УДК 004.942

АЛГОРИТМ КОНТРОЛЯ ВЫПОЛНЕНИЯ ЛОГИЧЕСКИХ ОПЕРАЦИЙ В ПРОГРАММИРУЕМЫХ ЛОГИЧЕСКИХ ИНТЕГРАЛЬНЫХ СХЕМАХ, ВЫПОЛНЯЮЩИХ КРИПТОГРАФИЧЕСКОЕ ПРЕОБРАЗОВАНИЕ ИНФОРМАЦИИ

Лукин М.В.

Научно-исследовательский центр Центрального научно-исследовательского института Военно-воздушных сил (Министерства обороны Российской Федерации), Щелково, e-mail: maxim@vishnin.ru

В статье приведены разработанные алгоритмы контроля выполнения логических операций (сложение по модулю 2, сложение по модулю 2 в 16-й степени, операция сдвига и подстановки) в программируемых логических интегральных схемах, используемых в качестве основных вычислительных компонентов (логических вентилей) в устройствах, осуществляющих криптографическое преобразование входной информации. Данные логические операции являются основными преобразованиями в криптографии. Преобразование осуществляется с использованием блочного симметричного шифра, в основе которого лежат ключ и псевдослучайная последовательность, называемая синхропосылкой. Данные алгоритмы позволяют блоку управления автономным техническим средством определять правильность выполнения одних из основных операций криптографического преобразования информации с целью выявления и, по возможности, устранения сбоев в работе. Благодаря использованию указанных алгоритмов повышаются такие важные характеристики, как надежность и достоверность. При использовании средних по производительности программируемых логических интегральных схем ущерб в скорости обработки входной информации сводится к минимуму, что позволяет функционировать автономному техническому средству без каких-либо задержек, но с более высокой эффективностью. Данные алгоритмы наиболее актуально использовать в автономных технических средствах, осуществляющих передачу телеметрической информации по радиоканалам с большого удаления по расстоянию от оператора автономного технического средства.

Ключевые слова: программируемая логическая интегральная схема, криптографическое преобразование информации, автономное техническое средство, алгоритм контроля, сложение по модулю

ALGORITHM FOR CONTROLLING THE EXECUTION OF LOGICAL OPERATIONS IN PROGRAMMABLE LOGIC INTEGRATED CIRCUITS PERFORMING CRYPTOGRAPHIC TRANSFORMATION OF INFORMATION

Lukin M.V.

Research Center of the Central Research Institute of Air Force (Ministry of Defense of the Russian Federation), Shchelkovo, e-mail: maxim@vishnin.ru

The article presents the developed algorithms for controlling the execution of logical operations (addition modulo 2, addition modulo 2 to the 16th degree, shift and substitution operation) in programmable logic integrated circuits used as the main computational components (logic gates) in devices performing cryptographic transformation of input information. These logical operations are the main transformations in cryptography. The conversion is carried out using a block symmetric cipher, which is based on a key and a pseudorandom sequence called a synchro link. These algorithms allow the control unit of an autonomous technical means to determine the correctness of performing one of the basic operations of cryptographic transformation of information in order to identify and, if possible, eliminate malfunctions. Thanks to the use of these algorithms, such important characteristics as reliability and reliability are increased. When using medium-performance programmable logic integrated circuits, the damage in the processing speed of input information is minimized, which allows the autonomous technical means to function without any delays, but with higher efficiency. These algorithms are most relevant to use in autonomous technical means that transmit telemetry information over radio channels from a great distance away from the operator of an autonomous technical means.

Keywords: programmable logic integrated circuit, cryptographic transformation of information, autonomous technical means, control algorithm, addition modulo

Криптографическим преобразованием информации (далее – КПИ) называется процесс изменения информации, зависящий от изменяемого параметра и обладающий свойством невозможности восстановления исходной информации по преобразованной, без знания действующего ключа, с трудоемкостью меньше заданной [1].

В настоящее время КПИ осуществляется программными, аппаратными и программно-аппаратными средствами. При этом в случае даже минимальной ошибки в 1 бит процесс восстановления информации становится невозможным.

Особенно актуальна проблема возникновения ошибки в ходе КПИ на автономных технических средствах (далее – АТС), осуществляющих работу на большом расстоянии от технического оператора и передающих криптографически преобразованную информацию на объект обработки по радиоканалам [2].

В связи с этим возникает необходимость автономного контроля выполнения КПИ на АТС для заблаговременного определения ошибки и, по возможности, устранения сбоя в работе устройства, осуществляющего КПИ.

Основным вычислительным устройством, выполняющим КПИ, в настоящее время служит программируемая логическая интегральная схема. Данное вычислительное средство является наиболее универсальным для реализации различных алгоритмов КПИ.

В основе наиболее криптостойких алгоритмов лежат четыре основные логические операции [3]:

- операция сложения по модулю 2;
- операция сложения по модулю 2^{16};
- операция циклического сдвига;
- операция подстановки.

Контроль данных логических операций позволит наиболее полно определять техническое состояние ПЛИС, осуществляемой КПИ на АТС.

Рассмотрим числовой контроль логических операций. В основе построения схем контроля лежат две теоремы [4, 5].

Теорема 1. Сумма чисел сравнима по модулю q с суммой остатков r этих же чисел, то есть:

$$\sum_{i=1}^{n} A_i = \sum_{i=1}^{n} r_{ai} \bmod q \qquad (1)$$

Теорема 2. Произведение чисел сравнимо по модулю q с произведением остатков r этих же чисел, то есть:

$$\prod_{i=1}^{n} A_i = \prod_{i=1}^{n} r_{ai} \bmod q \qquad (2)$$

Рассмотрим схему контроля для логической операции сложения по модулю 2.

Схема контроля преобразования [6] (суммирования) двух чисел A_1 и A_2 по модулю два представлена на рисунке 1.

Поясним работу данной схемы следующим образом [7, 8]. Результатом суммирования двух чисел A_1 и A_2 есть число A_3. В блоках B_1 и B_2 осуществляется вычисление остатков r_1 и r_2 чисел A_1 и A_2 до преобразования соответственно. Далее в блоке S происходит суммирование полученных остатков. Поскольку сумма остатков может быть больше модуля, то на выходе сумматора необходимо еще раз выполнить операцию нахождения остатка с помощью преобразования в блоке B_3. В результате выполняется сравнение остатка r_3, полученного от числа A_3, с суммой остатков чисел A_1 и A_2 – r_Σ с формированием признака «норма» (P) в виде логической единицы и нуля в противном случае.

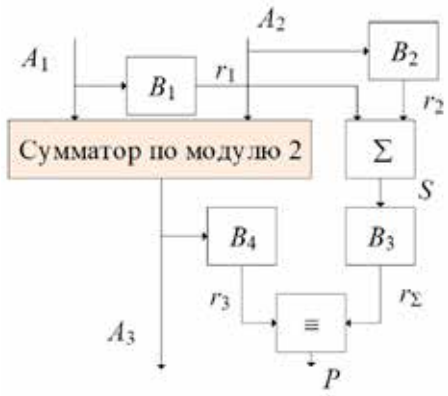

Рис. 1. Схема контроля суммирования по модулю 2

Алгоритм функционирования схемы контроля операции сложения по модулю 2 представлен на рисунке 2.

Условно работу данного алгоритма можно разбить на несколько этапов.

1. На начальном этапе осуществляются получение чисел A_1 и A_2, участвующих в операции сложения по модулю 2, а также получение остатков r_1 и r_2 из данных чисел.

2. На следующем этапе выполняются операция сложения по модулю 2 чисел A_1 и A_2 (получение числа A_3), а также суммирование остатков r_1 и r_2 ($S = r_1 + r_2$).

3. На третьем этапе осуществляется поиск остатков от результата суммирования S и A_3.

4. На заключительном этапе осуществляется сравнение остатков от суммы чисел A_1 и A_2 (числа A_3) и суммы остатков r_1 и r_2 (S).

С формальной точки зрения данные этапы могут быть представлены совокупностью отображений:

$$B_1 : A_1 \to r_1, \qquad (3)$$

где B_1 – оператор, характеризующий процедуру получения остатка r_1 от числа A_1;

$$B_2 : A_2 \to r_2, \qquad (4)$$

где B_2 – оператор, характеризующий процедуру получения остатка r_2 от числа A_2;

Необходимо отметить, что выражения (3) и (4) являются эквивалентными по функциональному представлению операторов B_1 и B_2.

$$S : \sum_{i=1}^{2} r_i \to c, \qquad (5)$$

где S – оператор, формализующий нахождение суммы остатков слагаемых r_1 и r_2, полученных в результате отображений (3) и (4);

$$B_3 : c \to r_\Sigma, \qquad (6)$$

где B_3 – оператор, характеризующий процедуру получения остатка r_Σ от суммы остатков r_1 и r_2;

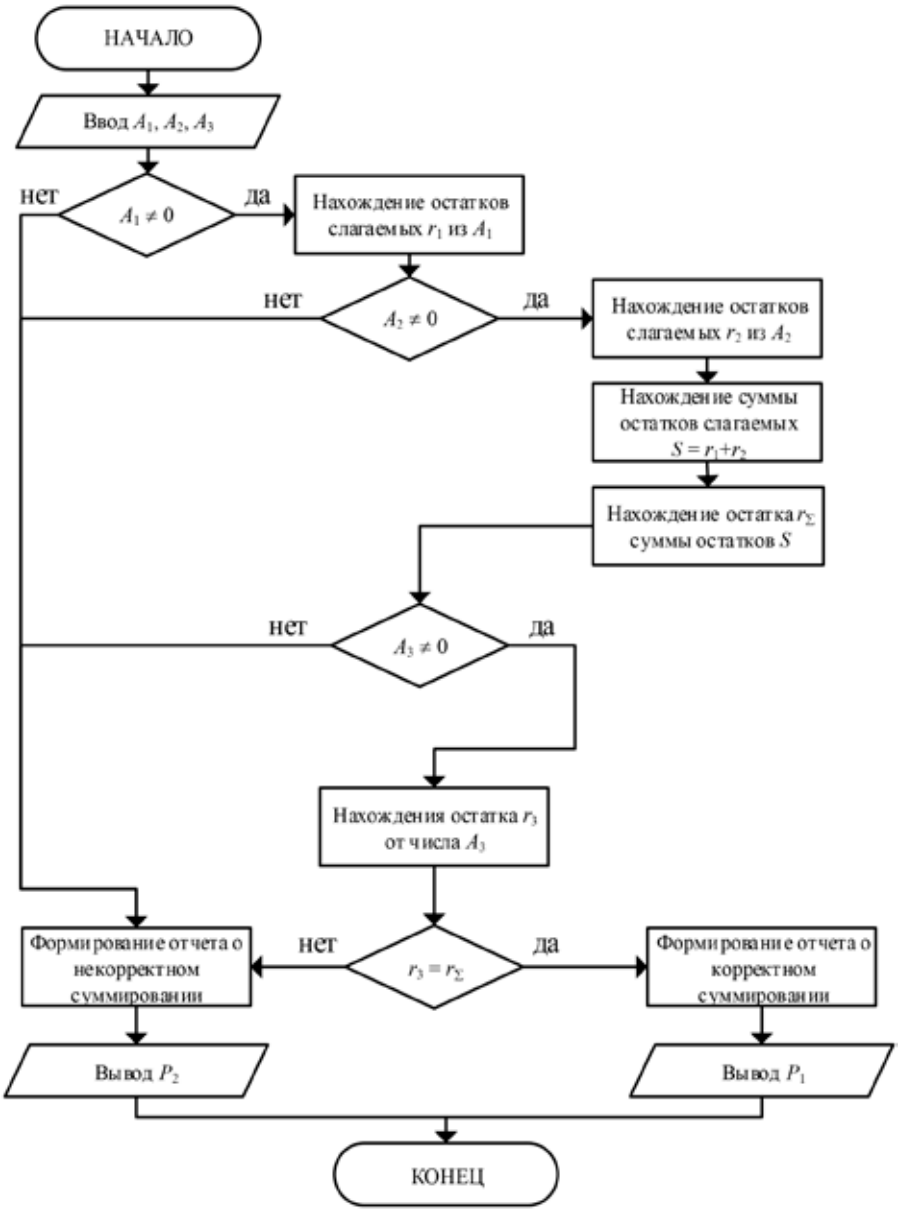

Рис. 2. Алгоритм выполнения контроля операции сложения по модулю 2

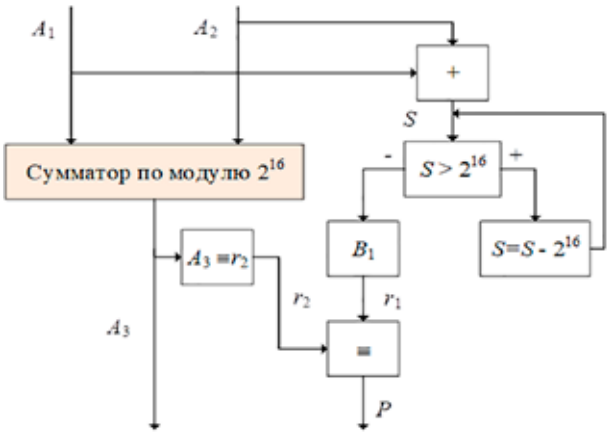

Рис. 3. Схема контроля суммирования по модулю 2^{16}

$$B_4 : A_3 \to r_3, \quad (7)$$

где B_4 – оператор, характеризующий процедуру получения остатка r_3 от суммы чисел A_1 и A_2 (числа A_3);

$$\alpha : r_3 \vee r_\Sigma \to p, p = \{0,1\}, \quad (8)$$

где α – оператор, формализующий процедуру сравнения r_3 и r_Σ.

При контроле сложения по модулю 2^{16} схема упрощается (рис. 3) с учетом того, что вычисляется остаток двух чисел в самом преобразовании.Необходимо отметить, что в данном случае результатом выполнения суммирования по модулю 2^{16} является получение непосредственно остатков чисел, что с практической точки зрения позволяет упростить реализацию процесса контроля. В свою очередь, формирование признака осуществляется аналогично описанному выше случаю.

Алгоритм функционирования представленной схемы контроля представлен на рисунке 4.

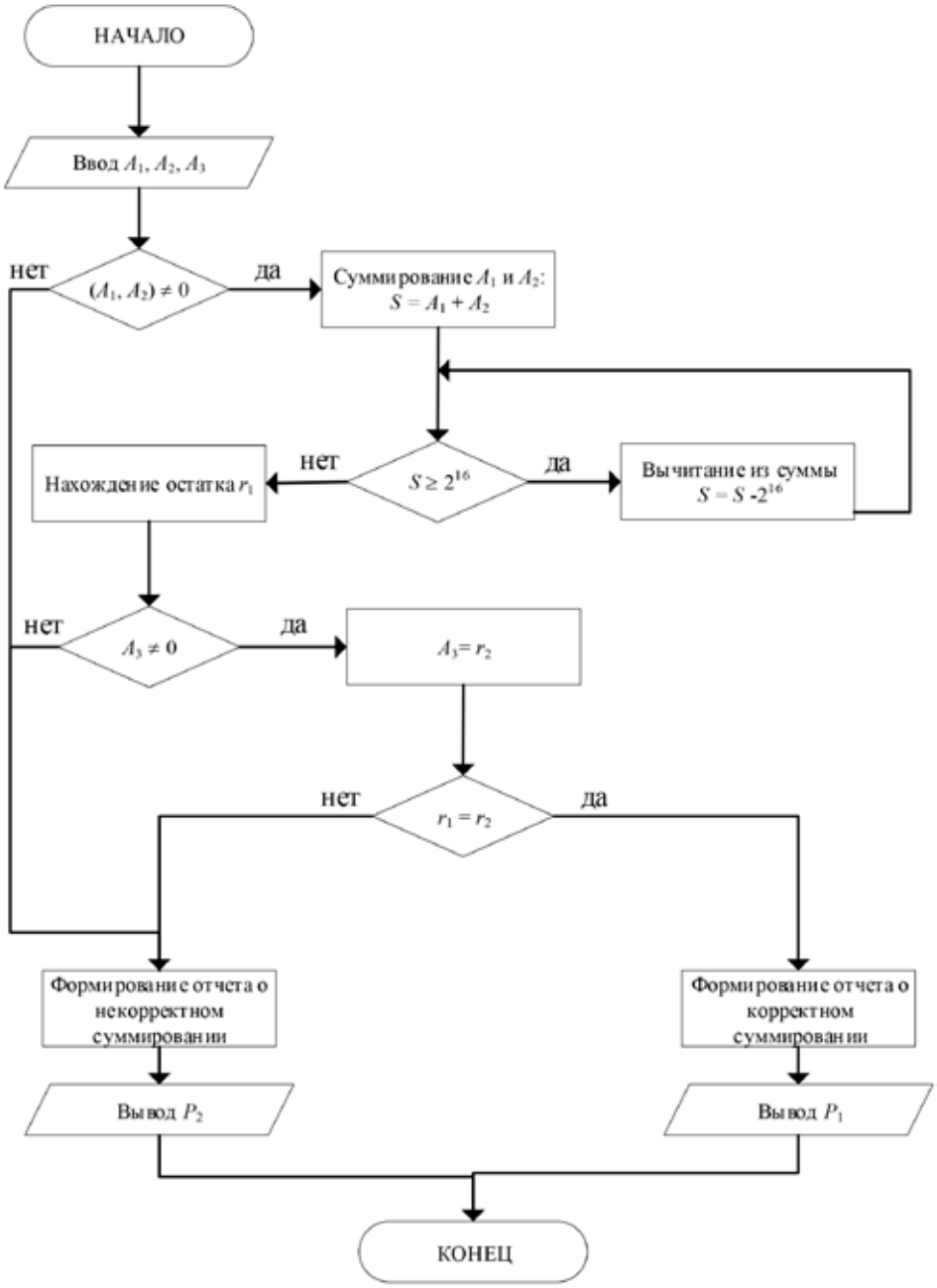

Рис. 4. Схема выполнения контроля операции сложения по модулю 2^{16}

Условно работу представленного алгоритма также можно разбить на несколько аналогичных этапов, представленных ранее. Отличие в данном случае заключается в отсутствии процедуры нахождения остатков при реализации отображений (3) и (4).

Следующим основным преобразованием, участвующим в алгоритмах КПИ, является операция сдвига. На рисунке 5 представлена обобщенная схема операции циклического сдвига.

Работу данной схемы можно описать следующим образом. Результатом сдвига числа A_1 есть число A_2. Для блоков B_1 и B_2 осуществляется разделение битовой последовательности двоичного представления числа A_1 на два блока $A_1 = \left\{ A_1^1, A_2^1 \right\}$, которые в ходе выполнения операции будут переставлены относительно старшего и младшего разрядов (рис. 6).

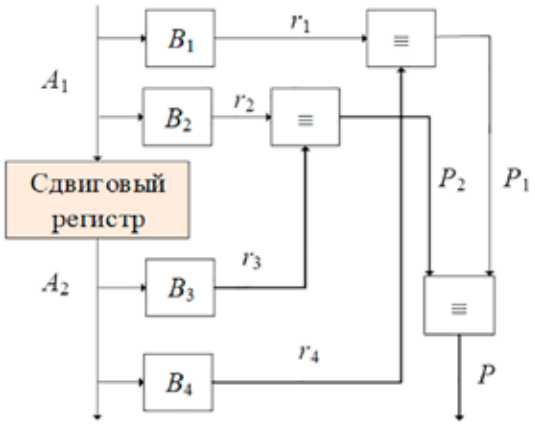

Рис. 5. Схема выполнения контроля операции сдвига

Поясним отдельно данное утверждение: в блоке B_1 преобразуется в остаток r_1 та часть двоичной последовательности (A_1^1), которая в дальнейшем будет участвовать в операции сдвига и в полном объеме будет смещена циклически в сторону младшего разряда; в свою очередь в блоке B_2 преобразуется в остаток r_2 та часть двоичной последовательности (A_2^1), которая будет смещена в сторону старшего разряда и будет переписана на месте двоичной последовательности числа A_1^1. Далее в блоке сдвигового регистра происходит выполнение операции сдвига, в результате которой появляется число A_2. Поскольку сдвиг является циклическим, смещенные биты не исчезают, а заполняют освободившиеся разряды с противоположной стороны битовой последовательности.

Исходя из этого, разделив битовую последовательность двоичного представления числа A_2 на A_1^2 и A_1^2, возможно осуществить контроль правильности выполнения операции сдвига следующим образом. В блок B_3 (формирование остатка r_3) необходимо записать количество разрядов, равное количеству разрядов для блока B_2. Аналогично заполняется блок B_4 (формирование остатка r_4). После выполнения операции сдвига, сравнив остатки r_1, r_4 и r_2, r_3 блоков B_1, B_2, B_3 и B_4, можно сделать вывод о правильности выполнения циклического сдвига, тем самым осуществив контроль правильности выполнения операции. Необходимо отметить, что в блоках сравнения сопоставляются не целые значения чисел, а их остатки, что повышает скорость процесса сравнения сколь угодно больших блоков информации.

Алгоритм функционирования представленной схемы контроля представлен на рисунке 7. Работа представленного алгоритма может быть разбита на несколько этапов.

1. На начальном этапе осуществляются получение чисел A_1 и A_2, участвующих в операции сдвига, а также разбиение числа A_1 на два блока A_1^1 и A_2^1 битовой последовательности. Также на данном этапе происходит нахождение остатков r_1 и r_2 в блоках B_1 и B_2 соответственно.

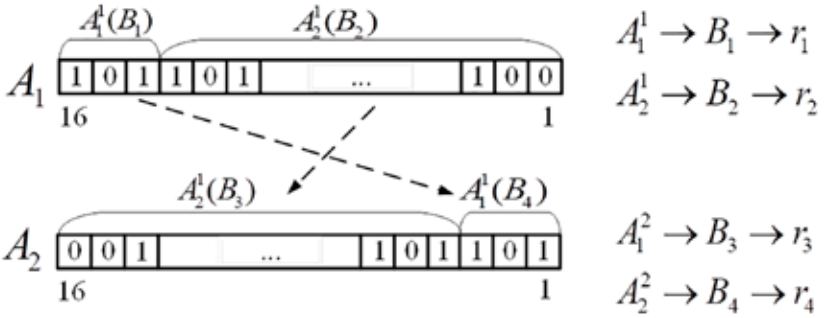

Рис. 6. Сдвиг в сторону старшего разряда на 3

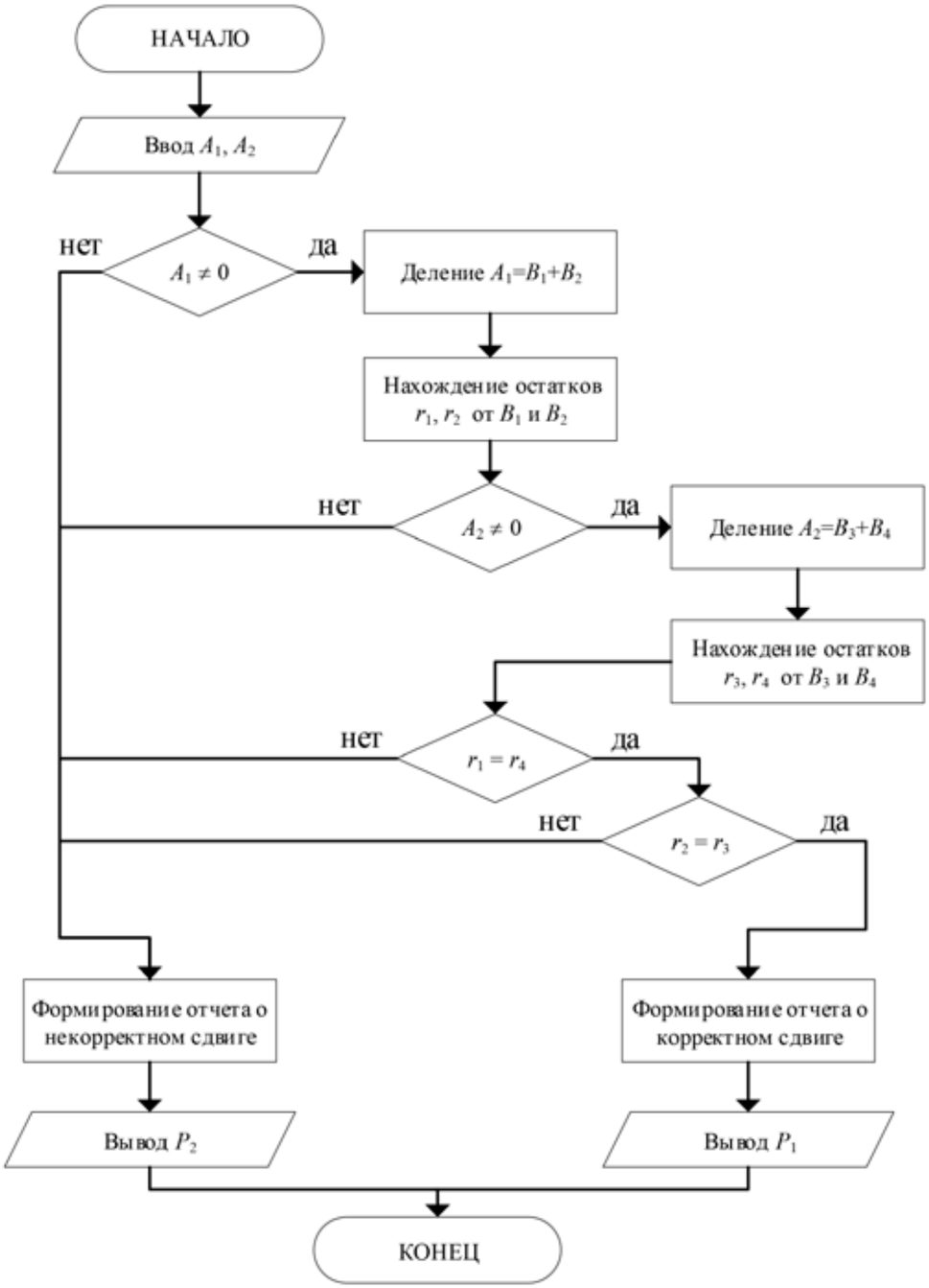

Рис. 7. Алгоритм выполнения контроля операции сдвига

2. Аналогично на следующем этапе осуществляются разбиение числа A_2 на два блока B_3 и B_4 битовой последовательности и нахождение остатков r_3 и r_4 от блоков B_3 и B_4 соответственно.

3. На заключительном этапе осуществляются сравнение остатков r_1 и r_4 от блоков битовой последовательности B_1 и B_4, остатков r_2 и r_3 от блоков B_2 и B_3, а также формирование отчета о корректном или некорректном сдвиге (вывод P_1 и P_2 соответственно).

Представим данные этапы в виде совокупности отображений:

$$\alpha^* : A_1 \to \sum_{i=1}^{2} A_i^1, \qquad (9)$$

где α* – оператор, характеризующий процедуру разбиения числа A_1 на битовые последовательности A_1^1 и A_2^1 соответственно;

$$\beta_1 : A_1^1 \to r_1, \quad (10)$$

где β_1 – оператор, характеризующий процедуру получения остатка r_1 от битовой последовательности A_1^1 в блоке B_1;

$$\beta_2 : A_2^1 \to r_2, \quad (11)$$

где β_2 – оператор, характеризующий процедуру получения остатка r_2 от битовой последовательности A_2^1 в блоке B_2.

Необходимо отметить, что отображения β_3 и β_4 имеют схожую интерпретацию для остатков r_3 и r_4.

$$\beta_3 : A_1^2 \to r_3, \quad (12)$$

где β_3 – оператор, характеризующий процедуру получения остатка r_3 от битовой последовательности A_1 в блоке B_3;

$$\beta_4 : A_2^2 \to r_4, \quad (13)$$

где β_4 – оператор, характеризующий процедуру получения остатка r_4 от битовой последовательности A_2^2 в блоке B_4;

$$\gamma : r_i \vee r_j \to p, p = \{0,1\}, i = \overline{1,2}, j = \overline{3,4}, \quad (14)$$

где γ – оператор, формализующий процедуру сравнения r_1, r_4 и r_2, r_3 и принятие решения о корректном или некорректном сдвиге.

Последним рассматриваемым преобразованием является операция подстановки. Она заключается в том, что входная последовательность из n-ого количества бит разделяется на два блока с указанным количеством бит, эти блоки определяют адрес ячейки из таблицы замен, в которой записано значение, которое и будет являться результатом подстановки. Но с целью повышения криптостойкости в данном преобразовании применяется также и операция циклического сдвига выходной последовательности. Обобщенная схема контроля правильности выполнения данного преобразования представлена на рисунке 8.

Описание функционирования данной схемы выглядит следующим образом. Поступающее число A_1, проходя через блок подстановки N, преобразуется в число A_2, которое и является результатом выполнения операции подстановки. Число A_1 разбивается на два числа A_1^1 и A_1^2, после чего они параллельно поступают в блок подстановки N и блок n. Блок n является таблицей остатков таблицы замены (блока N). Тем самым результат, полученный в ходе выполнения операции подстановки на выходе из блока n, будет представлен в остаточных классах. После этого над полученным результатом проводится операция циклического сдвига со сменой местами блоков двоичной записи результата.

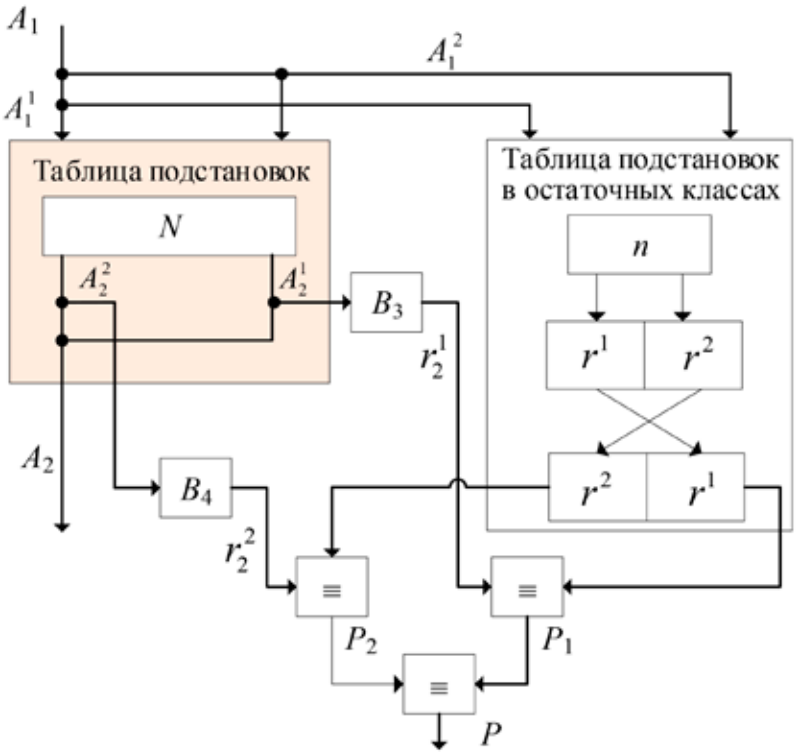

Рис. 8. Схема выполнения контроля операции подстановки

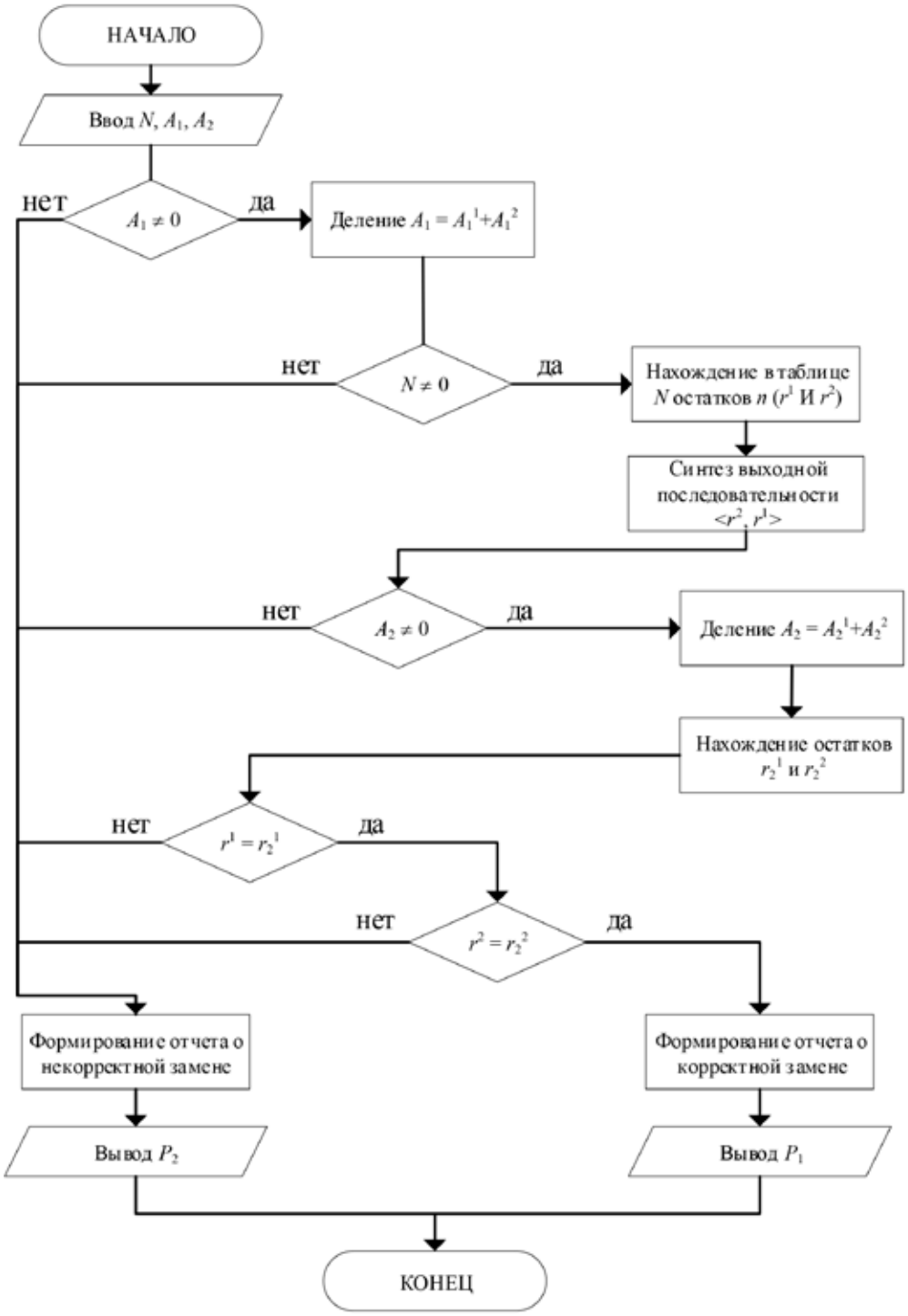

Рис. 9. Алгоритм выполнения контроля операции подстановки

Результатом прохождения чисел A_1^1 и A_1^2 через таблицу подстановок в остаточных классах n являются числа (остатки) r^1 и r^2 соответственно. После операции подстановки над блоками r^1 и r^2 выполняется операция циклического сдвига со сменой их местами. С целью контроля правильности выполнения преобразования подстановки число A_2 делят на блоки A_2^1 и A_2^2. Далее в блоках B_3 и B_4 вычисляются числа (остатки) r_2^1, r_2^2 битовых последовательностей A_2^1 и A_2^2, записанных в соответствующие блоки. Сравнение остатков r^1 и r^2, r_2^1 и r_2^2 в блоках сравнения (≡) с формированием

результатов P_1 и P_2 и P является результатом контроля правильности выполнения операции подстановки. В случае когда в результате сравнения получается, что $r^1 = r_2^1$ и $r^2 = r_2^2$, формируется отчет о правильности выполнения подстановки. В случае когда одно из равенств не выполняется либо не выполняются оба неравенства ($r^1 \neq r_2^1$, $r^2 \neq r_2^2$), следует вывод о неправильном выполнении контролируемого преобразования.

Алгоритм функционирования данной схемы контроля представлен на рисунке 9.

Работа представленного алгоритма может быть разбита на несколько этапов.

1. На первом этапе осуществляется ввод чисел A_1, A_2 и таблица замен N. Затем A_1 делится на A_1^1 и A_1^2. После этого находится таблица остатков n из таблицы замен N и находится выходная последовательность, представленная остатками r^1 и r^2.

2. Второй этап начинается с разделения числа A_2 на блоки A_2^1 и A_2^2 таким образом, чтобы количество бит в блоках было равно количеству бит в блоках A_1^1 и A_1^2 соответственно. Далее из блоков A_2^1 и A_2^2 формируются остатки r_2^1 и r_2^2.

3. На заключительном этапе остатки r^1 и r^2 сравниваются с r_2^1 и r_2^2, после чего, если равенство выполняется для обоих выражений, формируется отчет о правильности выполнения операции P_1, в противном случае формируется отчет о некорректном выполнении операции подстановки P_2.

С формальной точки зрения данные этапы могут быть представлены совокупностью отображений:

$$\alpha_1^* : A_1 \to \sum_{i=1}^{2} A_1^i, \qquad (15)$$

где α_1^* – оператор, характеризующий процедуру разбиения числа A_1 на битовые последовательности A_1^1 и A_1^2 соответственно;

$$\beta_1 : N \to n, \qquad (16)$$

где β_1 – оператор, характеризующий процедуру получения таблицы остатков n от таблицы замен N;

$$\beta_2 : A_1^1 \to r^1, \qquad (17)$$

где β_2 – оператор, характеризующий процедуру получения значения r^1 из таблицы остатков n от входной битовой последовательности A_1^1;

$$\beta_3 : A_1^2 \to r^2, \qquad (18)$$

где β_3 – оператор, характеризующий процедуру получения значения r^2 из таблицы остатков n от входной битовой последовательности A_1^2;

$$\alpha_2^* : A_2 \to \sum_{i=1}^{2} A_2^i, \qquad (19)$$

где α_2^* – оператор, характеризующий процедуру разбиения числа A_2 на битовые последовательности A_2^1 и A_2^2 соответственно;

$$\beta_4 : A_2^1 \to r_2^1, \qquad (20)$$

где β_4 – оператор, характеризующий процедуру получения значения r_2^1 из битовой последовательности A_2^1;

$$\beta_5 : A_2^1 \to r_2^2, \qquad (21)$$

где β_5 – оператор, характеризующий процедуру получения значения r_2^2 и битовой последовательности A_2^2;

$$\gamma : r^i \vee r^j \to p,$$

$$p = \{0,1\}, i = \overline{1,2}, j = \overline{3,4}, \qquad (22)$$

где γ – оператор, формализующий процедуру сравнения r^1 и r_2^1, r^2 и r_2^2, а также принятие решения о корректном или некорректном сдвиге.

Заключение

В ходе исследований были разработаны алгоритмы контроля основных операций алгоритма криптографического преобразования информации на основе модулярной арифметики и в соответствии с предложенными схемами контроля. Указанные алгоритмы лежат в основе формирования общего алгоритма технического диагностирования программируемых логических интегральных схем на автономных технических средствах.

Список литературы

1. Рыжов И.А., Харитонов А.С., Столяров А.В., Семенюк Д.Б. Оценка качества радиоканала передачи телеметрической информации // Инновационная наука: сборник статей по материалам XXII научно-практической конференции. 2019. Т. 4 (22). С. 65-73.

2. Cheng L., Huberman B.A. Auction-based Operation in LEO Satellite Systems for High-Efficiency Communications // IEEE Wireless Communications, 2020. Vol. 27, №. 2. P. 2-8.

3. Лоскутов А.И., Ряхова Е.А., Горбулин В.И. Концептуальная модель технического диагностирования бортовой аппаратуры автономных космических аппаратов на основе оптимальной реконфигурации в условиях априорной неопределенности появления неисправностей // Информационно-измерительные и управляющие системы. 2020. № 3. С. 43-55.

4. Охтилев М.Ю., Соколов Б.В., Юсупов Р.М. Теоретические и технологические основы концепции проактивного

мониторинга и управления сложными объектами // Известия ЮФУ. Технические науки. 2015. № 1(162). С. 162–174.

5. Лоскутов А.И., Клыков В.А. Идентификация и техническое диагностирование бортовой аппаратуры автономных космических аппаратов на основе биективного преобразования множества диагностических признаков // Контроль диагностика. 2016. № 4. С. 57-63.

6. Семенюк Д.Б., Лоскутов А.И., Бардаев Э.А., Клыков В.А. Системное моделирование при разработке математического обеспечения автоматизированных комплексов на основе полимодельного подхода с иерархическим принципом // Авиакосмическое приборостроение. М.: Научтехлитиздат, 2019. № 4. С. 33-44.

7. Столяров А.В., Никулин В.А., Лоскутов А.И., Клыков В.А., Ряхова Е.А. Методика построения математической модели процесса функционирования беспилотного авиационного комплекса с целью решения задачи технического диагностирования // Надежность и качество сложных систем. 2020. С. 45-67.

8. ГОСТ Р 34.13 – 2015. Информационная технология. Криптографическая защита информации. Режимы работы блочных шифров. М.: Стандартинформ, 2015.