УДК 004.624:004.021

Ускорение вычисления CRC в устройствах на базе программируемых логических интегральных схем

А. А. Блюдов, Е. А. Волков, Ю. В. Иванов, Г. Ю. Пронин

Петербургский государственный университет путей сообщения Императора Александра I, Российская Федерация, 190031, Санкт-Петербург, Московский пр., 9

Для цитирования: *Блюдов А. А., Волков Е. А., Иванов Ю. В., Пронин Г. Ю.* Ускорение вычисления CRC в устройствах на базе программируемых логических интегральных схем // Известия Петербургского университета путей сообщения. — СПб.: ПГУПС, 2025. — Т. 22. — Вып. 1. — С. 179–185. DOI: 10.20295/1815-588X-2025-1-179-185

Аннотация

Цель: Разработать метод, позволяющий увеличить скорость вычисления CRC для программируемых логических интегральных схем (ПЛИС), превосходящий по скорости метод непосредственного расчета путем циклического сдвига. **Методы:** Для проведения экспериментальных исследований использовалось компьютерное моделирование. Для теоретических исследований применены метод аналитического обзора, теория помехозащитного кодирования. **Результаты:** Предложен и описан метод расчета CRC для посылок постоянной длины с применением произвольных полиномов. Представлены некоторые результаты сравнения контрольных разрядов кодов с наименьшей избыточностью и циклических избыточных кодов. **Практическая значимость:** Описанный в статье метод позволяет качественно быстрее производить вычисление CRC на базе ПЛИС, нежели ранее используемые. Получен способ, позволяющий ускорить расчет циклического избыточного кода для произвольных полиномов при условии фиксированной длины посылки.

Ключевые слова: Помехозащитное кодирование, циклический избыточный код, ПЛИС, кодовое расстояние, разделимые коды.

Введение

Метод кодирования CRC получил широкое распространение в современной технике. В современных системах СЦБ существует множество каналов передачи информации, в которых используется именно этот метод помехозащитного кодирования. При этом не редка та ситуация, когда все сообщения в канале имеют постоянную длину [1, 2].

Также все активнее внедряются программируемые логически интегральные схемы [3, 4]. Принципиально другая относительно микроконтроллеров архитектура ПЛИС позволяет проектировать устройства, значительно опережающие по быстродействию аналогичные устройства на базе микроконтроллеров для целого ряда задач. Это не влечет необходимость использовать или иные алгоритмы, или иные методы непосредственных вычислений. Однако копирование методов, применяемых на другой элементной базе, ведет к неоптимальному использованию ресурсов, которые в ПЛИС значительно более ограничены, чем у микроконтроллеров. В тех же задачах, где приоритетным является не экономия ресурсов, а быстродействие устройства, копиро-

вание методов другой элементной базы может оказаться далеко не самым быстрым вариантом.

В данной работе исследуются способы получения контрольных разрядов, получаемых после вычисления CRC, а также способы их ускоренного вычисления на базе ПЛИС.

Используемые обозначения

k — число контрольных разрядов;

i_j — информационный разряд;

j — номер информационного разряда (0 — младший информационный разряд).

k_{xy} — контрольный разряд. $k_{xy}=i_x \oplus i_y$;

d — кодовое расстояние.

1. Краткие теоретические обоснования

Ускорение обработки информации может осуществляться различными способами. В данной статье преимущественно рассматривается ПЛИС в качестве кодирующего и декодирующего устройства в канале передачи информации. В частности, возможность вычисления логических операций, реализуемых с помощью комбинационных схем за один такт. Традиционный способ вычисления CRC, предполагающий многократное непосредственное деление на полином, может быть сведен к построению комбинационной схемы. Некоторое затруднение вызывает тот факт, что правила получения контрольных разрядов и, как следствие, комбинационная схема будут отличаться не только для каждого полинома, но и в том случае, когда используется отдельный полином, кодирует информационные векторы разной длины.

Также важным аспектами являются скорость передачи информации и минимальное кодовое расстояние. Коды с наименьшей избыточностью по определению обладают лучшим соотношением минимального кодового расстояния и скорости. Существенным недостатком для этих кодов является сложность декодирования. Для векторов разной длины при одном и том же полиноме циклические коды формируют контрольные разряды, аналогичные разрядам кодов с наименьшей избыточностью или имеющие такие же свойства, задачи кодирования и декодирования значительно упрощаются.

2. Сравнение контрольных разрядов кодов с наименьшей избыточностью и циклических избыточных кодов

В результате вычисления контрольных разрядов CRC могут быть получены разряды вида:

– нулевой разряд;

– разряд, повторяющий один из информационных;

– контрольный разряд, который представляет собой результат вычисления суммы по модулю 2 нескольких информационных разрядов.

Это очевидным образом следует из работ [5–7].

Для простоты рассмотрим случай $k = 3$. В [8] приведены коды с минимальной избыточностью:

Для сравнения контрольных разрядов, которые получаются в результате вычисления CRC, с приведенными в табл. 1 осуществим перебор полиномов. Верхнюю границу перебора определим для полиномов, соответствующих приведенным в табл. 1 структурам:

а) для k разрядов общее число способов вычисления возможных контрольных разрядов, представляющих собой все возможные суммы информационных разрядов по модулю 2, равно $C_k^1 + C_k^2 + ... + C_k^k$.

Так как $C_k^1 + C_k^2 + ... + C_k^k = 2^k - 1$, а C_k^1 представляет собой количество информационных раз-

Таблица 1. Коды с минимальной избыточностью k =3

Минимальное кодовое расстояние	$d_{\min} = 1$	$d_{\min} = 2$	$d_{\min} = 3$	$d_{\min} = 4$
Структура разрядов	i_2, i_1, i_0	i_2, i_1, i_0, k_{012}	$i_2, i_1, i_0, k_{01}, k_{02}, k_{12}$	$i_2, i_1, i_0, k_{01}, k_{02}, k_{12}, k_{012}$

Таблица 2. Контрольные разряды CRC от $k = 3$

Десятичный эквивалент	Двоичный эквивалент	Полином	Контрольные разряды
$d = 2$			
3	11	$x^1 \oplus x^0$	k_{012}
$d = 3$			
11	1011	$x^3 \oplus x^1 \oplus x^0$	k_{12}, k_{012}, k_{02}
13	1101	$x^3 \oplus x^2 \oplus x^0$	k_{01}, k_{12}, k_{012}
$d = 4$			
23	10111	$x^4 \oplus x^2 \oplus x^1 \oplus x^0$	$k_{12}, k_{01}, k_{012}, k_{02}$
29	11101	$x^4 \oplus x^3 \oplus x^1 \oplus x^2$	$k_{02}, k_{012}, k_{12}, k_{01}$

рядов, то максимальное количество отличных друг от друга контрольных разрядов равно $2^k - k - 1$;

б) если количество контрольных разрядов равно $2^k - k - 1$ и все контрольные разряды являются уникальными, то кодовое расстояние между двумя произвольными кодовыми словами одинаково;

в) в случае если кодовое расстояние между двумя произвольными кодовыми словами одинаково, то для увеличения минимального кодового расстояния на 1 требуется добавить k разрядов [9–10];

г) количество контрольных разрядов на единицу меньше длины полинома.

Определим длину полинома. Для этого рассчитаем количество контрольных разрядов. Исходя из частного случая бинома Ньютона следует, что общее количество разрядов, включая нулевой, составляет 2^k. Исключая нулевой и информационные разряды, получим:

$$2^k - C_k^0 - C_k^1 = 2^k - 1 - k.$$

Поскольку длина полинома на один больше количества контрольных разрядов, то перебор полиномов имеет смысл осуществлять до $2^{(2^k - k)} - 1$.

Из результатов рассмотрим только те контрольные разряды, вычисления которых представляют собой суммы по модулю 2 информационных разрядов.

Сравнивая табл. 1, 2, заметим, что код с контрольными разрядами k_{01}, k_{02}, k_{12} есть только в одной из них. Путем полного перебора показано, что не существует такого полинома, в результате вычисления которого получились только вышеописанные разряды. Это показывает, что в общем случае для произвольных кодовых разрядов может не существовать полинома, с помощью которого можно получить их путем циклического деления. Однако сам факт широкого распространения CRC показывает, что для многих протоколов связи его защитных свойств достаточно.

3. Вычисление CRC на базе ПЛИС

При переходе на элементную базу ПЛИС непосредственный расчет CRC занимает несколько тактов. Использование табличного подхода к расчету CRC, получившего широкое распространение в системах, построенных на базе микроконтроллеров, в зависимости от полинома может занимать достаточно большое количество памяти, количество которой в ПЛИС существенно ограниченно. При этом построение комбинационной схемы является несложной задачей, а результат ее вычислений может быть получен за один такт. Поэтому целесообразно перейти к непосредственному вычислению контрольных разрядов.

Основываясь на свойствах циклических кодов, описанных в статье [5], опишем способ вычисления контрольных разрядов для заданного полинома и k, не пользуясь методом перебора. В качестве примера рассмотрим случай $k = 3$, $p = 11$.

Пример:

Запишем информационный вектор в виде полинома.

$p = x_2 \oplus x_1 \oplus x_0$

Старшая степень полинома p равна 3. Умножим на нее информационный вектор:

$(x_2 \oplus x_1 \oplus x_0)\, x_3 = x_5 \oplus x_4 \oplus x_3$.

Рассчитаем остаток от деления каждого разряда на полином (рис. 1–3).

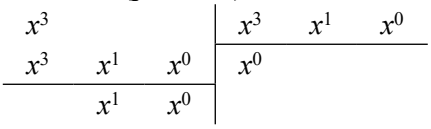

Рис. 1. Расчет остатка от деления x^3 на полином

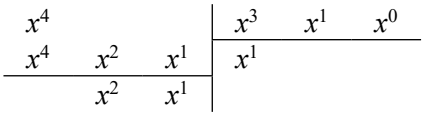

Рис. 2. Расчет остатка от деления x^4 на полином

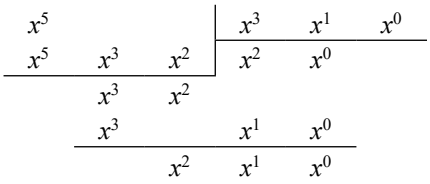

Рис. 3. Расчет остатка от деления x^5 на полином

Таблица 3. Таблица определения контрольных разрядов

Остатки от деления			Делимый полином	Информационный разряд
	x^1	x^0	x^3	x^0
x^2	x^1		x^4	x^1
x^2	x^1	x^0	x^5	x^2
Контрольные разряды				
k_{12}	k_{012}	k_{02}		

Результаты запишем в табл. 3. Количество строк в таблице определения контрольных разрядов (табл. 3) соответствует числу кодируемых информационных разрядов. Количество столбцов равно старшей степени полинома.

Остаток деления информационного разряда, умноженного на старший член полинома, записывается в соответствующую строку.

Каждому столбцу этой таблицы соответствует контрольный разряд. В тех столбцах, где есть x^i, этот информационный разряд будет участвовать в формировании контрольного. Таким образом, в результате циклического деления на полином $p = 11$ вектора из трех информационных разрядов получим контрольные разряды k_{12}, k_{012}, k_{02}. При этом порядок контрольных разрядов в табл. 3 соответствует порядку контрольных разрядов, получаемых при делении с остатком информационного вектора на полином.

Сравнение скоростей вычислительных методов

Вышеописанный метод актуален только для протоколов с блочным помехозащитным кодированием. Так как протоколы многих микропроцессорных систем являются корпоративной тайной, провести количественную оценку применимости разработанной методологии затруднительно. Из того, что можно привести в литературе общего доступа, авторы статьи могут указать только на тот факт, что все помехозащитные коды, применяемые в МПЦ-МПК, являются блочными.

Для сравнения скорости произведения расчетов рассмотрим три метода декодирования:

1. Метод, описанный в статье.
2. «Классический» метод декодирования.
3. Табличный метод декодирования.

Для корректной оценки времени учтем возможность декодирования в процессе приема сообщения. Также следует учесть пакетный характер посылок. Оценку времени будем проводить в тактах и условном времени.

1. Для описанного в статье метода время от получения последнего пакета с информационной частью до расчета контрольной части будет равно времени расчета комбинационной. На базе ПЛИС этот расчет займет один такт.

2. В «классическом» методе понадобится один такт для того, чтобы взять один информационный бит и еще один такт для сложения его по модулю 2 с ранее рассчитанным остатком. Таким образом, расчет займет $2k$ тактов.

3. При использовании табличного метода декодирования возможна различная организация памяти таблицы и поиска по ней. Количество тактов, необходимых для поиска по таблице, обозначим за t. Также необходим еще один такт на вычисление. Время, затраченное на вычисление табличным методом, составит $(t + 1)k / k_Б$ тактов, где $k_Б$ — это длина табличного блока.

В табл. 4 представлены скорости вычисления CRC при декодировании после приема сообщения.

В связи с тем, что предложенный метод служит для увеличения скорости декодирования, необходимо провести оценку скорости в случае декодирования в процессе приема сообщений.

1. Для описанного метода декодирование возможно только после полного приема сообщений. Поэтому время от приема последнего информационного бита до расчета контрольных бит не изменится.

2. Для «классического» метода время расчета сократится до скорости расчета последнего пакета, $2k_П$.

3. Для табличного метода время зависит от длины блока. В случае если она больше или равна длине пакета, то время декодирования составит 3 такта. Если же длина блока меньше длины пакета (что представляется нецелесообразным), то декодирование займет $(t + 1)\ k_П / k_Б$.

В табл. 5 представлены скорости вычисления CRC при декодировании во время приема сообщения.

Это показывает, что на одинаковой аппаратной базе предложенный метод является самым быстрым из решений.

В предложенном методе есть существенное ограничение. При использовании одного и того же полинома разное количество информационных бит требует разных декодирующих автоматов. Это вызывает необходимость строить их индивидуально.

Таблица 4. Скорости вычисления CRC при декодировании после приема сообщения

	Метод приведения к комбинационной схеме	«Классический» метод	Табличный метод
Количество тактов	1	$2k$	$(t + 1)k / k_Б$

Таблица 5. Скорости вычисления CRC при декодировании во время приема сообщения

	Метод приведения к комбинационной схеме	«Классический» метод	Табличный метод
Количество тактов	1	$2k_П$	$(t + 1)$

Заключение

В статье приведено сравнение некоторых кодов с наименьшей избыточностью с циклическими избыточными кодами. Путем перебора было показано, что контрольные разряды, полученные в результате вычисления некоторых полиномов, не всегда совпадают с контрольными разрядами кодов с минимальной избыточностью. Это показывает, что не все контрольные разряды могут быть вычислены как остаток от деления информационного вектора на полином.

Также представлен способ определения контрольных разрядов для произвольных полиномов и информационных векторов, позволяющий построить комбинационные схемы для произвольных полиномов.

Кроме того, в статье описан метод ускорения вычисления CRC на базе программируемых логических интегральных схем путем непосредственного расчета значения контрольных разрядов. Описанный в статье метод является актуальным способом ускорения расчетов для каналов с постоянной длиной сообщений, где функции кодирую-

щего и декодирующего устройства реализованы на базе ПЛИС. Он сводит расчет остатка от деления к простой комбинационной схеме, выходы которой могут быть получены за один такт.

Список источников

1. Прохорова Г. М. Оборудование станции устройствами микропроцессорной централизации ЭЦ-ЕМ с увязкой с системой диагностирования и мониторинга (АДК-СЦБ) / Г. М. Прохорова // Форум молодых ученых. — 2017. — № 6(10).

2. Калинин Т. С. Спектрально-сигнатурная диагностика микропроцессорных информационно-управляющих систем железнодорожной автоматики и телемеханики / Т. С. Калинин // ИВД. — 2012. — № 1.

3. Федухин А. В. ПЛИС-системы как средство повышения отказоустойчивости / А. В. Федухин, А. А. Муха // ММС. — 2010. — № 1. — URL: https://cyberleninka.ru/article/n/plis-sistemy-kak-sredstvo-povysheniya-otkazoustoychivosti.

4. Тарасов И. Проектирование конфигурируемых процессоров на базе ПЛИС / И. Тарасов // Компоненты и Технологии. — 2006. — № 57. — URL: https://cyberleninka.ru/article/n/proektirovanie-konfiguriruemyh-protsessorov-na-baze-plis-1.

5. Berlekamp E. R. A Construction for Partitions Which Avoid Long Arithmetic Progressions / E. R. Berlekamp // Canadian Mathematical Bulletin. — 1968. — Vol. 11. — Iss. 3. — Pp. 409–414. — DOI: 10.4153/CMB-1968-047-7.

6. Hamming R.W. Error detecting and error correcting codes / R.W. Hamming // The Bell System Technical Journal. — 1950. — Vol. 29. — Iss. 2. — DOI: 10.1002/j.1538-7305.1950.tb00463.x.

7. Sridevi N. Implementation of Error Correction Techniques in Memory Applications / N. Sridevi, K. Jamal, K. Mannem // 2021 5th International Conference on Computing Methodologies and Communication. — April 08–10 2021. — DOI: 10.1109/ICCMC51019.2021.9418432.

8. Блюдов А. А. Распределение мощности кодов с наименьшей избыточностью алфавитов в зависимости от количества бит и кодового расстояния / А. А. Блюдов, Д. В. Пивоваров, Г. Ю. Пронин // Известия Петербургского университета путей сообщения. — 2023. — № 2. — URL: https://cyberleninka.ru/article/n/raspredelenie-moschnosti-kodov-s-naimenshey-izbytochnostyu-alfavitov-v-zavisimosti-ot-kolichestva-bit-i-kodovogo-rasstoyaniya.

9. Shahariar Parvez A. H. M. Design and implementation of hamming encoder and decoder over FPGA / A. H. M. Shahariar Parvez et al. // International Conference on Computer Networks and Communication Technologies: ICCNCT 2018. — Springer Singapore, 2019. — Pp. 1005–1022.

10. Panem C. Polynomials in Error Detection and Correction in Data Communication System / C. Panem, V. Gad, R. Gad // Coding Theory. — 2019. — P. 29.

Дата поступления: 01.10.2024
Решение о публикации: 25.11.2024

Контактная информация:
БЛЮДОВ Антон Александрович — канд. техн. наук;
blyudov@pgups.ru
ВОЛКОВ Егор Алексеевич — аспирант;
volkov@crtc.spb.ru
ИВАНОВ Юрий Владимирович — аспирант;
DeusIlluminatus@yandex.ru
ПРОНИН Георгий Юрьевич — аспирант;
georgiy3pronin@gmail.com

Acceleration of Software CRC Calculation Based on Programmable Logic Integrated Circuits

A. A. Blyudov, E. A. Volkov, Yu. V. Ivanov, G. Yu. Pronin

Emperor Alexander I St. Petersburg State Transport University, 9, Moskovsky pr., Saint Petersburg, 190031, Russian Federation

For citation: Blyudov A. A., Volkov E. A., Ivanov Yu. V., Pronin G. Yu. Acceleration of Software CRC Calculation Based on Programmable Logic Integrated Circuits // *Proceedings of Petersburg State Transport University*, 2025, vol. 22, iss. 1, pp. 179–185. (In Russian) DOI: 10.20295/2223-9987-2025-1-179-185

Summary

Purpose: To develop a method for increasing the speed of CRC computation for programmable logic integrated circuits (PLICs) that exceeds the speed of the direct computation method by cyclic shift. **Methods:** Computer simulation has been used for experimental studies. The analytical review method and the theory of noise protection coding have been used for theoretical studies. **Results:** A CRC calculation method for constant length messages using arbitrary polynomials is proposed and described. Control digit codes with minimum redundancy and cyclic redundant codes have been compared and the results are presented. **Practical significance:** The method described allows for a qualitatively faster CRC calculation based on FPGAs than those previously used. A method for accelerating the calculation of cyclic redundancy code for arbitrary polynomials has been developed provided that the message length is fixed.

Keywords: Anti-jamming coding, cyclic redundancy code, FPGA, code distance, separable codes.

References

1. Prokhorova G. M. Oborudovanie stantsii ustroystvami mikroprotsessornoy tsentralizatsii ETs-EM s uvyazkoy s sistemoy diagnostirovaniya i monitoringa (ADK-STsB) [Equipping a station with EC-EM microprocessor centralization devices linked to the diagnostics and monitoring system (ADK-SCB)]. *Forum molodykh uchenykh* [Forum of young scientists]. 2017, Iss. 6(10). (In Russian)

2. Kalinin T. S. Spektral'no-signaturnaya diagnostika mikroprotsessornykh informatsionno-upravlyayushchikh sistem zheleznodorozhnoy avtomatiki i telemekhaniki [Spectral-signature diagnostics of microprocessor information and control systems of railway automation and telemechanics]. *IVD*, 2012, Iss. 1. (In Russian)

3. Fedukhin A. V., Mukha A. A. PLIS-sistemy kak sredstvo povysheniya otkazoustoychivosti [FPGA systems as a means of increasing fault tolerance]. *MMS*, 2010, Iss. 1. Available at: https://cyberleninka.ru/article/n/plis-sistemy-kak-sredstvo-povysheniya-otkazoustoychivosti. (In Russian)

4. Tarasov I. Proektirovanie konfiguriruemykh protsessorov na baze PLIS [Design of configurable processors based on FPGA]. *Komponenty i Tekhnologii* [Components and Technologies]. 2006, Iss. 57. Available at: https://cyberleninka.ru/article/n/proektirovanie-konfiguriruemyh-protsessorov-na-baze-plis-1. (In Russian)

5. Berlekamp E. R. A Construction for Partitions Which Avoid Long Arithmetic Progressions. Canadian Mathematical Bulletin, 1968, vol. 11, Iss. 3, pp. 409–414. DOI: 10.4153/CMB-1968-047-7.

6. Hamming R.W. Error detecting and error correcting codes. The Bell System Technical Journal, 1950, vol. 29, Iss. 2. DOI: 10.1002/j.1538-7305.1950.tb00463.x.

7. Sridevi N., Jamal K., Mannem K. Implementation of Error Correction Techniques in Memory Applications. 2021 5th International Conference on Computing Methodologies and Communication. April 08–10 2021. DOI: 10.1109/ICCMC51019.2021.9418432.

8. Blyudov A. A., Pivovarov D. V., Pronin G. Yu. Raspredelenie moshchnosti kodov s naimen'shey izbytochnost'yu alfavitov v zavisimosti ot kolichestva bit i kodovogo rasstoyaniya [Distribution of the power of codes with the least redundancy of alphabets depending on the number of bits and the code distance]. *Izvestiya Peterburgskogo universiteta putey soobshcheniya* [Bulletin of the Petersburg University of Railway Engineering]. 2023, Iss. 2. Available at: https://cyberleninka.ru/article/n/raspredelenie-moschnosti-kodov-s-naimenshey-izbytochnostyu-alfavitov-v-zavisimosti-ot-kolichestva-bit-i-kodovogo-rasstoyaniya. (In Russian)

9. Shahariar Parvez A. H. M. et al. Design and implementation of hamming encoder and decoder over FPGA. International Conference on Computer Networks and Communication Technologies: ICCNCT 2018. Springer Singapore, 2019, pp. 1005–1022.

10. Panem C., Gad V., Gad R. Polynomials in Error Detection and Correction in Data Communication System. Coding Theory, 2019, p. 29.

Received: October 01, 2024
Accepted: November 25, 2024

Author's information:
Anton A. BLYUDOV — PhD in Engineering;
blyudov@pgups.ru
Egor. A. VOLKOV — Postgraduate Student;
volkov@crtc.spb.ru
Yuriy V. IVANOV — Postgraduate Student;
DeusIlluminatus@yandex.ru
Georgy Yu. PRONIN — Postgraduate Student;
georgiy3pronin@gmail.com