

*Г.В. Кормаков, С.И. Гуров*

## **СБОЕУСТОЙЧИВЫЕ ОБРАТИМЫЕ СХЕМЫ И МЕТОД ИХ СИНТЕЗА В ПРОСТРАНСТВЕ ХЕММИНГА\***

### **Введение**

На сегодняшний день стал уже очевиден кризис классических информационных технологий на элементной базе интегральных микросхем (ИМС), которые перестают отвечать возрастающим требованиям по быстродействию, объёмам используемой памяти, плотности элементов на кристалле, надёжности вычислений и др. [1]. Основными недостатками классических методов вычислений является чрезмерное выделение тепла при обеспечении указанных требований и недостаточная устойчивость специализированных ИМС к внешним воздействиям.

Известно, что в выполнении логических операций классическими схемами при потере одного бита информации выделяется энергия

[Дж],

где  $k$  - постоянная Больцмана,  $T$  - температура в градусах Кельвина [2]. В пересчете на процессор суммарная рассеиваемая мощность вырастает уже до величин порядка  $1 \text{ Вт}^1$ . На практике, с учётом активных сопротивлений реальных полупроводников, получаем процессоры, *которые по тепловыделению обгоняют утюги*<sup>2</sup>. Эту ситуацию можно назвать «*тепловым проклятием*» существующей парадигмы ВТ.

В известной статье Р. Меркля и К. Дрекслера «Спиральная логика» отмечалось, что есть только два способа, как обеспечить рассеивание энергии ниже : проводить вычисления при низких температурах, уменьшая , или использовать обратимую логику [3]. При этом низкотемпературная работа не уменьшает общую рассеиваемую энергию, а просто переносит её затраты на холодильное оборудование. Ч. Беннет в 1973 г. показал, как можно построить обратимый компьютер, рассеивающий энергию, значительно меньшую, чем на каждую элементарную операцию, а в идеале и иметь нулевое рассеивание энергии [4]. В связи с этим понятны причины активизации в последнее время интереса к развитию методов синтеза обратимых схем, как комбинационных, так и последовательностных [5-9].

На этом пути, однако, возникает проблема надёжности функционирования ИМС, и, в первую очередь - проблема их устойчивости к кратковременным самоустраняемым отказам или *сбоям* (SEU, single event upsets), второй из указанных выше недостатков. Причинами сбоев явля-

---

\* Исследование выполнено при финансовой поддержке РФФИ, проект № 16-01-00196

<sup>1</sup> <http://old.computerra.ru/2004/538/204845>

<sup>2</sup> <https://habrahabr.ru/post/113332>

ются воздействия на схему различных видов помех: радиационных, скачков напряжений питания, деградаций сигналов во времени и др., и чем выше частота тактовой синхронизации, тем больше вероятность сбоя. Кроме того, современные специализированные вычислительные устройства часто работают в условиях воздействия внешней радиации. Ионизация высокой интенсивности, вызванная как  $\alpha$ - и  $\gamma$ -излучением, так и тяжелыми заряженными частицами, создает импульсы переходного тока, которые могут вызвать переключение битов в схемах функциональной логики, что делает схему временно неисправной. При уменьшении проектных норм до десятков и единиц нанометров, энергия активации, вызывающая ошибки, уменьшается. Поэтому сбоеустойчивость является важнейшим требованием к аппаратуре, работающей, например, в тяжелых условиях космоса. В настоящее время в мире активно развивается направление радиационно-стойкого проектирования (RHBD, Radiation Hardening By Design) основанное на использовании схемотехнических, топологических и алгоритмических методов повышения сбоеустойчивости [10-12].

В данной статье рассмотрены основные существующие модели построения сбоеустойчивых обратимых схем и предложен новый подход к данной проблеме.

### Обратимые комбинационные элементы: входы и выходы

Говорят, что вычисления *логически обратимы*, если по выходным величинам можно восстановить входные. Такие вычисления реализовывают на обратимых элементах или гейтах. Назовём элемент с  $n$  входами обратимым, если и только если он осуществляет некоторую биекцию на множестве всех  $2^n$  возможных входов или их перестановку. Как следствие, такой элемент должен иметь  $2^n$  выходов. В общем случае элементы с  $n$  входами и  $2^n$  выходами называют  $n$ -элементами.

Входы обратимых элементов разделяются на *управляющие* (*контролирующие, адресные*) и *управляемые*. Управляющие входы суть константы, определяющие значения конкретной функции, реализуемой гейтом, а управляемые преобразуются в *сигнальные*.

На выходе обратимого элемента кроме *информационных битов*, представляющих значения вычисляемых функций, как правило, формируются ещё и дополнительные биты, называемые (неудачно) *мусорными* (*garbage*) или *стоками* (*sink*) соответственно.

Мусорные биты в действительности играют очень важную роль: они дают возможность по выходному вектору однозначно и безошибочно восстановить входной, т. е. обратить вычисления. Если их не сохранить, произойдет рассеяние энергии, а именно этого требуется избежать.

В отличие от обычных схем, обратимые имеют важные ограничения. Во-первых, как уже указывалось, обратимая схема должна осуществ-

лять инъективное преобразование входа. Это требование называют *запретом на дублирование*. Во-вторых, итоговая схема должна быть ациклической, что даёт возможность обратимости при подаче полученных данных по схеме в обратном направлении.

Для реализации обратимой последовательностной логики необходимы ещё и элементы памяти. Они, например, могут содержать две области: рабочую, хранящую информацию для дальнейших вычислений, и вспомогательную, обеспечивающую поддержание обратимости.

### Обратимые комбинационные элементы: основные виды

Простейшими обратимыми элементами являются *Инвертор* (NOT) с одним входом и одним выходом, осуществляющий безусловное инвертирование входа, и *Обмен* (SWAP или EXCHANGE), меняющий на выходе местами свои два входа.

*Контролируемый инвертор* (CNOT, Controlled NOT, элемент Фейнмана, FG) - обратимый гейт, реализующий *условное инвертирование*. Гейт имеет два входа *A*, *B* и два выхода *P*, *Q*, первый вход *A* - управляющий, а второй вход будет инвертирован на выходе, если и только если

Изображения описанных элементов приведены на рис. 1.

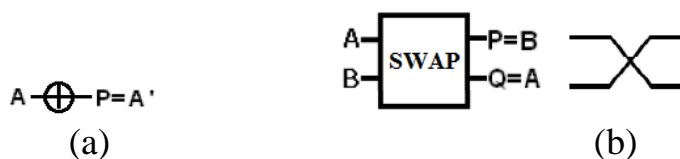


Рис. 1. Изображения элементов (a) NOT, (b) SWAP.

Эти простейшие элементы *вычислительно не универсальны*: используя только их, невозможно реализовать произвольную логическую функцию. Ясно, что универсальный элемент должен обеспечить реализацию функционально полной системы булевых функций, например, пары функций AND2 и NOT, или пары OR2 и NOT. Кроме того, такие элементы должны обеспечивать возможность *каскадирования* -организацию последовательного соединения гейтов. Заметим, что ветвление сигналов в обратимых схемах нельзя организовать «проводным» способом, поскольку это нарушит требование биективности преобразования информации.

Входы рассматриваемых далее элементов будем, как правило, обозначать литерами с начала алфавита: *A*, *B*, ..., а выходы - с середины алфавита: *P*, *Q*, .... На схемах часто входы и выходы обозначают литерами *x*, *y*, ..., а их порядок указывают положением на элементе сверху вниз. В формулах (‘) обозначают инвертирование, знак конъюнкции принято опускать.

*Гейт Тоффоли* (CCNOT, Controlled Controlled NOT, TG) - универсальный контролируемый обратимый  $3 \times 3$ -гейт, реализующий формулы

Очевидно, TG вычислительно универсален: при он реализует инвертирование NOT, а при - конъюнкцию AND2. На элементах Тоффоли можно осуществить операцию ветвления и передачу сигналов, обеспечивая каскадирование: при входном векторе  $(A, 1, 0)$  на его выходе будет вектор  $(A, 1, A)$ .

*Гейт Фредкина* (CSWAP, Controlled SWAP, управляемый обмен, симметрический переключатель, FRG) - универсальный контролируемый обратимый  $3 \times 3$ -гейт. Первый вход  $A$  - управляющий: если он равен 1, то остальные два входа  $B, C$  на выходе поменяются местами, в противном же случае этого не происходит:

Нетрудно видеть, что элемент CSWAP также вычислительно универсален. Сам факт того, что управляемо перемешивая проводки в кабеле между входом и выходом и не используя никаких других элементов, можно получить на выходе значение любой булевой функции от входа, далеко не очевиден.

*Гейт Переса* (PG) -  $3 \times 3$ -гейт со входами  $A$  (управляющий),  $B, C$  и выходами  $P, Q, R$ :

При  $C=0$  гейт Переса реализует полусумматор:  $A$  и  $B$  - суммируемые разряды,  $Q$  - сумма,  $R$  - перенос в следующий разряд и  $P$  - мусорный бит.

Для удобства изображения схем часто элементы изображают указанием соединений ( $\bullet$ ) и операции «сумма по mod 2» ( $\oplus$ ) на параллельных горизонтальных линиях, на которые подаются входные переменные. К схеме возможно добавление дополнительных линий, на которые подается константы 0 или 1 и играющие роль «памяти».

Гейт Переса может быть реализован на гейтах TG и CNOT, соединённых последовательно, как показано на рис. 2 (с).

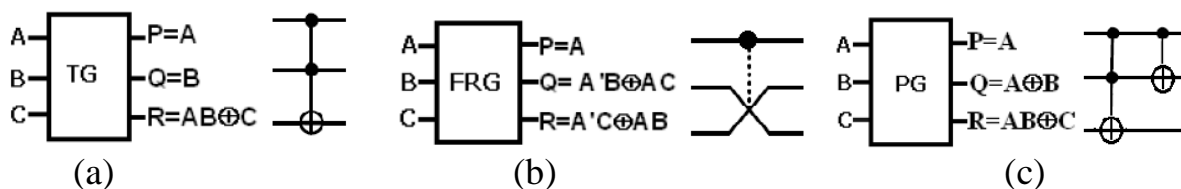


Рис. 2. Изображение элементов (а) TG, (b) FRG, (с) PG.

Рассмотренные элементы описаны, например, в [7, 13-15]. Там же, а также в [8, 16-18] описаны и обратимые гейты других типов.

## Методы синтеза обратимых схем

Для целей синтеза обратимых схем часто приходится разрабатывать новые обратимые элементы, объединяя их с традиционными в библиотеки синтеза. При этом общих методов синтеза не существует: в основном применяют традиционные методы, адаптированные под обратимую схемотехнику [19-22].

Метод отображений основан на изучении таблицы истинности. Идея метода - используя обратимость проводить синтез от выхода к входу, преобразовывая заданную обратимую функцию к тождественной.

PPRM (Positive-Polarity Reed-Muller) - создаётся дерево поиска с корневым узлом, от которого расходятся все выходящие переменные. В последствии из всех возможных представлений лучшее преобразуется в элемент Гоффоли.

ESOP<sup>3</sup>. Метод основан на представлении булевой функции в виде суммы по mod 2 элементарных конъюнкций. Булева функция представляется в виде списка ячеек, каждая из которых соответствует одному или нескольким обратимым элементам.

Метод решающих диаграмм основан на представлении функции алгебры логики в форме *двоичной диаграммы решений* (BDD, Binary Decision Diagram}). Для этого можно применить, например, инструмент CUDD [23] [25]. Далее узлы построенной диаграммы далее преобразовываются в элементы Гоффоли.

RFTPLA. Регулярная структура обратимых отказоустойчивых программируемых логических матриц (RFTPLAs) и даны алгоритмы их построения предложены в [26]. Эти алгоритмы реализованы для представления функций в форме ESOP. Синтезированные схемы имеют минимальное количество ключей и битов мусора.

### Сбоеустойчивость. Модели сбоев. Синтез сбоеустойчивых обратимых схем

Возможны следующие подходы к решению задачи повышения сбоеустойчивых схем: (1) применение самокорректируемых схем, создаваемых на основе различных методов и (2) применение избыточного кодирования [25, 26]. На сегодняшний день в области обратимой схемотехники исследования идут исключительно в рамках первого, традиционного подхода.

*Самокорректируемость* функциональных блоков есть свойство обнаруживать и исправлять неисправности как в основной, так и в дополнительной аппаратуре. Последнее свойство позволяет избежать проблемы

---

<sup>3</sup> SOP - Sum of Product, сумма произведений, дизъюнкция элементарных конъюнкций, ДНФ; ESOP - строгая сумма элементарных конъюнкций

«сторожа над сторожем». В этом случае говорят о *схемной избыточности*. Создание самокорректируемых схем является задачей синтеза вычислительных устройств с дополнительными требованиями.

При сбое вычислительное устройство отклоняется от своего нормального функционирования, что с большой долей вероятности ведёт к ошибкам выходных данных. Для анализа сбоев и разработке методов их парирования рассматривают различные модели возникающих ошибок. Укажем некоторые модели логических неисправностей, используемых при анализе сбоеустойчивости обратимых схем [27-33].

- *Константная ошибка* (stuck-at fault model) - данный бит выхода всегда принимает константное значение 0 или 1.
- *Битовая ошибка* (bit fault model) - инвертирование выходного бита.

Эти модели неисправности используются при исследовании и обычных, необратимых схем.

- *Узловая ошибка* (crosspoint fault model) - специфическая ошибка обратимых элементов Тоффли описанных типов, описывает ошибки возникновения (Appearance Fault) или исчезновения (Disappearance Fault) в нём контрольных разрядов, см. рис. 3.

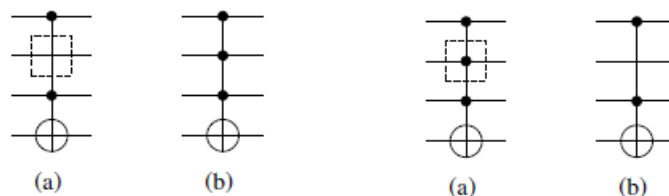


Рис. 3. Схемы появления узловых ошибок возникновения (1) и исчезновения (2): (a) - до появления сбоя, (b) - после.

В простейшем случае сбоеустойчивая схема обеспечивает лишь обнаружение возникшей ошибки. Исправить обнаруженную ошибку далее возможно методом многократного пересчёта выхода, используя, таким образом, *временную избыточность* схемы<sup>4</sup>. На сегодняшний день исследования методов синтеза сбоеустойчивых обратимых схем ограничиваются практически исключительно указанным простейшим вариантом, не приводящий к значительному увеличению сложности схемы.

В [8] дан обзор известных подходов к синтезу обратимых схем. Эти подходы можно разделить на два класса. Первый заключается в построении элементов, обеспечивающих контроль чётности своих выходов и использованием далее общих методов синтеза. Второй связан с приданием свойств контроль чётности уже синтезированных схем.

<sup>4</sup> Понятно, что использование любого вида избыточности, временной или информационной, в конечном счёте реализуется дополнительными схемами.

Большинство методов находятся в рамках первого класса. Разработан универсальный способ преобразования произвольных обратимых элементов в гейты, сохраняющие чётность.

Методы второго класса часто требуют значительного перепроектирования уже имеющихся схем. При этом увеличивается количество ключей, добавляются новые проверяющие элементы и увеличивается мусор.

В последнее время предложены следующие схемы сбоеустойчивых вычислительных устройств, использующих контроль чётности; все ссылки см. в [34]:

- полный сумматор из сохраняющих чётность блоков (2010);
- АЛУ (2013);
- компрессор (устройство для сжатия динамического диапазона звукового сигнала);
- полный сумматор (2015).

Отметим, что существует простейший метод *троирования* или *тройного модульного резервирования* (TMR, Triple Modular Redundancy) построения сбоеустойчивых, при котором результат определяется *мажорированием* «голосования 2 из 3» выходов трёх экземпляров основной схемы. Данную схему мажорирования называют *воутером* (от англ. voter).

## Помехоустойчивое кодирование в хэмминговом пространстве

Предложим новый метод построения сбоеустойчивых обратимых элементов на основе помехоустойчивого кодирования в пространстве Хэмминга [35]. Метод заключается в замене обратимых гейтов на их сбоеустойчивые аналоги, обеспечивающих гарантированное автоматическое исправление любой одиночной ошибки, т. е. основанный на принципе селективной защиты на уровне отдельных элементов.

В простейшем случае помехоустойчивого кодирования будем кодировать булевы значения 0 и 1 тремя битами 000 и 111, используя 3 проводника вместо одного и называя данные 3-битовые значения *Полюсом\_0* и *Полюсом\_1* соответственно. Пространство  $n$ -мерного единичного куба называют *пространством Хэмминга*; мы будем работать в 3-мерном таком пространстве. Булевы операции будем производить над сигналами как над соответствующими полюсами. При возникновении не более чем одиночной ошибки хэммингово расстояние (число несовпадающих бит) не превосходит 1. Коррекция ошибки происходит автоматически в неявном виде.

Для обратимой схемотехники это существенно: происходит исправление ошибки, а не просто фиксация факта, что ошибка произошла.

Нетрудно заметить, что предлагаемый подход имеет сходство с методом TMR на уровне элементов. Различие заключается в том, что при троировании элемента имеется воутер, который не защищен от ошибок; т.е. один элемент защищается элементами, каждый из которых также подвержен сбоям. При использовании метода кодирования в пространствах Хэмминга эта проблема принципиально отсутствует [36]. Дело в том, что воутером в этом подходе выступают последующие элементы в схеме, воутером которых в свою очередь, являются следующие за ними и т.д. Это обеспечивает исправление любой однократной битовой ошибки элемента. Более того, предлагаемый метод имеет существенные преимущества относительно многократных ошибок. Строго говоря, любое число кратных ошибок гарантированно исправляется, при условии, что на один расширенный элемент приходится не больше одного сбоя.

Очевидной платой за столь высокий уровень сбоеустойчивости являются значительные аппаратные затраты. Кратное увеличение числа используемых ключей делают этот метод вряд ли применимым для всей схемы, оставляя его целесообразным для наиболее уязвимых элементов и подсхем, сбоеустойчивость которых критически важна для функционирования всей вычислительной системы.

На рис. 4 представлен<sup>5</sup> обратимый воутер «2 из 3», который будем обозначать  $V1(x_1, x_2, x_3)$ , реализующий на выходе функцию  $x_1x_2+x_1x_3+x_2x_3$  от входов  $x_1, x_2$  и  $x_3$ .

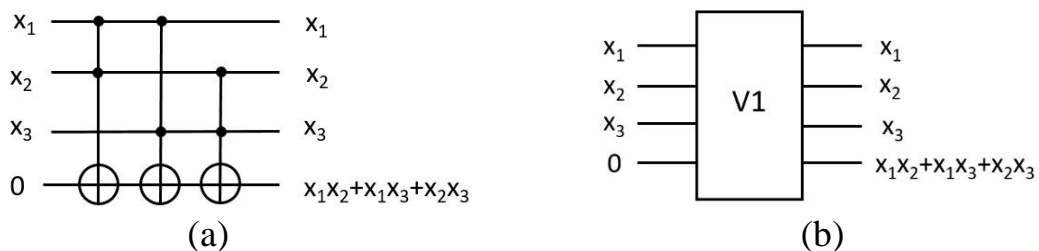


Рис. 4. Обратимый воутер «2 из 3» с одним сигнальным выходом: (а) схема на элементах Тoffоли, (б) изображение в виде блока.

Троированный сигнал  $A$  в пространстве Хэмминга будем обозначать нижними индексами от 1 до 3:  $A_1, A_2, A_3$ , и аналогично для других сигналов.

Гейт HNOT, с 6 выходами и входами, реализующий инвертирование в пространстве Хэмминга, показан на рис. 5. Он построен «последовательным соединением» трёх воутеров  $V1$  с инвертированием их сигнальных выходов и, очевидно, содержит 12 элементов суммирования по mod 2.

<sup>5</sup> Это одна из возможных реализаций; то же относится и к другим элементам, рассмотренным далее. Авторы старались найти и представить наиболее простые схемы.



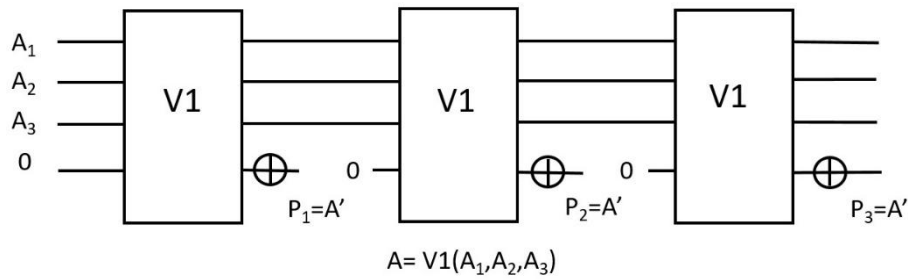


Рис. 5. Элемент HNOT

На рис. 6 представлена имеющая 15 вентилях суммирования по mod 2 схема  $10 \times 10$  управляемого инвертирования HCNOT в пространстве Хэмминга.

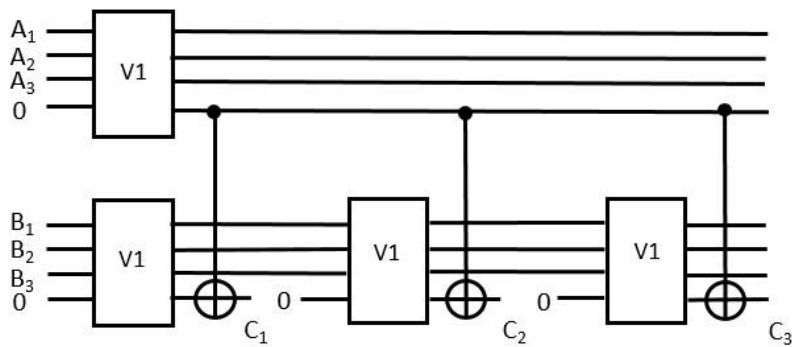


Рис. 6. Гейт HCNOT управляемого инвертирования.

Гейт Тоффоли НТГ в пространстве Хэмминга представлен на рис.7.

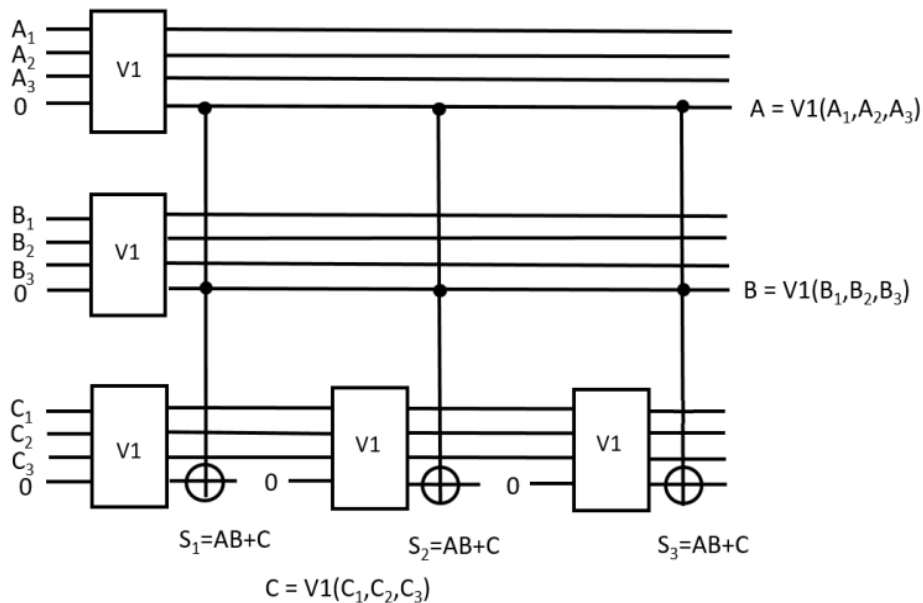


Рис. 7. Гейт Тоффоли НТГ в пространстве Хэмминга

Этот  $14 \times 14$  элемент образован пятью воутерами V1 и трёх схем суммы обычных гейтов Тоффоли. Сигнальные выходы всей схемы суть  $S_1$ ,  $S_2$  и  $S_3$ .

«Облегчённый» HLTG  $11 \times 11$  элемент Тоффли в пространстве Хэмминга с исправлением возможной ошибки на  $C_1, C_2, C_3$  на входах гейта схемы показан на рис. 8.

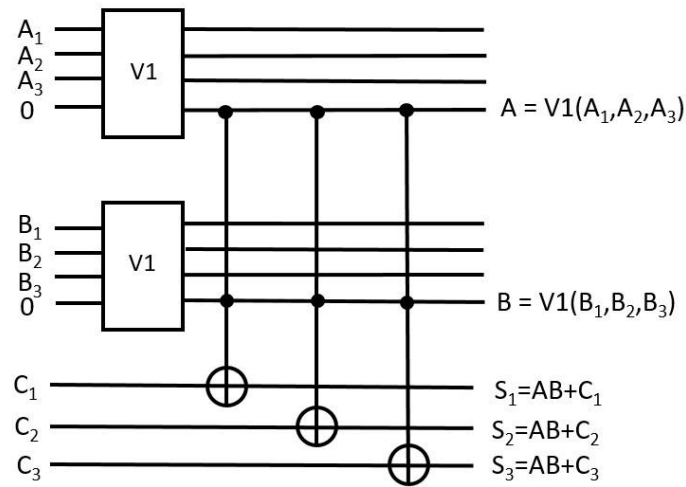


Рис. 8. Облегчённая схема гейта HLTG в пространстве Хэмминга.

Элемент HFRG – гейт Фредкина в Хэмминговом пространстве показан на рис. 9.

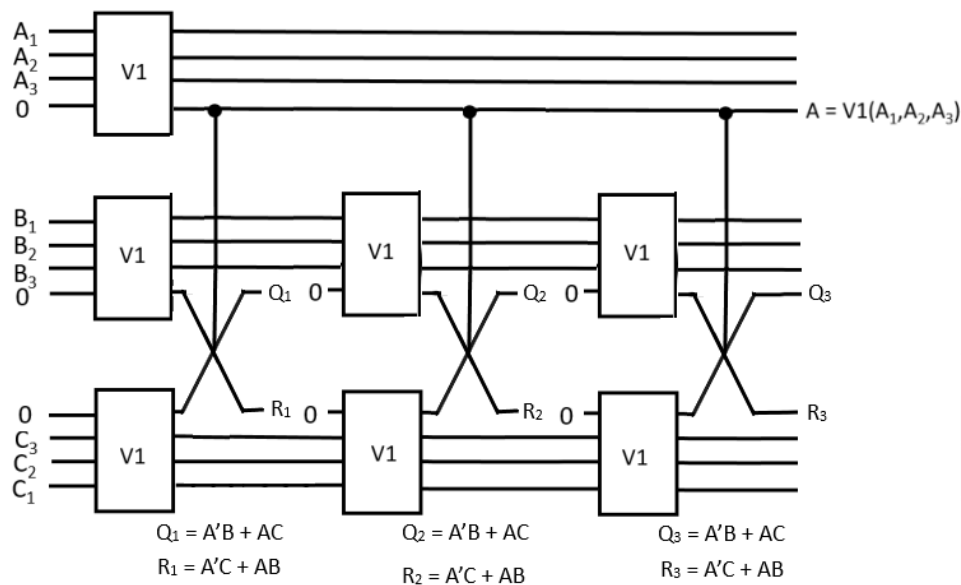


Рис. 6. Элемент HFRG.

Ясно, что обратимый полусумматор в пространстве Хэмминга может быть реализован последовательным соединением гейтов HTG и HCNOT. Схема полного одноразрядного сумматора HADD показана на рис. 10 (на нём можно выделить схему полусумматора).

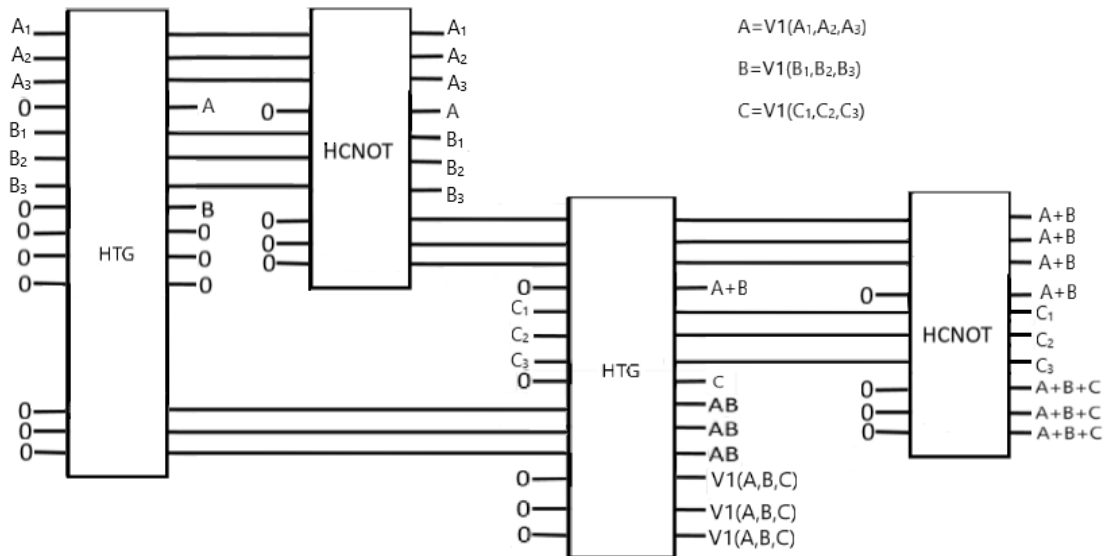


Рис. 10. Обратимый сбоеустойчивый сумматор HADD.

Сравнение сложности и моделирование сбоеустойчивости обычных необратимых схем с их аналогами, построенных на элементах в хэмминговом пространстве позволило сделать следующие выводы [36].

Во-первых, число вентилях в методе тройного резервирования возрастает примерно в четыре раза, в то время как при кодировании в пространстве Хэмминга оно возрастает на порядок. Примерно то же мы видим и в случае обратимых схем. Действительно, сложность элемента (gate cost) принято оценивать количеством символов  $\oplus$  на их изображениях; например, сложность TG есть 1. Тогда сложность элементов V1 и B3 равны 3 и 2 соответственно, для синтеза гейта НТГ потребовалось 18 элементов TG, а для НЛТГ – 9. Сложность  $\oplus$ -элемента сумматора HADD есть 66.

Во-вторых, сбоеустойчивость предлагаемого метода во всех случаях превосходит метод тройного резервирования, что также продемонстрировано при соответствующем сравнении.

В целях более сбалансированного представления сигналов 0 и 1, и, соответственно, нагрузки на транзисторные вентили при реализации схем на кристалле, более эффективным является выбор при кодировании в пространстве Хэмминга в качестве *Полюса\_0* некоторого 3-битного вектора единичного веса и, соответственно, инверсного ему в качестве *Полюса\_1*. Для этого, например, могут быть выбраны векторы 010 и 101. Первый назовём *вектором поляризации*, он является двоичным представлением числа 2, а второй, очевидно, есть двоичное представление 5. Переход к такому *поляризованному хеммингову пространству* геометрически эквивалентен вращению единичного куба вокруг своего центра с переносом нулевого вектора в вершину, представляемую вектором поляризации.

Несложно понять, как изменятся рассмотренные схемы с учётом такого представления. Воутер V1, например, в рассматриваемом поляризованном пространстве Хэмминга будет отличаться от представленного на рис. 4 инвертором  $\oplus$  на входе  $x_2$ .

Чтобы отличать схемы в указанном поляризованном пространстве от ранее построенных, литеру Н в их обозначении будем снабжать нижним индексом 2. На рис. 11 представлен элемент  $H_2NOT$ , реализующий инвертирование в поляризованном пространстве Хэмминга с вектором поляризации 010.

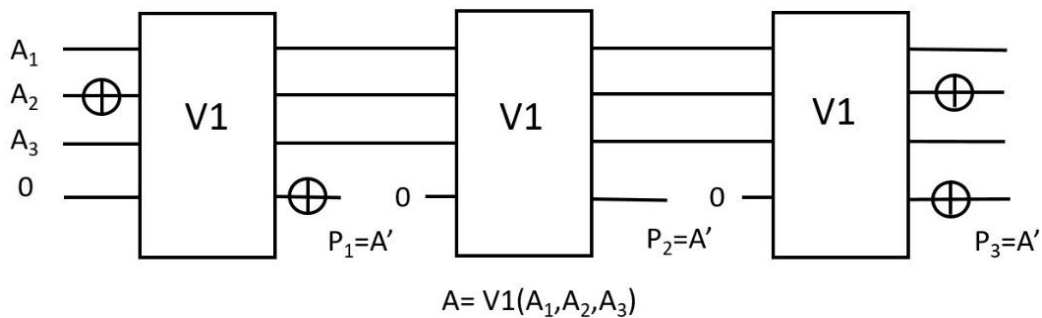


Рис. 7. Элемент  $H_2NOT$

### Заключение

В работе рассмотрены основные понятия, связанных с обратимыми вычислениями. Данная новая, альтернативная существующей, парадигма вычислений обеспечивает принципиальную возможность выхода из ситуации «теплового проклятия». Это послужило причиной их более детального изучения обратной схемотехники, в частности, сбоеустойчивости и тестируемости синтезируемых схем. Благодаря указанным преимуществам, обратимая схемотехника отвечает основным требованиям современной технологии ВТ.

В последнее время проектные нормы ИМС уменьшаются до единиц нанометров, что влечёт резкое снижение порога энергии заряженных частиц, вызывающего сбой. В связи с этим предложен оригинальный метод синтеза сбоеустойчивых обратимых элементов в поляризованном пространстве Хэмминга, обладающий высокой способностью корректировать как единичные, так и многократные ошибки.

Предложенные сбоеустойчивые схемы уже являются важными для обратной схемотехники. В то же время на сегодняшний день настоятельной необходимостью является создание таких обратимых сбоеустойчивых стандартных элементов, как мультиплексор, дешифратор, триггеры, полный сумматор на требуемое количество разрядов и др. Также ждёт своего развития общая теория синтеза обратимых сбоеустойчивых схем.

## Литература

1. Frank, Michael P.: The physical limits of computing, IEEE Computing in Science and Engineering Magazine, May/June 2002.
2. Landauer, R.: Irreversibility and heat generation in the computing process, IBM Journal of Research and Development, 5: 183-191, July 1961.
3. Merkle R. C. Drexler K. E.: Helical logic // Nanotechnology, 7, 4 (1996), 325-339.
4. Bennet, C.: Logical Reversibility of Computation, IBM Journal of Research and Development, vol. 17, no. 6, pp. 525-532, 1973.
5. Бобровский С. Будет ли обратимым зеттафлопсный компьютер? // PC Week/RE (474) 12'2005.
6. Kalyan, S. Perumalla.: Introduction to Reversible Computing. - CRC Press, 2014.
7. Saeedi, M., Markov, I. L.: Synthesis and Optimization of Reversible Circuits - A Survey. Oct 2011. [<https://arxiv.org/pdf/1110.2574>].
8. Jain, A.: Fault Tolerant Synthesis of Reversible Circuits, A Dissertation submitted in partial fulfillment for the award of degree of Master of Technology (with specialization in Computer Science) in Department of Computer Science and Engineering, CoRR. [<https://arxiv.org/abs/1310.5231v2>], [<https://dblp.org/rec/bib/journals/corr/Jain13>].
9. Закаблукоев Д.В. Методы синтеза обратимых схем из функциональных элементов NOT, CNOT и 2-CNOT. - Дисс. ... канд. физ.-мат. наук: 01.01.09: защищена 31.05.2018. - М., 2018.
10. Юдинцев В. Радиационно-стойкие интегральные схемы. Надежность в космосе и на земле // Электроника: Наука, технология, бизнес, 2007. Вып. 5. С. 72-77.
11. Попович А. Топологическая норма и радиационная стойкость // Компоненты и Технологии, Выпуск № 110, 2010. С. 100-102.
12. Телец В., Цибин С., Быстрицкий А., Подъяпольский С. ПЛИС для космических применений. Архитектурные и схемотехнические особенности // ЭЛЕКТРОНИКА: Наука, Технология, Бизнес. 6/2005. - С. 44-48.
13. Feynman, R.: Quantum mechanical computers, Optical News, vol. 11, 1985, pp. 11--20.
14. Toffoli, T.: Reversible computing, In Automata, Languages and Programming, Springer-Verlag, pp. 632-644, 1980.
15. Fredkin, E., Toffoli, T.: Conservative Logic, International Journal of Theoretical Physics, Vol. 21, No. 3/4, 1982, Pp. 632--644.
16. Wille, R., Große, D, Teuber, L., Dueck, G. W. and Drechsler, R. Revlib: An online resource for reversible functions and reversible circuits, International Symposium on Multiple Valued Logic, pp. 220-225, May 2008.

17. Maslov, D. Dueck, G. W. and Scott, N.: Reversible logic synthesis benchmarks page. [<http://www.cs.uvic.ca/~dmaslov/>].
18. Бобков С.Г. Высокопроизводительные вычислительные системы / под ред. академика РАН В. Б. Бетелина. - М.: НИИСИ РАН. - 2014.
19. Miller, D., Maslov, D. and Dueck, G. W.: A transformation based algorithm for reversible logic synthesis, In Proceedings of the 40th annual Design Automation Conference (DAC), pp. 318-323, 2003.
20. Gupta, P., Agrawal, A. and Jha, N. K.: An algorithm for synthesis of reversible logic circuits, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 25 (11): 2317-2330, 2006.
21. Fazel, K., Thornton, M. and Rice, J. E.: ESOP-based Toffoli gate cascade generation, In Proceedings of the IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM), pp. 206-209, Victoria, BC, Canada, 22-24 Aug. 2007.
22. Wille, R. and Drechsler, R.: BDD-based synthesis of reversible logic for large functions, In Proceedings of the 46th Annual Design Automation Conference (DAC), pp. 270-275, 2009.
23. Somenz, F.: CUDD: CU Decision Diagram Package Release 2.3.1, University of Colorado at Boulder, 2001.
24. Rahman, R., Jamal, L., Babu, H. M. H.: Design of reversible fault tolerant programmable logic arrays with vector orientation. Int. J. Inf. Commun. Technol. Res., 1 (8), pp. 337-342 (2011).
25. Согомоян Е. С., Слабаков Е. В. Самопроверяемые устройства и отказоустойчивые системы. - М.: Радио и связь, 1989.
26. Щербаков Н. С. Самокорректирующиеся дискретные устройства. - М.: Машиностроение, 1975.
27. Lala, P. K.: An Introduction to Logic Circuit Testing, Morgan & Claypool, 2008.
28. Patel, K. N., Hayes, J. P. and Markov, I. L.: Fault testing for reversible circuits, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 23 (8): pp. 1220-1230, 2004.
29. Rice, J. E.: An Overview of Fault Models and Testing Approaches for Reversible Logic, accepted for presentation at the 2013 Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM), Aug. 2013, Victoria, BC, Canada.
30. Nayeem, N. M. and Rice, J. E.: Online fault detection in reversible logic, In Proceedings of the IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems (DFT), Vancouver, Canada, October 2011.
31. Vasudevan, D. P., Lala, P. K. and Parkerson, J. P.: Online testable reversible logic circuit design using NAND blocks, In Proceedings of

- IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems, pp. 324–331, Los Alamitos, CA, USA, 10-13 October 2004.
32. Mahammad, S. N., Hari, S., Shroff, S. and Kamakoti, V.: Constructing online testable circuits using reversible logic, In Proceedings of 10th IEEE International VLSI Design and Test Symposium (VDATE), pp. 373–383, Goa, India, August 2006.
  33. Zhong, J. and Muzio, J. C.: Analyzing fault models for reversible logic circuits, In Proceedings of IEEE Congress on Evolutionary Computation (CEC), pp. 2422-2427, Vancouver, BC, 2006.
  34. Przigoda, N., Dueck, G., Wille R. and Drechsler, R.: Fault Detection in Parity Preserving Reversible Circuits. In IEEE 46th International Symposium on Multiple-Valued Logic (ISMVL), 2016, pp. 44-49.
  35. Alagoz, B. Baykant.: Boolean Logic with Fault Tolerant Coding, OncuBilim Algorithm and Systems Labs, 2009, Vol. 09, Art. No: 03. [<https://arxiv.org/ftp/arxiv/papers/0903/0903.4046.pdf>]
  36. Стемповский А. Л., Тельпухов Д. В., Жукова Т. Д., Гуров С. И., Соловьев Р. А. Методы синтеза сбоеустойчивых комбинационных КМОП схем, обеспечивающих автоматическое исправление ошибок // Известия ЮФУ. Технические науки. 2017. № 7 (192). - С. 197-210.