Разработка портативного устройства и комплекса программ для тестирования и настройки многофункциональных аварийных вычислителей-регистраторов

Андрей Васенев (vasenev-aa@yandex.ru, РТУ МИРЭА)

Многофункциональные аварийные вычислители-регистраторы используются для регистрации параметров состояния поезда метрополитена и являются комплексными изделиями. В процессе их тестирования и настройки применяется специализированное ПО и внешние аппаратные средства, однако некоторые процессы взаимодействия с регистратором могут быть автоматизированы и реализованы в отдельном портативном устройстве. В статье описывается разработка и реализация такого устройства и комплекса программ для его работы.

Введение

Многофункциональный аварийный вычислитель-регистратор производится АО «НИИВК им. М.А. Карцева» и представляет собой малогабаритное бортовое устройство, предназначенное для регистрации параметров состояния поезда метрополитена и сохранения зарегистрированной информации в рабочих и аварийных условиях. Регистратор рассчитан на применение в головных вагонах поездов метрополитена и применяется совместно с прочими системами регистрации данных, которые обеспечивают считывание, обработку, долговременное хранение и отображение зарегистрированной информации.

Регистратор выполняет следующие основные функции:

- принимает сигналы, характеризующие значения параметров состояния поезда, и переводит их в цифровую форму в соответствии с заданными правилами;
- записывает принятые данные в накопитель;
- отображает своё состояние и состояние входных каналов при помощи светодиодных индикаторов;
- обеспечивает возможность считывания записанных данных в персональный компьютер (ПК) при его подключении к регистратору посредством интерфейса Ethernet.

При этом использование подключённого к регистратору ПК не ограничивается возможностью считывания записанных данных и применяется также для считывания диагностической информации и настройки. Для решения таких задач используется специализированное ПО.

Проектирование устройства

При анализе процессов такого взаимодействия с регистратором были выявлены задачи, нуждающиеся в автоматизации, и методы, которые могут быть усовершенствованы, – это задача анализа диагностических журналов, формируемых встроенной программой регистратора и используемых на некотором этапе их тестирования, а также метод установки актуальных даты и времени. Для сокращения времени тестирования многофункциональных аварийных вычислителей-регистраторов и совершенствования метода установки даты и времени было разработано и впоследствии реализовано портативное устройство, а также комплекс программ. В комплекс программ входит встроенная программа работы устройства и прикладная программа, предназначенная для обмена данными с устройством через персональный компьютер (ПК).

Устройством решаются следующие задачи:

- проверка наличия соединения с регистратором;
- загрузка диагностических журналов на внутренний накопитель устройства;
- автоматический анализ загруженных диагностических журналов;
- выгрузка диагностических журналов с накопителя устройства на ПК;
- установка точных даты и времени в регистратор при помощи GPS;
- предоставление человеко-машинного интерфейса для взаимодействия с устройством.

В качестве платы центрального микроконтроллера устройства была выбрана Raspberry Pi Pico с 26-контактным интерфейсом ввода/вывода (GPIO), поддерживающая такие протоколы, как SPI, I^2C и UART. Отличительной особенностью платы по сравнению с прочими решениями является встроенная поддержка программирования контроллера на языке MicroPython. Это интерпретируемый язык с синтаксисом стандартного языка Python, созданный специально для работы на микроконтроллерах. Язык прост в освоении и позволяет быстро и эффективно разрабатывать как прототипы программ, так и финальные сборки. Объём и читаемость исходного кода программных модулей значительно снижает сложность программирования, снижая вероятность допущения ошибок и повышая тем самым надёжность и качество работы портативного устройства.

Многофункциональность устройства обусловливает использование нескольких модулей, подключаемых к плате центрального микроконтроллера, а именно: модуля часов реального времени, модуля канала Ethernet, модуля GPS и модуля бесперебойного питания. В качестве внутреннего накопителя устройства была выбрана карта памяти MicroSD, размещающаяся в соответствующем слоте на плате памяти. Для эффективного взаимодействия с устройством было решено использовать OLED-дисплей с небольшой диагональю и индикационные светодиоды; для управления устройством применяются две

тактовые кнопки и переключатель питания.

Проектирование интерфейсной печатной платы

При анализе состава устройства взаимодействия с регистратором было решено спроектировать интерфейсную печатную плату, выполняющую следующие задачи:

- обеспечение электрического соединения модулей и платы центрального микроконтроллера;
- крепёж модулей и элементов человеко-машинного интерфейса.

При проектировании взаимного расположения модулей и элементов устройства определяющими факторами стали их габаритные размеры и возможности по подключению. При создании эскиза расположения компонентов устройства в первую очередь использовалась документация на компоненты от производителя-поставщика. Были определены общие габаритные размеры проектируемой печатной платы.

Печатная плата была спроектирована с использованием отечественной САПР Delta Design компании Eremex. Была создана библиотека используемых в устройстве компонентов, разработаны принципиальная схема и непосредственно интерфейсная печатная плата. В процессе формирования библиотеки компонентов использовалась документация (даташиты) на компоненты и ГОСТ 2.743-91 единой системы конструкторской документации «Обозначения условные графические в схемах». При разработке электрической схемы применялся ГОСТ 2.702-2011 единой системы конструкторской документации «Правила выполнения электрических схем». При проектировании печатной платы были учтены ограничения технологических возможностей будущего изготовителя печатной платы – компании «Резонит» с производственной линией в г. Москве.

Спроектированная печатная плата была экспортирована в формате производственных Gerber-файлов и направлена на изготовление.

Разработка комплекса программ

В состав разработанного комплекса программ, как было сказано выше, входят встроенная программа и прикладная. Встроенная программа представляет собой интерпретируемый код на языке MicroPython и предназначена для загрузки в портативное устройство для обеспечения решения им всех поставленных задач. Архитектура программы представляет собой набор программных модулей, разделённых по назначению:

- считывание и обработка сигнала GPS;
- обмен данными по каналу Ethernet;
- автоматический анализ диагностических журналов регистратора;
- обработка действий оператора и отображение информации;
- передача команд отображения на дисплей;
- чтение и запись на карту памяти MicroSD;
- считывание уровня заряда аккумулятора и потребляемого тока;
- главный цикл работы портативного устройства.

При включении устройства запускается главный цикл работы, который абстрагирован от деталей реализации конкретного физического модуля или функционала и оперирует высокоуровневыми методами, предоставляемыми прочими программными модулями. На дисплей выводится меню возможных действий с устройством, переключение и выбор пунктов меню осуществляется тактовыми кнопками. По окончании выполнения выбранной функции на дисплей выводится сообщение о результате, сопровождаемое соответствующим светодиодом. На протяжении всего сеанса работы устройства работают светодиод уровня заряда и статуса аккумулятора (> 15% заряда, ≤ 15% заряда, заряжается), а также светодиод, сигнализирующий об актуальности времени, отсчитываемого в устройстве.

Прикладная программа представлена в виде меню взаимодействия с подключённым по USB устройством, открывающимся через системный трей ОС Windows. Прикладная программа позволяет выполнить программное подключение и отключение портативного устройства, загрузку диагностических журналов с накопителя устройства в выделенную директорию на ПК, а также открыть окно настройки параметров взаимодействия с устройством. Программа написана на объектно-ориентированном языке C# с использованием интерфейса программирования приложений Windows Forms.

Реализация проекта

По окончании разработки устройства и закупки всех необходимых вычислительных модулей был создан макетный образец, который использовался в процессе разработки и отладки комплекса программ.

По завершении программирования и изготовления интерфейсной печатной платы был собран и протестирован демонстрационный образец устройства. После завершения этапа проектирования и изготовления корпуса портативное устройство может быть применено на предприятии АО «НИИВК им. М.А. Карцева» для тестирования и настройки многофункциональных аварийных вычислителей-регистраторов, повысив эффективность работы отдела встраиваемых электронных систем.

Литература

1. Документы по САПР Delta Design: [Электронный ресурс] // URL: https://www.eremex.ru/knowleage-base/delta-design/docs/ (дата обращения: 21.03.2022).
2. *Лапин А.А.* Интерфейсы. Выбор и реализация. М.: Техносфера, 2005. 168 с.
3. *Макконнелл С.* Совершенный код. Мастер-класс / пер. с англ. СПб.: БХВ, 2018. 896 с.
4. *Одуан К.* Измерение времени. Основы GPS / К. Одуан, Б. Гино. М.: Техносфера, 2002. 400 с.
5. Резонит – Технологические возможности производства: [Электронный ресурс] // URL: https://www.rezonit.ru/directory/tekhnologicheskie-osobennosti-proizvodstva/ (дата обращения: 04.04.2022).
6. Требования к проектированию печатных плат: [Электронный ресурс] // URL: https://gostost.ru/pechatnie-platy/ (дата обращения: 05.04.2022).
7. MicroPython: [Электронный ресурс] // URL: https://micropython.org/ (дата обращения: 26.04.2022).
8. Raspberry Pi Documentation: [Электронный ресурс] // URL: https://www.raspberrypi.com/documentation/microcontrollers/rp2040.html (дата обращения: 15.02.2022).
9. Waveshare Wiki: [Электронный ресурс] // URL: https://www.waveshare.com/wiki/Main_Page (дата обращения: 24.02.2022).