Мыцко Евгений Алексеевич, студент кафедры вычислительной техники Института кибернетики ТПУ.
E-mail: EvgenRus70@mail.ru
Область научных интересов: программная и аппаратная реализации алгоритмов вычисления контрольной суммы, тестирование и сравнение по быстродействию алгоритмов вычисления CRC.

Мальчуков Андрей Николаевич, канд. техн. наук, доцент кафедры вычислительной техники Института кибернетики ТПУ.
E-mail: jgs@tpu.ru
Область научных интересов: помехоустойчивое кодирование, полиномиальные коды, системы проектирования помехоустойчивых полиномиальных кодов, алгоритмы поиска образующих полиномов, быстродействующие алгоритмы кодирования и декодирования данных полиномиальными кодами.

УДК 681.3.06

ОСОБЕННОСТИ АППАРАТНОЙ РЕАЛИЗАЦИИ АЛГОРИТМОВ ВЫЧИСЛЕНИЯ КОНТРОЛЬНОЙ СУММЫ CRC32

Е.А. Мыцко, А.Н. Мальчуков

Томский политехнический университет
E-mail: EvgenRus70@mail.ru, jgs@tpu.ru

Приведено описание аппаратных реализаций матричного и табличного алгоритмов вычисления контрольной суммы CRC32 на ПЛИС Cyclone фирмы Altera макета SDK-6.1. Показаны особенности аппаратной реализации на примере описания блоков вычисления CRC32 и работоспособность спроектированных устройств на конкретных примерах.

Ключевые слова:
Контрольная сумма, табличный алгоритм, матричный алгоритм, CRC32, аппаратная реализация, структурная схема, функциональная схема.

В работе [1] рассмотрены некоторые особенности программной реализации алгоритмов вычисления CRC32, которые заключались в требуемом объёме памяти и скорости вычисления CRC32. В отличие от программной реализации, для изучения особенностей аппаратной реализаций необходимо спроектировать функциональную схему устройства вычисления контрольной суммы CRC32 с применением языка описания аппаратуры, а также выбрать программируемую логическую интегральную схему (ПЛИС), в которую будет загружена конфигурация.

Аппаратная реализация алгоритмов

На кафедре вычислительной техники Томского политехнического университета активно используются учебно-лабораторные стенды SDK-6.1 при изучении дисциплины «Схемотехника ЭВМ». В связи с этим, для аппаратной реализации алгоритмов вычисления контрольной суммы CRC32 выбран макет SDK 6.1 [2].

Ввод данных в макет для расчёта контрольной суммы CRC32 производится с персонального компьютера (ПК). Передача данных с ПК на макет осуществляется по последовательному интерфейсу RS-232, используя терминал для передачи данных (term_1b). CRC32 рассчитывается для блока данных и записывается в регистр. Итоговая рассчитанная контрольная сумма отображается на жидкокристаллической индикации (ЖКИ) макета SDK 6.1 при переводе движкового переключателя № 0 (крайний слева) в верхнее положение. На структурной схеме устройства (рис. 1) 4 основных блока. В блоке приёма данных осуществляется приём последовательности битов данных от ПК по интерфейсу RS-232, определение старт и стоп-битов, а также контроль единичного уровня сигнала при передаче. Блок расчёта контрольной суммы вычисляет контрольную сумму CRC32 для данных, поступивших с блока приёма, и записывает её в регистр. Блок вывода на ЖКИ осуществляет перевод значения контрольной суммы в шестнадцатеричную форму и выводит её на дисплей SDK 6.1. При этом в течение расчёта CRC32 блок генерации управляющих импульсов осуществляет синхронизацию всех блоков схемы.

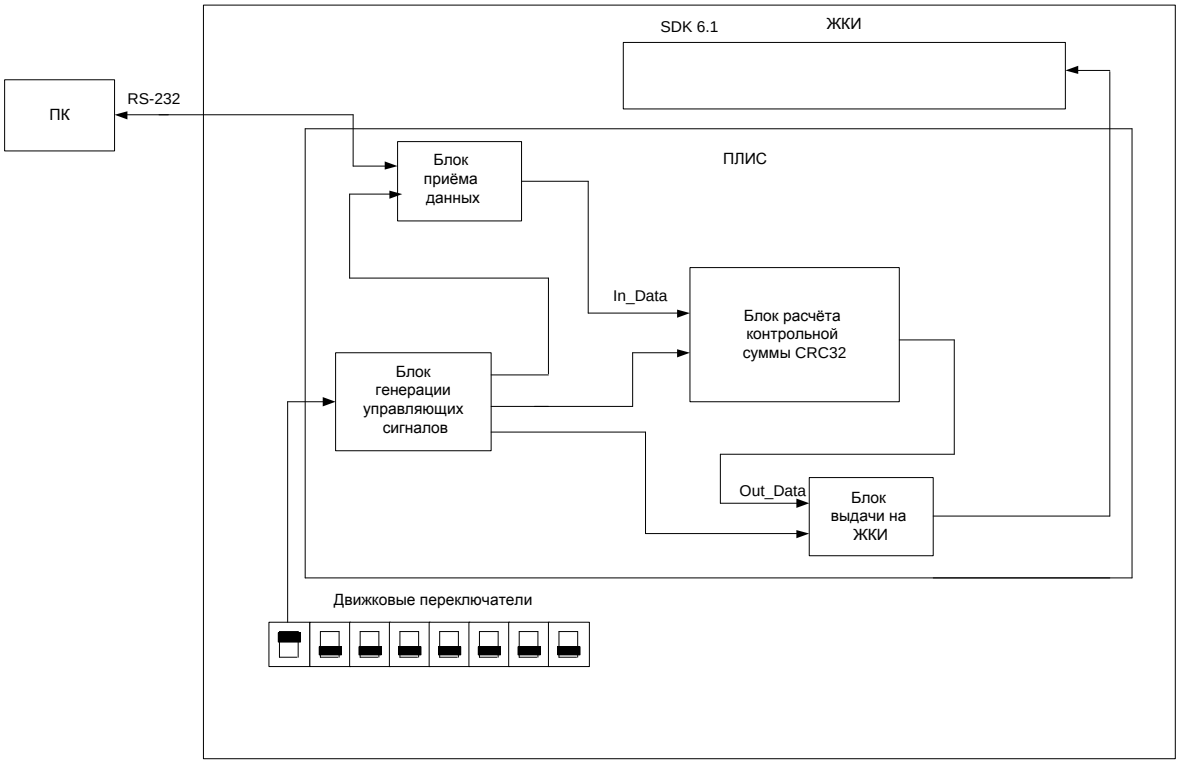

Рис. 1. Структурная схема устройства расчёта контрольной суммы CRC32

Табличный алгоритм

На основе структурной схемы, используя блочно-ориентированный подход [3], в среде QuartusII [4] спроектирована функциональная схема устройства расчёта контрольной суммы CRC32 с использованием языка описания аппаратуры VHDL [5].

Основным блоком в функциональной схеме является блок расчёта CRC32 «CRC32_vhdl» (рис. 2).

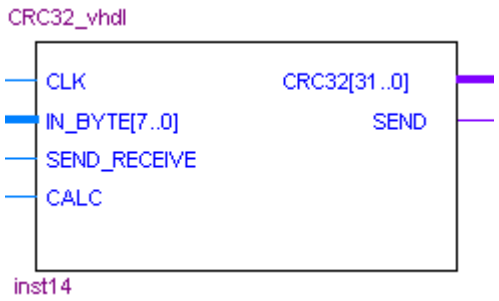

Рис. 2. Блок расчёта CRC32 табличным алгоритмом

Блок расчёта CRC32 (рис. 2) содержит 4 входа и 2 выхода. Вход CLK служит для подачи на блок управляющего синхроимпульса частотой 40 МГц. IN_BYTE [7..0] является входным байтом данных для расчёта контрольной суммы. Управляющий сигнал SEND_RECEIVE устанавливается с помощью движкового переключателя, что позволяет задать режим расчёта CRC или выдачи результата на ЖКИ. При нулевом уровне сигнала SEND_RECEIVE (нижнем положении переключателя) осуществляется приём данных по байту с последующим расчётом контрольной суммы и записью её в регистр. При единичном уровне сигнала SEND_RECEIVE (верхнем положении переключателя) рассчитанная контрольная сумма поступает на выход CRC32 [31..0] для выдачи её на ЖКИ SDK 6.1. Вход CALC служит для получения сигнала расчёта CRC от блока «main_vhdl» для принятого байта. Выходной сигнал SEND служит для

управления блоком преобразования контрольной суммы из символьного представления в шестнадцатеричное.

Матричный алгоритм

В работе [1] описаны различные варианты матричного алгоритма, такие как однобайтовый, двухбайтовый и четырёхбайтовый.

Функциональная схема и блок расчёта CRC32 для однобайтового матричного алгоритма выглядит аналогично, как и для табличного алгоритма. Отличия заключаются только в описании работы блока вычисления CRC32 на языке VHLD [5].

Матричный двухбайтовый алгоритм является модификацией однобайтового матричного алгоритма. Для расчёта CRC32 необходимо формировать данные по 2 байта и передавать их на блок расчёта контрольной суммы. Соответственно, блок расчёта CRC32 при аппаратной реализации будет отличаться визуально (рис. 3) и по функциональному описанию.

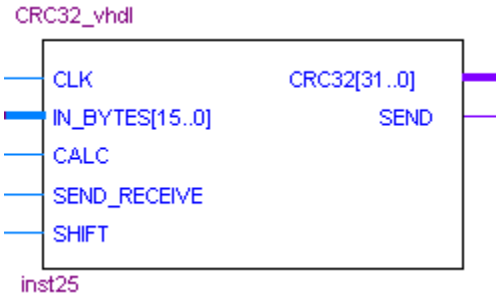

Рис. 3. Блок расчёта CRC32 матричным двухбайтовым алгоритмом

Данный блок имеет шестнадцатиразрядный информационный вход IN_BYTES[15..0] и в отличие от приведённого выше блока для табличного и матричного однобайтового алгоритма имеет дополнительный вход SHIFT, который служит для задания режима однобайтового (при единичном уровне сигнала) или двухбайтового (при нулевом уровне) расчёта CRC32. Это связано с тем, что объём данных в байтах не всегда кратен 2, поэтому для оставшихся байтов нужно применять однобайтовую схему расчёта.

В программной реализации особенность матричного четырёхбайтового алгоритма заключалась в том, что сдвиг данных осуществлялся блоками по 4 байта. Поэтому функциональная схема для аппаратной реализации будет отличаться от описанных ранее алгоритмов. Изменения заключаются в том, что из последовательной передачи данных по порту RS-232 нужно формировать блоки данных по 4 байта, для которых будет рассчитываться контрольная сумма. Таким образом, была спроектирована функциональная схема, реализующая расчёт CRC32 данным алгоритмом.

Блок расчёта CRC32 четырёхбайтовым матричным алгоритмом (рис. 4) содержит 5 входов и 2 выхода. Вход CLK, как и ранее, служит для подачи на блок управляющего синхроимпульса частотой 40 МГц. IN_BYTES [31..0] является информационным тридцатидвухразрядным входом для расчёта CRC32. Двухразрядный вход SHIFT[1..0] выполняет функцию выбора способа расчёта CRC32 в зависимости от кратности набора данных. Пока на информационный вход поступают блоки по 4 байта, данный входной сигнал имеет значение «00». Если же на информационный вход поступает неполный блок данных 1, 2 или 3 байта, то на входе формируется соответствующий двухразрядный сигнал расчёта по однобайтовой («01»), двухбайтовой («10») или трёхбайтовой («11») схеме, что необходимо для расчёта CRC32 при наборах данных не кратных четырём.

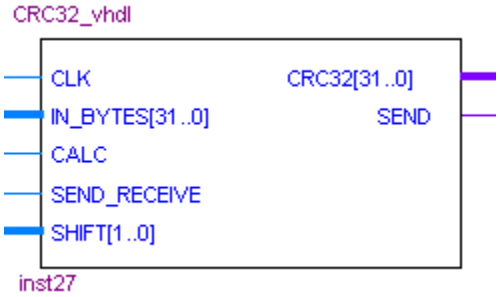

Рис. 4. Блок расчёта CRC32 четырёхбайтовым матричным алгоритмом

Примеры расчёта контрольной суммы CRC32

Далее приведены некоторые примеры расчёта CRC32 с применением SDK 6.1. Ввод данных осуществляется в специальное окно терминала (рис. 5, *а*), а результат отображается на ЖКИ макета (рис. 5, *б*).

а

б

Рис. 5. а) Окно ввода данных терминала term_1b с ПК на макет; б) макет SDK-6.1 с контрольной суммой CRC32 для сообщения «qwerasdfzxcvbnm,» на дисплее

Также терминал позволяет передавать файлы, для которых можно рассчитать CRC32 (рис. 6–8).

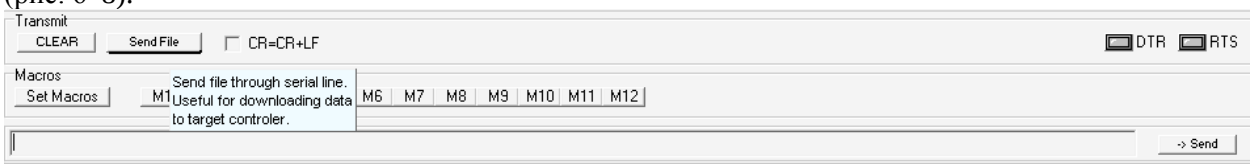

Рис. 6. Использование функции «Send file» в терминале

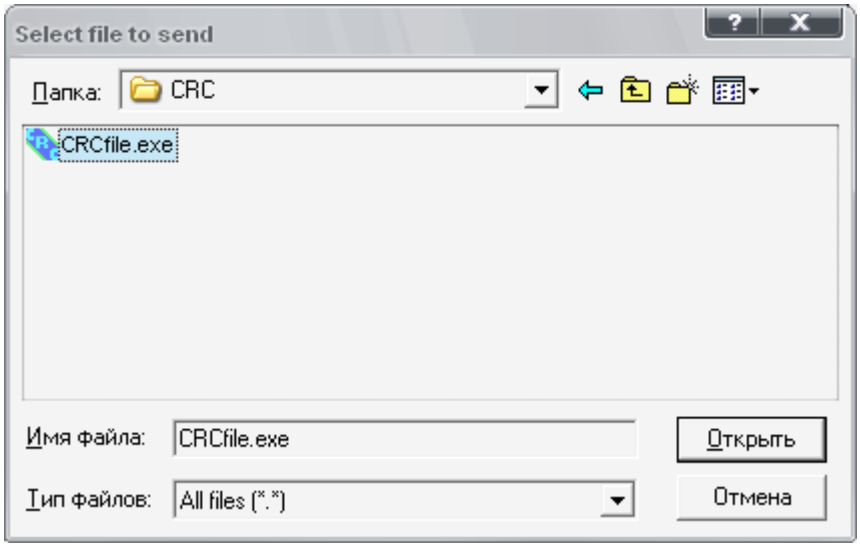

Рис. 7. Открытие файла для передачи через RS-232 на SDK 6.1

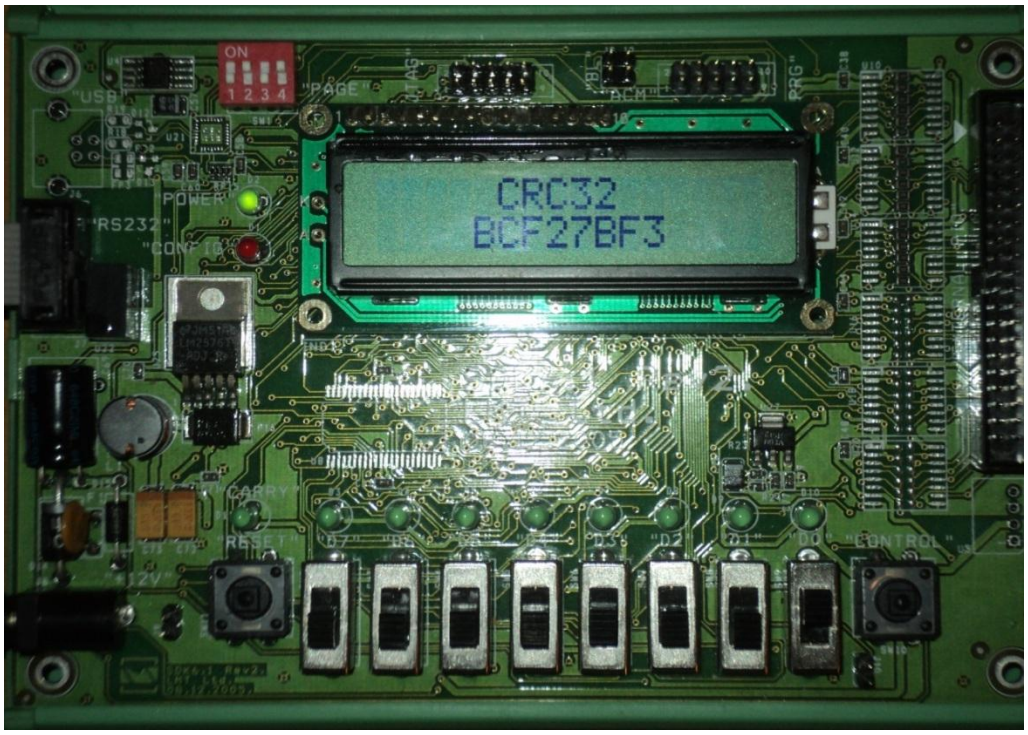

Рис. 8. CRC32 для файла CRCfile.exe на SDK 6.1

На примере архиватора WinRaR, в котором используется алгоритм CRC32, можно удостовериться в правильности расчёта контрольной суммы (рис. 9).

Имя	Размер	Сжат	Тип	Изменён	CRC32
..			Папка		
CRCfile.exe	151 552	60 772	Приложение	15.01.2000 12:20	BCF27BF3

Рис. 9. CRC32 для файла CRCfile.exe в программе WinRaR

Заключение

На основе алгоритмов расчёта контрольной суммы CRC32, используемых в программных реализациях [1], были спроектированы функциональные схемы устройств для аппаратной

реализации алгоритмов на ПЛИС Cyclone макета SDK-6.1. При проектировании функциональной схемы использовался блочно-ориентированный подход (BBD) с описанием блоков на языке VHDL. Основные особенности аппаратной реализации данных алгоритмов заключаются в том, что для табличного и однобайтового матричного алгоритма блоки вычисления CRC идентичны. Для двухбайтового и четырёхбайтового алгоритмов блок вычисления контрольной суммы содержит дополнительный вход, задающий режим расчёта, а разрядность информационного входа соответственно увеличивается. Данные аппаратные средства с использованием полученных конфигураций с реализованными алгоритмами расчёта CRC позволяют осуществлять контроль целостности данных при передаче по последовательному порту.

СПИСОК ЛИТЕРАТУРЫ

1. Мыцко Е.А., Мальчуков А.Н. Исследование программных реализаций табличного и матричного алгоритмов вычисления контрольной суммы CRC32 // Вестник науки Сибири. Серия Информационные технологии и системы управления. – 2011. – № 1 (1). – С. 273–278. URL: http://sjs.tpu.ru/journal/issue/view/2/ showToc/sect/4 (дата обращения: 06.08.2012).
2. Учебный лабораторный стенд SDK-6.1 // Embedded systems. 2005. URL: http://embedded.ifmo.ru/index.php/support/sdk-61 (дата обращения: 06.08.2012).
3. Еремин В.В., Мальчуков А.Н. О применении блочно-ориентированного подхода к разработке устройств на ПЛИС // Вестник науки Сибири. Серия Информационные технологии и системы управления. – 2011. – № 1 (1). – С. 379–381. URL: http://sjs.tpu.ru/journal/issue/view/2/showToc/sect/4 (дата обращения: 06.08.2012).
4. Система проектирования Quartus II // ГАММА. 2007. URL: http://www.icgamma.ru/linecard/altera/kits/quartus2/ (дата обращения: 06.08.2012).
5. Бибило П.Н. Основы языка VHDL. 3-е изд., доп. – М.: Изд-во ЛКИ, 2007. – 328 с.

Поступила 10.09.2012 г.