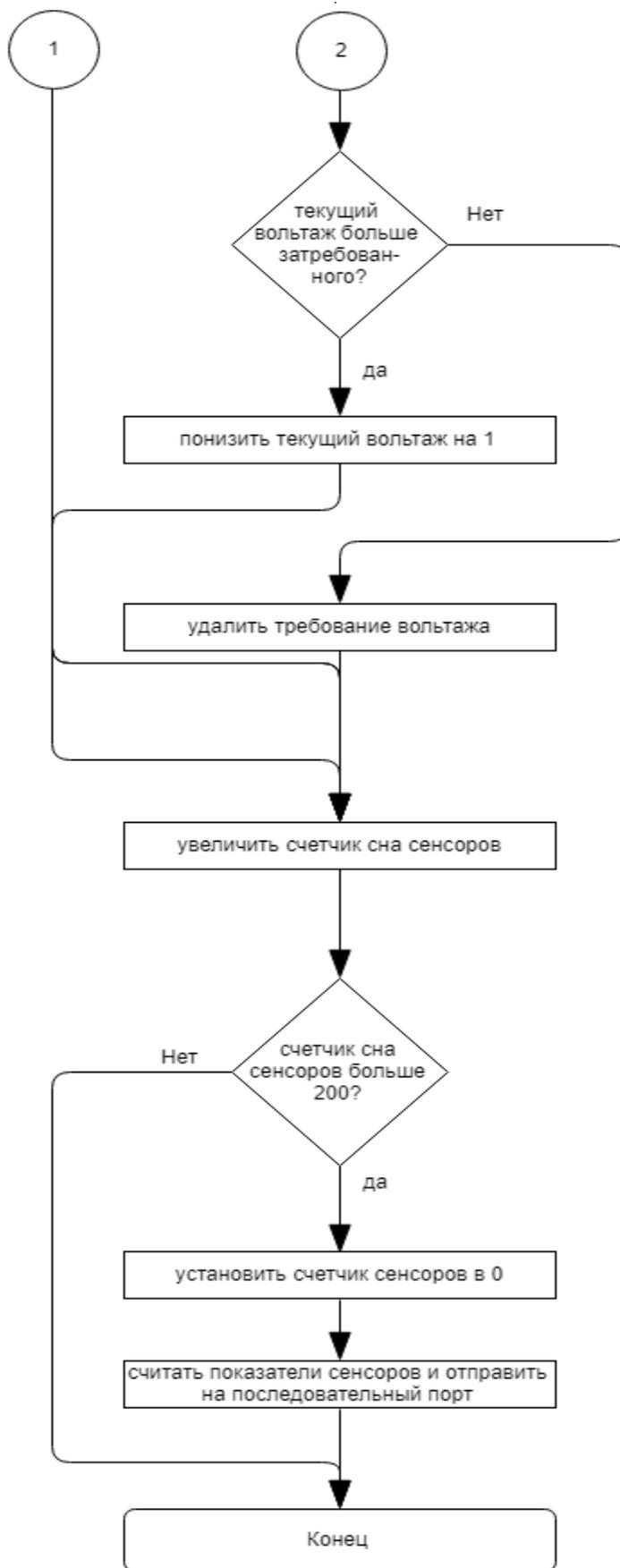


Рис. 6. Блок-схема взаимодействия приложения с микроконтроллером (4/4).

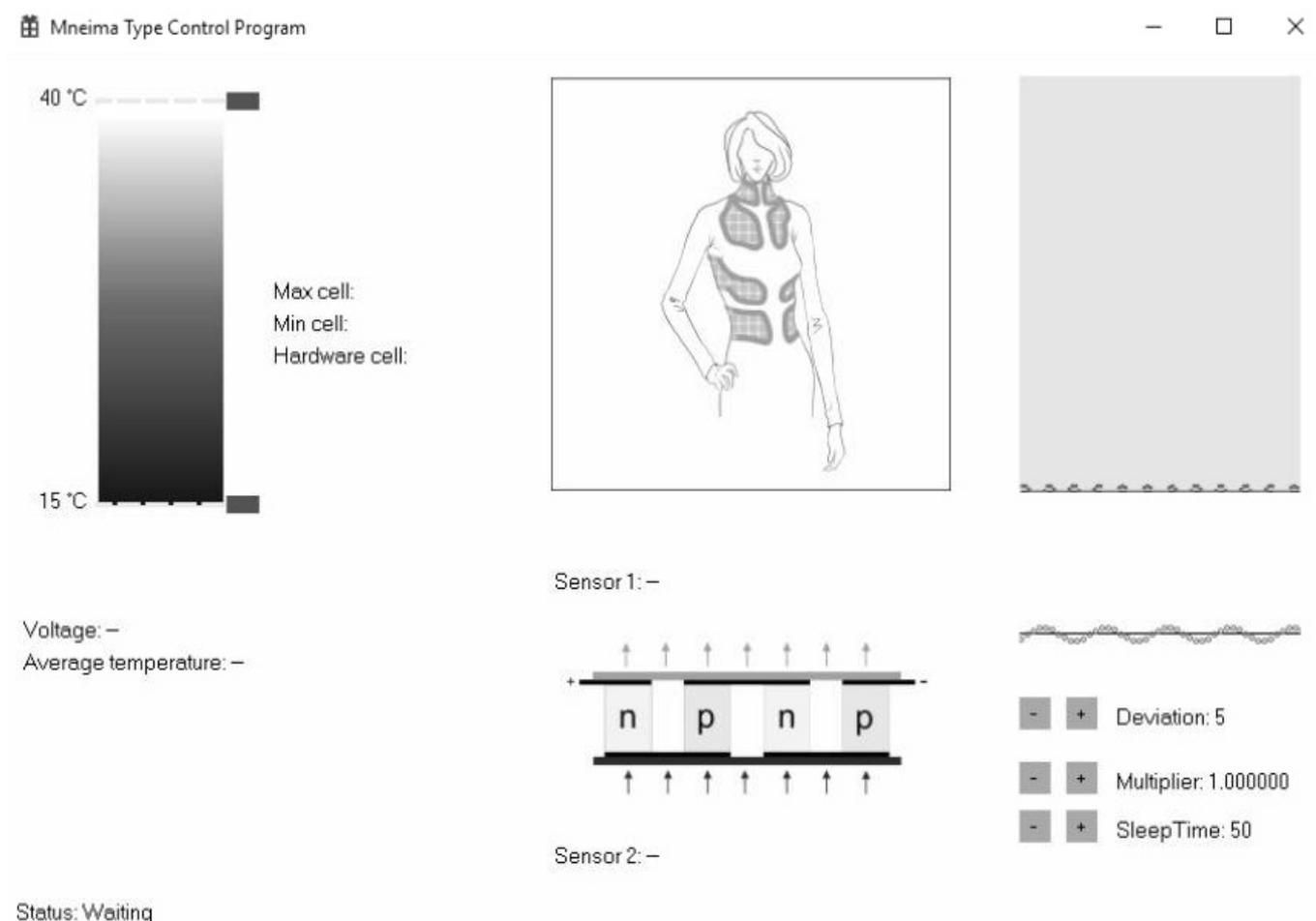


Рис. 7. Графический интерфейс приложения

кожной температуры человека [4]. КПМ — это развивающаяся технология, применимая в различных областях и индустриях для решения широкого круга задач. Наиболее распространенными примерами активного терморегулирования являются предметы гардероба со встроенными нагревательными элементами, вентиляторами, а также системами водяного охлаждения. В системе динамической терморегуляции (далее — СДТ) присутствует ранее не используемый в аналогичных разработках элемент Пельтье [5]. Выбор элемента в качестве основы для СДТ обусловлен его функциональной особенностью осуществлять как нагрев, так и охлаждение без применения вентиляторов и жидкостей.

Описание структуры приложения

На этапе проектирования программного обеспечения была разработана UML диаграмма классов [6], а также блок-схема процесса взаимодействия с микроконтроллером ATmega328 на плате Arduino (далее микроконтроллер). Диаграмма классов (Рис. 1 и 2) отражает устройство основного программного обеспече-

ния (далее приложение), через которое производится управление системой терморегуляции.

Блок-схема на рисунках 3 и 4 отражает алгоритм синхронизации данных между приложением и программным обеспечением, установленным в микроконтроллер.

Обзор графического интерфейса

Графический пользовательский интерфейс приложения показан на рис. 7, разработан с использованием библиотеки «C++GdiMetaframe» [7]. Библиотека «C++GdiMetaframe» позволяет легко создавать окно приложения и GDIPlus контекст для рисования в окне.

Большинство компонентов и классов программы активно используют паттерн объектно-ориентированного программирования, называемый наблюдателем, он включает в себя такие понятия, как «слушатель» и «генератор событий». Слушателем обычно выступает функция объекта, который хочет внедриться в соответствующий генератор событий.

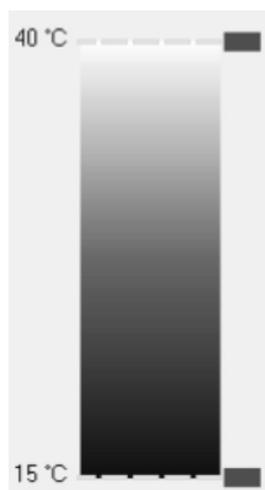


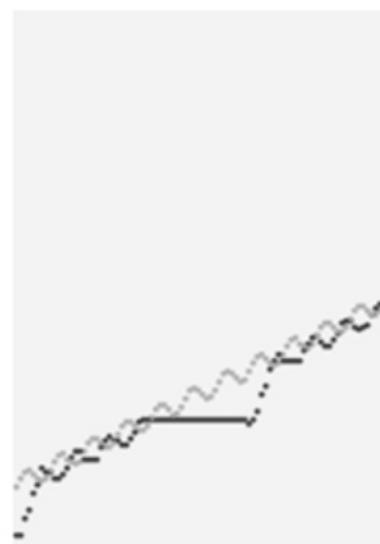
Рис. 8. Слайдер

В программе инициализируется класс `ArduinoBackend`, отвечающий за поддержку связи с микроконтроллером. В `ArduinoBackend` отправляются необходимые слушатели компонентов интерфейса, реализованные в виде методов класса `MneimaWindow`.

Класс `MneimaWindow` содержит компоненты интерфейса для удобного взаимодействия со слушателями событий. Некоторые классы наследуются от `CppGdiMetaframe::FrameElement`, что позволяет им легко участвовать в иерархии графических элементов `CppGdiMetaframe::Window`. Создание каждого компонента состоит из выделения памяти для этого компонента и последующего помещения указателя на данную память в умный указатель (`std::unique_ptr`), который является полем класса `MneimaWindow`. После создания компонента выполняется его конфигурирование с помощью так называемых сеттеров (методов, начинающихся с префикса `Set`). Далее производится помещение указателя на данный элемент в `MneimaWindow` с помощью метода `Add()`.

Компоненты «Статусная строка» (`StatusLabel`) и «Статусная картинка» (`StatusImage`) используются для оповещения пользователя о статусе соединения с микроконтроллером. Статусная строка может принимать состояния ожидания, ошибки соединения и успешного соединения с устройством.

Статусная картинка при успешном соединении принимает цветное изображение, а при разрыве — черно-белое. Данные действия реализованы в методах `MneimaWindow::ConnectEvent()` и `MneimaWindow::DisconnectEvent()`, которые помещаются в качестве слушателей в `ArduinoBackend`. При событии разрыва соединения также сбрасываются показатели всех сенсоров.



- + Deviation: 5

- + Multiplier: 1.000000

- + SleepTime: 50

Рис. 9. Пульсар

Показатели сенсоров также имеют соответствующие слушатели вида `MneimaWindow::Sensor*UpdateEvent()`, которые помещаются также в `ArduinoBackend`. Для того, чтобы не допустить состояния гонки [8] слушателей, в них внедрен мьютекс `EventMutex`, таким образом повышается потокобезопасность, что обеспечивает отказоустойчивость приложения.

Поскольку человек не способен точно определить температуру, так как полагается на собственные органы чувств и различает температуру лишь в относительном диапазоне значений «горячо-холодно», без применения абсолютных значений градации, было принято решение создать уникальный вид графического манипулятора. С помощью слайдера, изображенного на рис. 8, пользователь может устанавливать максимальную и минимальную температуру, смещая манипуляторы вверх-вниз по графической шкале, отображающей температурный градиент. Эффективность применения в данном случае

не зависит от абсолютных значений и обусловлена субъективной оценкой пользователя.

Ввиду отсутствия стандартного графического компонента, необходимого для обеспечения надлежащего управления, был реализован класс `MneimaSlider`.

Часть `back-end` разработана для микроконтроллера, который является компонентом системы динамической терморегуляции. Для программного управления СДТ необходима синхронизация данных в микроконтроллере, и программы, с помощью которой пользователь может контролировать подаваемый на микроконтроллер управляющий сигнал с компьютера. Без осуществления передачи данных между приложением и управляющей программой на микроконтроллере, управление системой динамической терморегуляции невозможно.

Наиболее эффективным в случае использования элемента Пельтье является метод подачи питания на элемент импульсно, без резких переключений (например, при непосредственном использовании ШИМ [9]). Для осуществления этого разработан компонент Пульсар, который показан на рис. 9. Он генерирует соответству-

ющее событие управляющего сигнала, представляющее собой дискретную синусоиду. Контроллеры Пульсара позволяют менять время такта (`sleep time`), размах синусоиды (`Deviation`), ширину синусоиды или время её квантования (`Multiplier`).

Данный компонент также демонстрирует синусоиду на графике.

Событие управляющего сигнала позволяет модифицировать прямой сигнал с помощью синусоиды, что обеспечивает импульсность сигнала и повышает эффективность элемента Пельтье.

В итоге разработка представленного программного обеспечения [10] существенно упрощает процесс прототипирования автоматизированной системы управления комплекса поддержки микроклимата, обеспечивая необходимыми данными о работе элемента Пельтье. Полученные данные ускорят усовершенствование системы динамической терморегуляции, а также её внедрение в большее количество «умной одежды», что способствует увеличению объема данного сегмента носимой электроники.

ЛИТЕРАТУРА

1. Богатов Н.М., Григорьян Л.Р., Лыжко Е.В. Макросистема терморегуляции организма человека // *Современные наукоемкие технологии*, 2008, № 2, с. 110–111.
2. Колосов А.С., Прошин А.В. Зависимость заболеваемости острыми инфекциями дыхательных путей от суровости погоды в зимний период года (на примере города Кирова) // *Международный научно-исследовательский журнал*, 2016, № 1 (43), Часть 3, с. 51–55.
3. Shirer M., Llamas R., Ubrani J. Shipments of Wearable Devices Reach 118.9 Million Units in the Fourth Quarter and 336.5 Million for 2019, According to IDC // *International Data Corporation (IDC)*. 10.03.2020. URL: <https://www.idc.com/getdoc.jsp?containerId=prUS46122120>
4. (дата обращения: 01.11.2020).
5. Богданова Т.М., Бакуткин В.В., Большаков А.А., Бакуткин И.В., Мельников Л.А., Спиринов В.Ф., Наливаева А.В. Мониторинг кожной температуры тела человека и его применение в клинической практике // *Международный журнал экспериментального образования*, 2013, № 10–2, с. 242–245.
6. Макаров Д. Что такое элемент Пельтье, его устройство, принцип работы и практическое применение // *Заметки Электрика*. URL: <https://www.asutpp.ru/cto-takoe-element-pelte-i-ego-primenenie.html> (дата обращения: 01.11.2020).
7. Коптенок Е.В., Трунников М.В., Сухарев Е.А. Применение UML-диаграмм для проектирования программных комплексов // *Молодой ученый*, 2020, № 19 (309), с. 133–135.
8. Манаев Р.Г. `CppGdiMetaframe` // *GitHub*. URL: <https://github.com/manavrion/CppGdiMetaframe> (дата обращения: 25.10.2020).
9. Клещев А.С., Крылов Д.А. Корректность результата вычислений в условиях состояния гонки // *Информатика и системы управления*, 2013, № 3(37), с. 99–109;
10. Гринкевич В.А. Синтез регулятора тока для элемента Пельтье // *Сборник научных трудов НГТУ*, 2018, № 3–4 (93), с. 16–39.
11. Манаев Р.Г. `mneima_type_control_program` // *GitHub*. URL: https://github.com/manavrion/mneima_type_control_program (дата обращения: 01.11.2020).
12. Васильев И.С., Рафиков Д.И. Анализ существующих нагревательных элементов, интегрируемых в верхнюю одежду // *Телекоммуникационное оборудование российского происхождения: проблемы и перспективы. Сборник тезисов докладов Открытой региональной научно-практической конференции*, 2017, с. 34–36.
13. Корнева Е.А., Казакова Т.Б. Современные подходы к анализу влияния стресса на процессы метаболизма в клетках нервной и иммунной систем // *Медицинская иммунология*, 1999, № 1–2.