УДК 004.031.6

ОБЕСПЕЧЕНИЕ ЦЕЛОСТНОСТИ ФАЙЛОВОЙ СИСТЕМЫ ВО ВСТРАИВАЕМЫХ СИСТЕМАХ ПОСРЕДСТВОМ КОНТРОЛЯ CRC

магистрант гр. 115401 Ващилов А. Д.
магистрант гр. 115401 Туровец Н. О.
Научный руководитель - канд. техн. наук Ролич О. Ч.
Белорусский государственный университет
информатики и радиоэлектроники
Минск, Беларусь

Методы обнаружения ошибок предназначены для выявления повреждений сообщений при их передаче или хранении. Для этого устройство, отвечающее за передачу или сохранение сообщения, вычисляет некоторое число, называемое контрольной суммой и являющееся функцией сообщения, и добавляет его к этому сообщению. Устройство, которое принимает или считывает сообщение, используя тот же самый алгоритм, рассчитывает контрольную сумму принятого сообщения и сравнивает её с первоначальным значением. Как правило, контрольная сумма посылается (считывается) в конце сообщения [1]. На рисунке 1 представлен пример сообщения, с вычисленной для него контрольной суммы.

Рисунок 1. Блок информации, содержащий сообщение и контрольную сумму

Так, например, в приборах инерциальной навигации часть показаний, собранных с MEMS-датчиков, необходимо сохранять во внутреннюю или внешнюю память. Очевидно, что при работе прибора в зашумленной среде, сохраненные показания могут быть подвержены коллизии, что при последующем их считывании, фактические значения отличались бы от настоящих [2]. Следовательно, необходимо отделять действительные значение от недействительных.

Применение кодов циклического резервирования (CRC) для обнаружения ошибок во встраиваемых системах предполагает поиск компромисса между

скоростью выполнения алгоритма, потреблением памяти и эффективностью обнаружения ошибок.

Поскольку многие встраиваемые системы имеют значительные ограничения в ресурсах, необходимо понимать доступные варианты компромисса и, по возможности, находить способы достижения наилучшего качества обнаружения ошибок при меньших вычислительных затратах.

Качественным критерием оценки контрольной суммы, как правило, понимают вероятность возникновения коллизии. Причём, чем длиннее сообщение, тем больше вероятность её появления. Для встраиваемых систем наиболее вызывает интерес расстояние Хэмминга или метрика Минковского – минимально возможное число бит сообщения, инверсия которых может привести к коллизии [3].

В основе вычисления CRC лежит понятие полинома. В общем случае, любой блок информации всевозможной длины в памяти прибора можно считать полиномом.

Для вычисления контрольной суммы необходим еще один полином, называемый порождающим полином.

Порождающий полином – это предварительно специальным образом подобранный полином, на который впоследствии будет делиться информационный полином для вычисления контрольной суммы. Оттого, порождающий полином какой степени используется, будет зависеть эффективность обнаружения ошибок [4].

Выбор порождающего полинома – нетривиальная задача. Его выбор должен основываться не только на размере контрольной суммы, но и на размере сообщения. В работе [5, с.6] приведена таблица оптимальных порождающих полиномов для некоторых значений расстояний Хэмминга с указанием размера самого полинома и максимально возможной длины кодируемой информации.

Полиномы длиной 8 бит нашли свое широкое применение во встраиваемых системах. Алгоритм CRC-8 хоть и является одним из наиболее распространенных, однако не показывает такого быстродействия как другие восьмибитные алгоритмы [5, с.4]. Так, алгоритм CRC-8-CCITT показывает высокую производительность благодаря заранее известной таблице значений.

При подсчете контрольной суммы данных большего размера необходимо использовать алгоритм CRC разрядностью 16. Ярким представителем такого алгоритма является CRC-16-CCITT. Данный алгоритм отличается высокой

степенью производительности [6], а также заслужил популярность среди решений для встраиваемых систем [7].

Критически важные с точки зрения безопасности встраиваемые системы требуют большие значения расстояние Хэмминга, которые может обеспечить алгоритм нахождения контрольной суммы. Так, например, в последовательной шине бортовой сети железнодорожного состава кадр данных использует алгоритм CRC-32 с длиной Хэмминга равной шести для обеспечения информационной безопасности в критически важных сообщениях [4, с.1].

В качестве примера вычисления контрольной суммы данных на рисунках 2 и 3 предлагаются блок-схемы алгоритмов записи и считывания значений, полученных с MEMS-акселерометра в файловую систему внутренней памяти микроконтроллера.

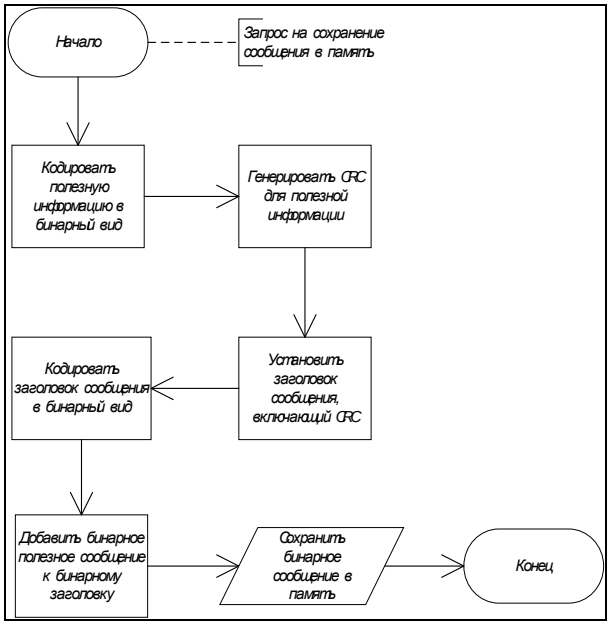

Рисунок 2. Блок-схема алгоритма кодирования сообщения в файловую систему

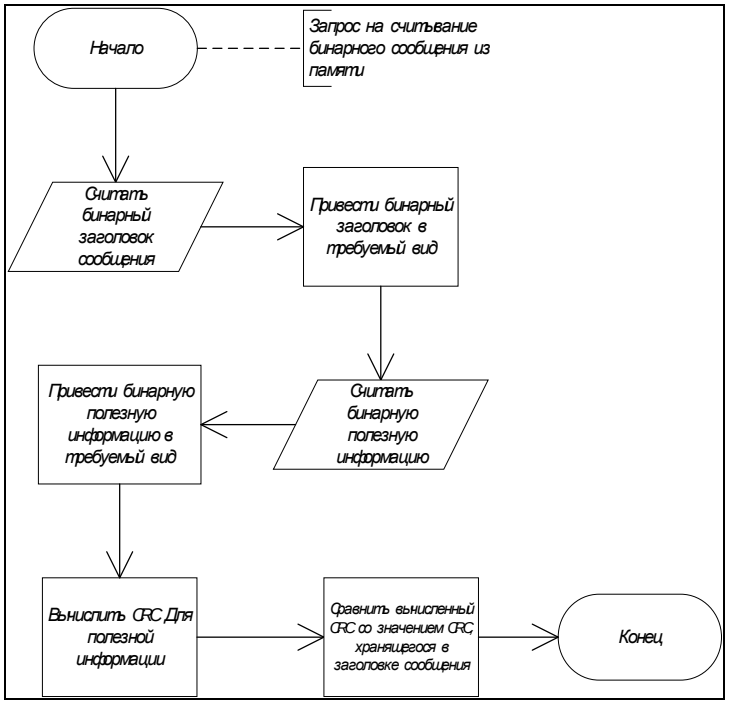

Рисунок 3. Блок-схема алгоритма декодирования сообщения из файловой системы

В результате выполнен анализ алгоритмов вычисления контрольных сумм. Приведены критерии, на которые следует обращать внимание при выборе того или иного алгоритма вычисления контрольной суммы. Представлены алгоритмы вычисления контрольных сумм, наиболее подходящих для встраиваемых систем. Предложены алгоритмы записи, считывания и обработки данных, содержащие контрольную сумму.

Литература

1. Мыцко, Е.А. Исследование программных реализаций табличного и матричного алгоритмов вычисления контрольной суммы CRC32 / Е.А. Мыцко, А. Н.Мальчуков // Вестник науки Сибири. 2011. №1(1). – Томск : ТПУ, 2011. – С. 273-278.

2. Кузьменко, С.В. Разновидности инерциальной навигации и её дальнейшее развитие / С.В. Кузьменко // Материалы X Международной студенческой научной конференции «Студенческий научный форум-2018». – Москва : МГУ, 2018. – 4 с.

3. Журнал Эмбеддед-Инженера // Криптография [Электронный ресурс]. – Режим доступа : http://idoka.ru/cryptography/.

4. Гродненский государственный университет им. Янки Купалы // Практическая работа «Вычисление циклического контрольного кода» по курсу «Системное программное обеспечение» [Электронный ресурс]. – Режим доступа: http://mf.grsu.by/UchProc/livak/po/Labs/teor_lab_0.htm.

5. Koopman, P. Cyclic Redundancy Code (CRC) Polynomial Selection For Embedded Networks / P. Koopman, T. Chakravarty // IEEE Access. – 2004. – [Electronic resource]. – Mode of access: https://users.ece.cmu.edu/~koopman/roses/dsn04/koopman04_crc_poly_embedded.pdf.

6. StackOverflow // Understanting an efficient CRC-CCITT-16 implementation [Electronic resource]. – Mode of access: https://stackoverflow.com/questions/58795372/understanting-an-efficient-crc-ccitt-16-implementation.

7. StackOverflow // Function to Calculate a CRC16 Checksum [Electronic resource]. – Mode of access: https://stackoverflow.com/a/23726131.