

Е. Н. БАЗЫЛЕВ

П. Ю. БРАНЦЕВИЧ, *д-р. техн. наук, доц.*

Белорусский государственный университет информатики и радиоэлектроники
(Минск, Беларусь)

ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ ВСТРОЕННЫХ СИСТЕМ В ВИБРОИЗМЕРИТЕЛЬНЫХ ПРИБОРАХ

Аннотация. Для обнаружения редких кратковременных изменений структуры вибрационного сигнала и выявления причинно-следственных связей между их появлением и развитием дефектов контролируемых объектов требуется создавать приборы, способные накапливать и обрабатывать длительные непрерывные вибрационные сигналы. Рассмотрены структура, функциональные возможности, программное обеспечение регистратора вибрационных сигналов на встроенном процессоре.

Ключевые слова: вибрация, сигнал, накопление, встроенная система, Arduino.

При анализе существующих аналогов накопителей сигналов было установлено, что использование SD Card (Secure Digital – компактный портативный флеш-накопитель) в качестве накопителя данных позволяет записывать сигналы длительностью значительно больше, чем одни сутки, имея при этом низкое энергопотребление. Поэтому для решения задачи накопления вибросигнала на протяжении длительных временных интервалов представляется возможным создавать устройства на базе современных микроконтроллеров [1, 2].

В состав микроконтроллеров, как правило, входит аналого-цифровой преобразователь (АЦП), запускаемый аппаратно таймером. При этом нет необходимости обрабатывать прерывание от таймера. Запрограммировав период повторения таймера, достаточно обрабатывать прерывание от АЦП с последующей буферизацией кода с его выхода. Буферизируемые данные перемещаются в накопитель данных, не допуская переполнения. В качестве накопителя используется энергонезависимая память или, в лучшем случае, носитель с файловой системой.

Современные встроенные системы оснащены часами реального времени, что позволяет устройству работать независимо длительное время без внешнего вмешательства. Кроме того, микроконтроллеры имеют последовательные и параллельные интерфейсы для ввода-вывода данных. Посредством интерфейсов ввода-вывода возможна организация передачи данных для дальнейшей обработки на современные multifunctional мобильные либо настольные платформы с последующим отображением данных в табличном и графическом виде.

Через микроконтроллерные интерфейсы осуществляется управление процессом накопления цифрового вибросигнала, задание коэффициентов передачи и усиления, частоты дискретизации, размера выборки, времени

запуска и останова накопления. Для контроля скорости передачи и целостности данных обмен данными организуется по принципу запрос–ответ.

Поток данных, передаваемый посредством интерфейса ввода вывода, разделяется на пакеты данных, каждый из которых должен заканчиваться контрольной суммой (хэш-значение) для контроля целостности передачи данных. Первые два байта такого пакета – идентификатор (директива) и номер пакета.

При разработке программного обеспечения используются аппаратно-независимые языки программирования. Для упрощения сопровождения программного проекта, а также минимизации трудозатрат при переходе на другую микроконтроллерную платформу, архитектура программного средства встроенной системы подразделяется на два уровня – аппаратно-зависимый и независимый. Следовательно, при переходе на другой микроконтроллер требуется переработка только аппаратно-зависимого уровня программного обеспечения.

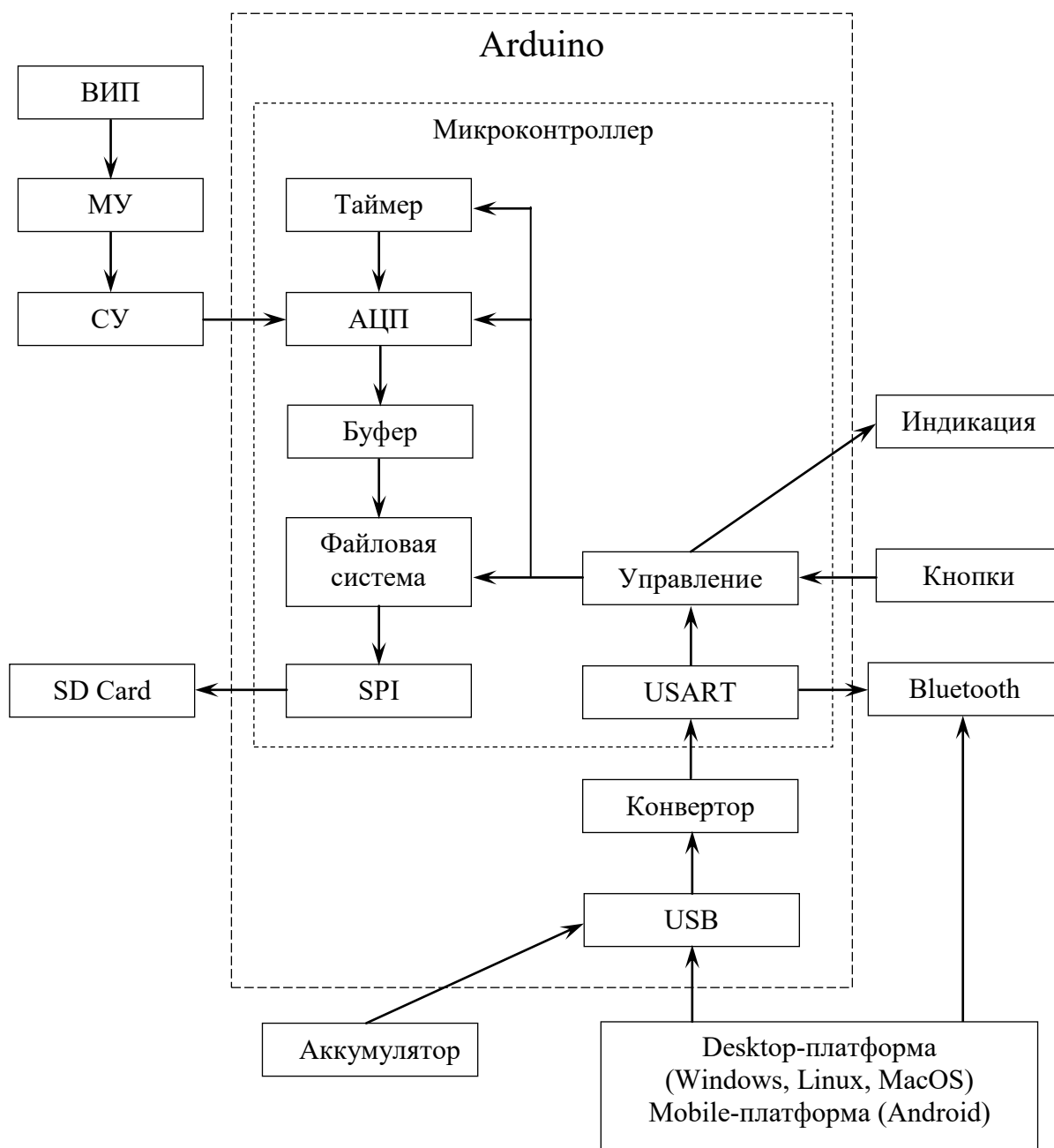
Передача потока данных на мобильную или настольную платформу организуется независимо от конкретного интерфейса обмена данными, а обработка данных на мобильной или настольной платформе осуществляется с использованием технологии, упрощающей перенос программного кода на другие операционные системы, а также в интернет.

Разработан прототип системы регистрации длительных во времени вибрационных сигналов (рис. 1). Встроенная система использует 8-битный микроконтроллер ATmega фирмы Microchip [3], который имеет встроенный 10-битный АЦП, управляемый 8-битным или 16 битным таймером.

В качестве накопителя данных применяется SD Card [4], управляемый с помощью SPI интерфейса, аппаратный модуль которого уже встроен в микроконтроллер. SPI (Serial Peripheral Interface) – это последовательный синхронный интерфейс для связи между микроконтроллерами и периферийными устройствами (датчиками, дисплеями, памятью). Он обеспечивает высокую скорость и полнодуплексную передачу данных (одновременный прием и отправку).

Значительная часть вычислительной мощности микроконтроллера задействована для управления файловой системой, для чего использована популярная в интернете библиотека FatFs [5]. FatFs – это популярная универсальная библиотека с открытым исходным кодом, написанная на языке Си, которая используется для реализации файловых систем FAT (File Allocation Table) и exFAT во встраиваемых системах и микроконтроллерах. Она позволяет легко организовать чтение, запись и хранение файлов на SD-картах, USB-флешках или внешней Flash-памяти независимо от используемого оборудования.

Управление встроенной системой осуществляется посредством популярного интерфейса последовательного порта USART, благодаря которому происходит обмен данными через ноль-модем либо модуль Bluetooth.



ВИП – виброизмерительный преобразователь; МУ – масштабный усилитель; СУ – согласующее устройство; SPI – Serial Peripheral Interface (стандарт последовательной передачи данных); USART – Universal Synchronous-Asynchronous Receiver-Transmitter (универсальный синхронно-асинхронный приемопередатчик); USB – Universal Serial Bus (универсальный стандарт для передачи данных и питания между устройствами).

Рис. 1. Архитектура устройства накопления данных

Для контроля целостности пакетной передачи данных применяется циклическая избыточная проверка CRC16 (Cyclic Redundancy Check 16 bits – циклический избыточный код, 16 бит), обладающая хорошими свойствами хэш-функций [6]. Со стороны мобильной либо настольной операционной системы для организации байтовой или пакетной передачи данных через порты ввода вывода при программировании применен шаблон объектно-ориентированного проектирования – «Декоратор».

Для разработки устройства регистрации длительных вибрационных сигналов использован проект Arduino [7]. Arduino (Ардуино) – это открытая аппаратно-программная платформа для создания простых электронных устройств и роботов. Она состоит из небольших управляющих плат (с микроконтроллером) и удобной программы Arduino IDE (Integrated Development Environment, интегрированная среда разработки) для написания кода. Это рабочая программа для программиста, которая объединяет все необходимые инструменты для создания программных средств в одном удобном интерфейсе.

Так как 8-битный микроконтроллер не обеспечивает требуемое быстродействие вычислений, полученные от АЦП двухбайтные коды напрямую сохраняются в файл, и в последующем производится дальнейшая обработка на мобильной или настольной платформе. При этом, при частоте дискретизации 25кГц для записи 10 минут вибросигнала требуется 30 мегабайт.

Для последующей обработки накопленные данные передаются на мобильное устройство. В современных многофункциональных операционных системах, как правило, имеется три способа обмена данными: USB- интерфейс, Bluetooth, Wi-Fi. При использовании USB целесообразно использовать CDC (Communications Device Class) интерфейс, при этом создается виртуальный последовательный порт, работающий со скоростью USB интерфейса.

При работе с Bluetooth использованы радио-модули HC-06 [8], при этом реализован беспроводный ноль-модем, скорость которого достигаем 1,3 Мегабит. Радио-модули HC-06 – это популярные и недорогие Bluetooth-адаптеры, применяемые для беспроводной связи между микроконтроллерами (например, платами Arduino) и другими устройствами, такими как смартфоны, планшеты или компьютеры.

В случае Wi-Fi применяется WizFi250-EVB [9], при этом обеспечивается доступ к ресурсам современных TCP/IP сетей. WizFi250-EVB – это отладочная плата (Evaluation Board) от южнокорейской компании WIZnet, созданная для быстрого тестирования и разработки проектов на базе беспроводного Wi-Fi модуля WizFi250. Основной компонент WizFi250 компактный Wi-Fi модуль (стандарт 802.11 b/g/n), предназначенный для интеграции устройств в интернет вещей (IoT), домашнюю автоматизацию или промышленные системы.

Структуру программного средства условно можно разбить на несколько подсистем: 1. Подсистема АЦП; 2. Файловая система; 3. Подсистема времени; 4. Подсистема передачи данных; 5. Подсистема управления.

Подсистема АЦП определяет все параметры цифрового вибросигнала, которые хранятся в энергонезависимой памяти и сохраняются в ini-файл в момент запуска накопления результатов преобразования. Запуск АЦП осуществляется таймером, период которого определяется частотой дискретизации. При завершении АЦП преобразования срабатывает соответствующее прерывание и двухбайтное оцифрованное значение сохраняется в буфере. Очень важно оценить необходимый размер буфера в процессе тестирования, который зависит от параметров и скорости работы

файловой системы. Необходимо, чтобы по времени размера буфера было достаточно на открытие, закрытие файлов и сохранение данных.

Файловая система записывает и читает данные порциями кратными 512 байт. При форматировании диска надо учесть, что чем больше размер кластера, тем меньше изменений приходится вносить в таблицу FAT при записи больших файлов. Запись на SD Card одного только байта физически занимает до двух миллисекунд, поэтому скорость записи можно увеличить только путем параллельной записи массива байт размером кратным 512 байт. Буфер АЦП организован в виде нескольких последовательных массивов, размер каждого равен или кратен 512 байт, чем больше размер буфера, тем выше скорость записи. Это необходимо учитывать в первую очередь при выборе микроконтроллера.

Подсистема времени необходима для организации расписания запуска/останова процесса накопления вибросигнала и для файловой системы при создании файла. Время удобно хранить в виде числа, которое является счетчиком секунд от 1 января 1970 года. Для инициализации счетчика времени используется календарь в виде отдельной микросхемы с отдельным аккумулятором и кварцевым резонатором 32768 кГц.

Подсистема передачи данных позволяет обмениваться потоками данных с настольными или мобильными платформами и использовать любые интерфейсы передачи данных, такие как последовательный порт, USB, Bluetooth, Ethernet, Wi-Fi. Разработаны отдельные аппаратно-зависимые драйверные функции для побайтного чтения/записи соответствующего интерфейса, с тем учетом, что при адаптации к новой аппаратуре перерабатывать придется только их. Аппаратно-независимая часть подсистемы обрабатывает поток данных, кодирует/декодирует пакеты данных, проверяя их целостность (номера пакетов, контрольные суммы). Первый байт каждого пакета представляет собой директиву, определяющую структуру всего пакета. Директивы и соответствующие им пакеты делятся на две группы: пакеты, организованные по принципу запрос/ответ, и одиночные пакеты, формируемые и передаваемые по инициативе самого устройства. По умолчанию, разработаны директивы чтения/записи памяти и специальный пакет (текстовое сообщение) для трассировки работы устройства, облегчающий понимание процессов внутри системы. В зависимости от конфигурации к ним могут быть добавлены директивы чтения/установки времени, директивы для работы с файловой системой (чтение содержимого директории, удаление, переименование и копирование файлов) и транспортная директива передачи любых типизированных данных.

Подсистема управления ожидает пакет с типизированными данными, посредством которых настраиваются все параметры устройства, сохранение в энергонезависимую память и пуск/останов процесса накопления вибросигнала. Запуск/останов осуществляется путем нажатия кнопки на устройстве, или при достижении определенного значения времени, если предварительно настроено

расписание, при этом устройство посылает соответствующий одиночный пакет-уведомление через подсистему передачи данных.

Таким образом, разработаны принципы создания приборов, способных накапливать и обрабатывать длительные непрерывные вибрационные сигналы при обнаружении редких кратковременных изменений структуры вибрационного сигнала и выявления причинно-следственных связей между их появлением и развитием дефектов контролируемых объектов. Рассмотрены структура, функциональные возможности, программное обеспечение регистратора вибрационных сигналов на встроенном процессоре

Список литературы

1. Бранцевич, П. Ю. Цифровая обработка вибрационных сигналов / П. Ю. Бранцевич. – Минск: Бестпринт, 2022. – 297 с.
2. Бранцевич, П. Ю. Компьютерные системы и комплексы обработки вибрационных сигналов / П. Ю. Бранцевич. – Минск: Бестпринт, 2023. – 282 с.
3. Atmel AVR 8-bit and 32-bit Microcontrollers [Электронный ресурс] – Электронные данные. – Режим доступа: <https://static.chipdip.ru/lib/934/DOC011934714.pdf>
4. SD Card Shield V4 [Электронный ресурс] - Электронные данные. - Режим доступа: <https://static.chipdip.ru/lib/126/DOC001126515.pdf>
5. FatFs - Generic FAT File System Module [Электронный ресурс] – Электронные данные. – Режим доступа: <https://elm-chan.org/fsw/ff>
6. Циклический избыточный код [Электронный ресурс] – Электронные данные. – Режим доступа: https://ru.wikipedia.org/wiki/Циклический_избыточный_код
7. Arduino MEGA 2560 [Электронный ресурс] – Электронные данные. – Режим доступа: <https://static.chipdip.ru/lib/042/DOC001042722.pdf>
8. HC-05/06 Bluetooth Modules [Электронный ресурс] - Электронные данные. - Режим доступа: <https://dcc-ex.com/reference/hardware/bluetooth/hc-05-06.html>
9. WizFi250 [Электронный ресурс] - Электронные данные. - Режим доступа: https://github.com/Wiznet/WizFi250_arduino_library

Контакты:

e.bazylev@bsuir.by (Базылев Евгений Николаевич);
branc@bsuir.edu.by (Бранцевич Петр Юльянович).