

УДК 621.391.7:004.942:621.3.049.77

АЛГОРИТМЫ ПОМЕХОУСТОЙЧИВОГО КОДИРОВАНИЯ CLC И OPCOSA

Т. А. ТУМАНОВ, Л. ВЕЙ, А. Д. ВАСЮТИЧ, Ц. МА, С. ЖЭНЬ, А. В. КУРИЛОВИЧ

*Белорусский государственный университет информатики и радиоэлектроники, Республика Беларусь**Поступила в редакцию 6 апреля 2026*

Аннотация. Проведено сравнение алгоритмов двумерного помехоустойчивого кодирования и декодирования информации, представленной в виде матрицы 4×4 : Collum-Line Codes (CLC) и Optimized Product Codes for Space Applications (OPCoSA). Алгоритмы реализованы в программе моделирования цифровых логических схем Logisim и среде проектирования Vivado для программируемых логических интегральных схем (ПЛИС) Xilinx. Получены количественные оценки занимаемых кодами ресурсов и потребляемой мощности ПЛИС.

Ключевые слова: коды с коррекцией ошибок, CLC, OPCoSA, моделирование, комбинации многократных ошибок.

Введение

Так как в процессе передачи по каналам связи информация может исказиться, из-за действия внешних факторов, приводящих к одиночным или множественным ошибкам, разработаны и широко применяются методы помехоустойчивого кодирования и декодирования, позволяющие обнаружить и устранить несоответствия в принятом сообщении.

В случае передачи информации в космическом пространстве фактором, влияющим на сигнал, могут быть высокоэнергетические радиационные частицы окружающего пространства. Во многих случаях, действие этого фактора, приводит к нежелательным последствиям, которые уменьшаются при использовании кодеров с коррекцией ошибок [1, 2]. На сегодняшний день кодеры с возможностью корректирования только одиночных ошибок устарели, так как с развитием технологий передачи информации они перестали удовлетворять потребностям. В связи с этим разработаны различные варианты алгоритмов, способные восстановить информацию в случае возникновения многократных ошибок [3]. Представителями этой группы являются кодеры Collum-Line Codes (CLC) и OPCoSA, построенные на базе расширенного кода Хэмминга с проверкой на четность. Они обладают высокой корректирующей способностью при сравнительно небольшом потреблении вычислительных ресурсов приемно-передающих устройств [4–6].

Коррекция ошибок в CLC и OPCoSA

Кодек CLC (16, 40) представляет собой двунаправленную матрицу кодирования, которая формирует контрольные биты по столбцам и биты проверки четности по строкам. Для CLC 40-битное кодовое слово, состоит информационных бит в виде матрицы 4×4 , которая дополняется проверочными разрядами (8, 4) расширенного кода Хэмминга по строкам и для каждого столбца дополняется битом проверки четности [4]. Итоговая структура позволяет корректировать одиночные ошибки с помощью алгоритма Хэмминга и множественные ошибки с помощью битов проверки четности. Кроме того, есть и другие разновидности кодера CLC, такие как CLC (16, 39) и CLC (16, 54) [5]. Отличие CLC (16, 39) от CLC (16, 40) заключается в том, что данные 16 информационных битов записываются в 2 строки по 8 бит, к которым применяется расширенный код Хэмминга и добавляются биты проверки четности в столбец и строку. Для CLC (16, 54) применяется следующий алгоритм: 16 бит информации распределяются

в 8 строк по 2 бита, далее аналогично применяется расширенный код Хэмминга и записываются контрольные биты четности. Декодирование по алгоритму CLC может проходить в стандартном режиме (CLC-S) и расширенном (CLC-E). Расширенный режим, используя двухитерационный процесс корректирования ошибок, позволяет корректировать более сложные комбинации многократных ошибок [5]. Структура кодового слова CLC (16, 40) схематически представлена на рис. 1, где D – информационные биты, C – контрольные биты, P – биты проверки четности.

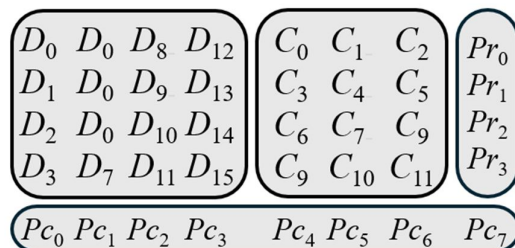


Рис. 1. Структура кодового слова алгоритма CLC (16, 40) [5]

Optimized Product Codes for Space Applications (OPCoSA) является дальнейшим развитием кодека Product Codes for Space Applications (PCoSA), предложенный в источнике [7]. Алгоритм кодирования заключается в применении к 16-битной информационной матрице расширенного кода Хэмминга по строкам и столбцам, тем самым формируются контрольные биты и биты проверки четности для 4-х строк и столбцов [6]. В сравнении с кодеком PCoSA, у которого кодовое слово состоит из 64 бит (16 бит информации и 48 проверочных бит), нововведением кодека OPCoSA является сокращение кодового слова до 48 бит (рис. 2). Такие результаты были достигнуты при помощи удаления области повторного кодирования для уже полученных проверочных бит, тем самым уменьшив на 33 % общее количество проверочных бит. Несмотря на уменьшение количества проверочных бит, корректирующая способность показывает такую же эффективность для большинства комбинаций многократных ошибок [6].

D_0	D_1	D_2	D_3	CI_0	CI_1	CI_2	PI_0
D_4	D_5	D_6	D_7	CI_3	CI_4	CI_5	PI_1
D_8	D_9	D_{10}	D_{11}	CI_6	CI_7	CI_8	PI_2
D_{12}	D_{13}	D_{14}	D_{15}	CI_9	CI_{10}	CI_{11}	PI_3
$C2_0$	$C2_1$	$C2_2$	$C2_3$				
$C2_4$	$C2_5$	$C2_6$	$C2_7$				
$C2_8$	$C2_9$	$C2_{10}$	$C2_{11}$				
$P2_0$	$P2_1$	$P2_2$	$P2_3$				

Рис. 2. Структура кодового слова алгоритма OPCoSA (48, 16) [6]

Моделирование в программе Logisim

На первом этапе сравнительного анализа кодеков помехоустойчивого кодирования CLC и OPCoSA было проведено сравнение их фактической корректирующей способности многократных ошибок. Основываясь на предложенных в источниках [3 – 5] методах кодирования и декодирования, в программе Logisim были реализованы: кодер, модель канала передачи информации и декодер (рис. 3) кодеков CLC (16, 40) и OPCoSA (48, 16).

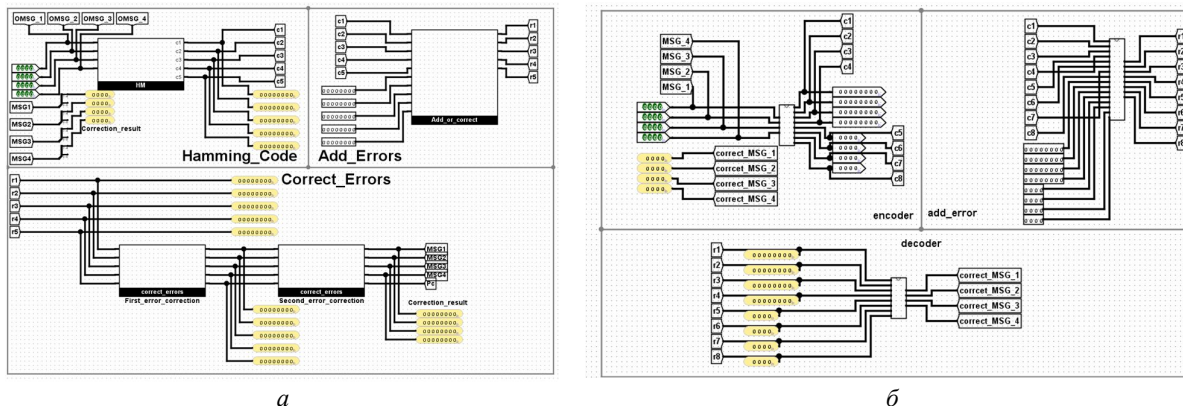


Рис. 3. Структурные схемы кодов помехоустойчивого кодирования в программе Logisim: *a* – CLC; *б* – OPCoSA

Как показано на рис. 3, модули «Hamming_code» и «encoder» являются кодерами. Модуль «add_error» является моделью канала связи, которая позволяет вносить ошибки в кодовое слово. Модули «Correct_Errors» и «decoder» являются декодерами и способны корректировать ошибки в полученном из канала связи кодовом слове. Информация на входе кодера и информация на выходе декодера расположены рядом, для удобства сравнения.

Оценка корректирующей способности кодов производилась путем внесения через модуль «add_errors» случайных комбинаций ошибок кратности от 1 до 8. Для каждого из проектов CLC и OPCoSA отдельно была применена симуляция по методу Монте-Карло. По итогам симуляции было определено количество успешно скорректированных ошибок для каждого алгоритма. Все тесты были проведены в одних и тех же условиях и при одинаковых комбинациях ошибок с целью сравнения фактической корректирующей способности кодов CLC и OPCoSA.

По итогам моделирования построены диаграммы, представленные на рис. 4.

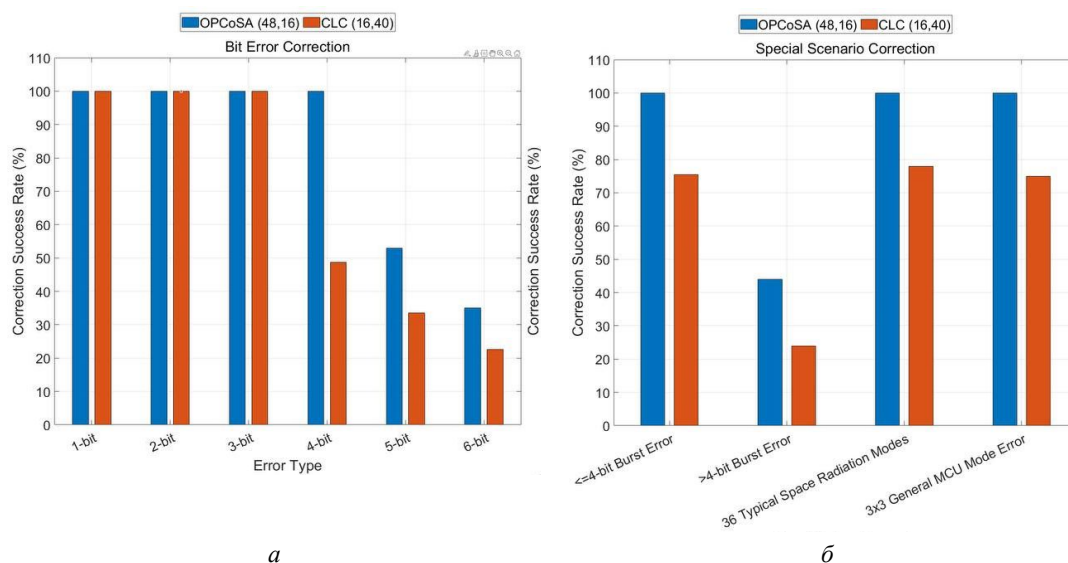


Рис. 4. Сравнение корректирующей способности CLC и OPCoSA: *a* – общий случай; *б* – специальные комбинации ошибок

Результаты экспериментов показывают, что оба алгоритма достигают 100 % корректирующей способности при возникновении от одной до трех ошибок. В случае четырех ошибок корректирующая способность алгоритма OPCoSA достигает 100 %, в то время как у CLC она падает до 48, 69 %. При 5-кратных и 6-кратных ошибках, корректирующая способность для

OPCoSA составляет 53 % и 35 % соответственно, что существенно больше, чем у алгоритма CLC, для которого данные показатели составляют 33,58 % и 22,58 %. При некоторых особых комбинациях ошибок преимущество OPCoSA еще более очевидно: корректирующая способность данного алгоритма составляет 100 % для пакетных ошибок кратности ≤ 4 и 36 типичных шаблонов ошибок в поле 3×3 , вызванных действием космической радиации. В свою очередь, при тех же условиях для CLC корректирующая способность варьируется от 75 % до 78 %. При пакетных ошибках кратности больше 4, OPCoSA также демонстрирует свое преимущество в корректирующей способности: для данного алгоритма возможна коррекция в 44 % случаев, в то время как у CLC данный показатель составляет 24 %. Обнаружено принципиальное различие между двумя алгоритмами, заключающееся в том, что коррекция ошибок CLC работает эффективно только в случае, если они располагаются в области 16-битного информационного поля, в то время как для OPCoSA представляется возможным корректировать ошибки вне зависимости от их положения в 48-битном кодовом слове. Это делает OPCoSA более предпочтительным для условий с частым возникновением многократных ошибок, так как для CLC изменение контрольных бит кодового слова приведет к неправильной коррекции ошибок.

Моделирование в САПР Xilinx Vivado

После симуляции кодеков в программе Logisim и оценки их корректирующих способностей, схема Logisim была переведена на язык описания аппаратуры Verilog, который впоследствии был импортирован в САПР Xilinx Vivado, представляющую собой среду для симуляции и программирования логических элементов ПЛИС фирмы Xilinx. В качестве платы для симуляции была использована Xilinx Vertex-7 серии XC7V485TFFG1157-1, так как она обладает высоким быстродействием и достаточным набором логических ресурсов для реализации алгоритмов CLC и OPCoSA на своей базе.

Дальнейшее моделирование проводилось с использованием инструментов САПР Xilinx Vivado. Для сравнительной оценки алгоритмов CLC и OPCoSA были получены данные об используемых платой ресурсах и потребляемой мощности. Результаты приведены на рис. 5.

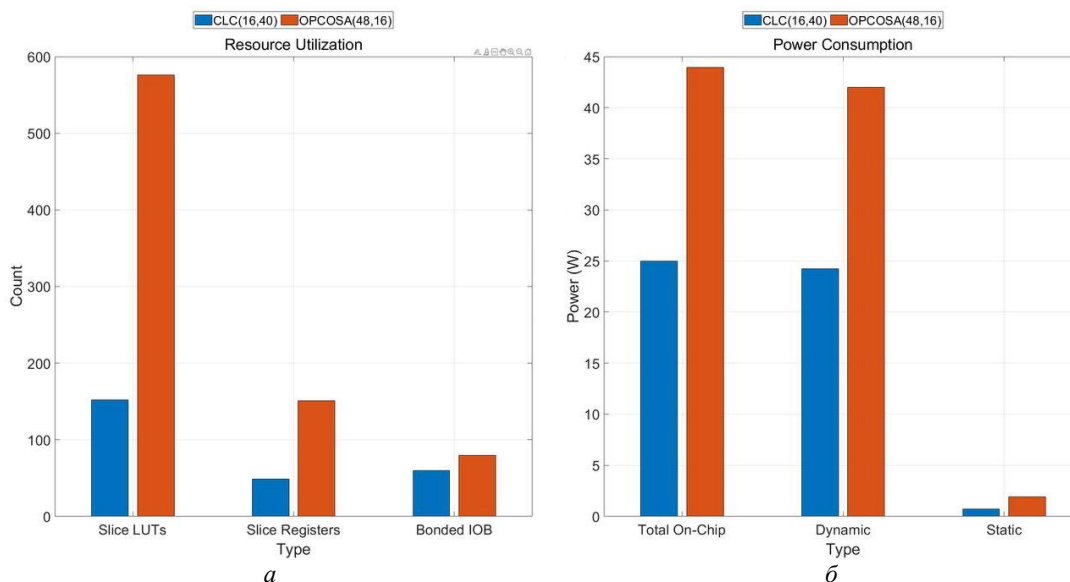


Рис. 5. Ресурсозатраты алгоритмов CLC и OPCoSA на ПЛИС:

а – использование логических ресурсов; *б* – энергопотребление

Результаты моделирования алгоритмов показали, что ресурсозатраты у OPCoSA больше чем у CLC приблизительно в 2 раза. Занятость LUT для алгоритма OPCoSA составила 1454 единиц, а регистров – 475 единиц. Для CLC эти показатели составляют 752 LUT и 251 регистров. Энергопотребление демонстрирует аналогичное различие у обоих алгоритмов. У OPCoSA

данный показатель составляет 7,6 Вт, в то время как энергопотребление алгоритма CLC на симулируемой плате составляет 5,2 Вт.

Заключение

Сравнительный анализ кодеров CLC (16, 40) и OPCoSA (48, 16) показал различие таких параметров как корректирующая способность и потребление логических ресурсов на ПЛИС. CLC подходит для устройств с ограниченными вычислительными ресурсами, которые будут применяться в областях с низкой вероятностью появления многократных ошибок. Несмотря на то, что OPCoSA представляет собой вариант, требующий больших логических ресурсов на ПЛИС, этот недостаток компенсируется достаточно высокой корректирующей способностью данного алгоритма. В условиях с большой вероятностью возникновения многократных ошибок, данный алгоритм позволяет создать гораздо более эффективное устройство передачи данных.

CLC AND OPCOSA: NOISE-RESISTANT CODING ALGORITHMS

T. A. TUMANOV, L. WEI, A. D. VASIUTICH, J. MA, X. REN, A.V. KURILOVICH

Abstract. A comparison was conducted between two-dimensional error-correction coding and decoding algorithms for information presented in a 4×4 matrix: Column-Line Codes (CLC) and Optimized Product Codes for Space Applications (OPCoSA). The algorithms were implemented using the Logisim digital logic simulator and the Vivado design suite for Xilinx Field-Programmable Gate Arrays (FPGAs). Quantitative assessments of the resources utilized by the codecs and the power consumption of the FPGAs were obtained.

Keywords: error-correcting codes, CLC, OPCoSA, modeling, multiple error combination.

Список литературы

1. Hazucha P., Svensson C. // IEEE Transactions on Nuclear Science. 2000. Vol. 47(6). P. 2586-2594.
2. Ibe E., Taniguchi H., Yahagi Y., Shimbo K., Toba T // IEEE Trans. Electron Devices. 2010. Vol. 57(7). P. 1527-1538.
3. Argyrides C., Zarandi H., Pradhan D. // Proc. IEEE Int. Symp. Defect Fault-Tolerance VLSI Syst. (DFT). 2007. P. 340-348.
4. Castro H., Silveira J., Coelho A., Silva F., Magalhaes P., Lima O. // Proc. IEEE Int. New Circuits Syst. Conf. (NEWCAS). 2016. P. 1-4.
5. Silva F., Silveira J., Marcon C., Vargas F., Lima O. Jr. // Journal of Electronic Testing. 2018. Vol. 34(4). P. 417-433.
6. Freitas D., Silveira J., Marcon C., Naviner L., Mota J. // Integration. 2022. Vol. 84. P. 131-141.
7. Freitas D., Mota D., Goerl R., Marcon C., Vargas F., Silveira J., Mota J. // Integration. 2020. Vol. 74. P. 71-80.