РОССИЙСКАЯ ФЕДЕРАЦИЯ

(19) **RU** (11) **2 666 303**(13) **C1**

(51) МПК
G06F 17/10 (2006.01)
H04L 9/06 (2006.01)

ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(52) СПК
G06F 7/724 *(2006.01);* ***G06F 17/10*** *(2006.01);* ***H04L 9/0637*** *(2006.01);* ***H04L 9/0643*** *(2006.01);* ***H04L 9/3236*** *(2006.01)*

(21)(22) Заявка: **2017143805, 14.12.2017**

(24) Дата начала отсчета срока действия патента:
14.12.2017

Дата регистрации:
06.09.2018

Приоритет(ы):
(22) Дата подачи заявки: **14.12.2017**

(45) Опубликовано: **06.09.2018** Бюл. № **25**

Адрес для переписки:
127287, Москва, Старый Петровско-Разумовский пр-д, 1/23, стр. 1, Открытое акционерное общество "Информационные технологии и коммуникационные системы"

(72) Автор(ы):
Калистру Илья Иванович (RU)

(73) Патентообладатель(и):
Открытое акционерное общество "Информационные технологии и коммуникационные системы" (RU)

(56) Список документов, цитированных в отчете о поиске: **US 7970130 B2, 28.06.2011. US 2010/0115017 A1, 06.05.2010. US 2009/0080646 A1, 26.03.2009. US 2007/0081668 A1, 12.04.2007. RU 2598781 C1, 27.09.2016.**

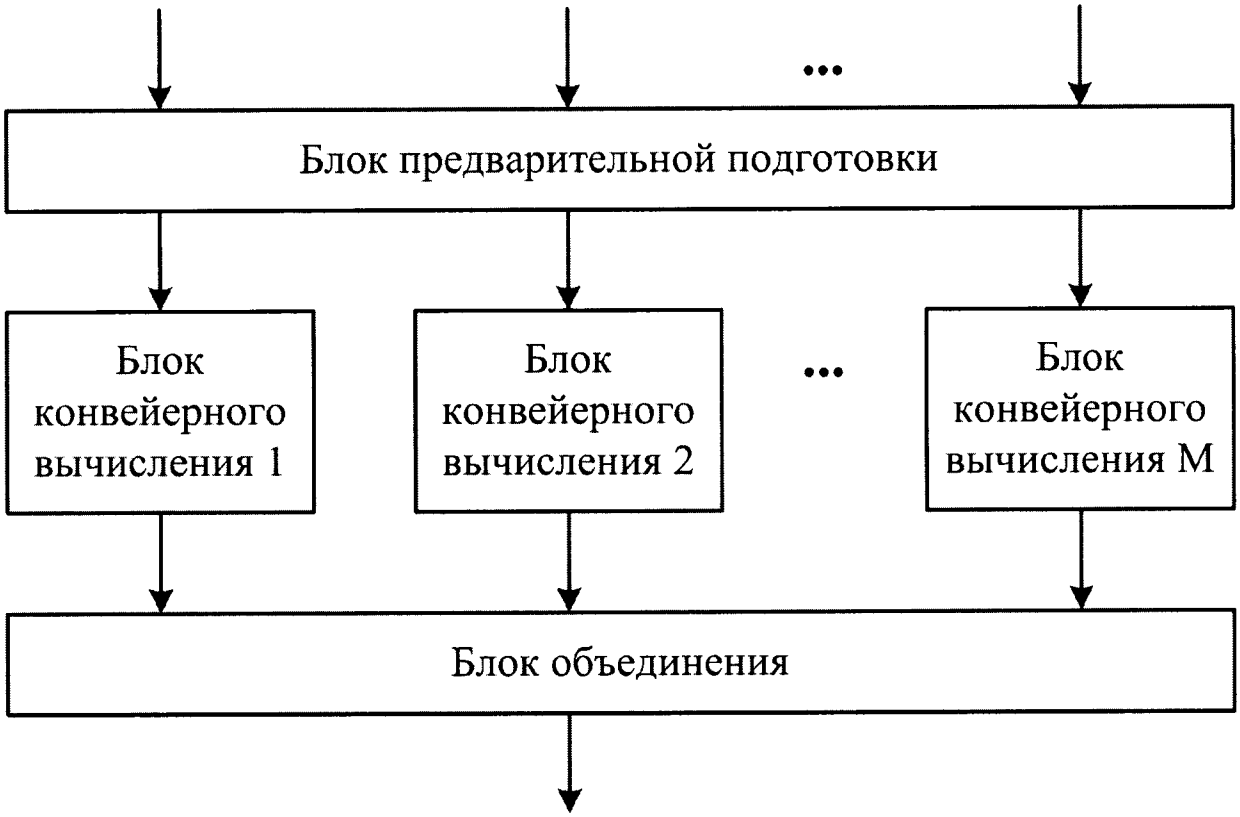
(54) Способ и устройство для вычисления хэш-функции

(57) Реферат:

Группа изобретений относится к вычислительной технике и может быть использована для вычисления хэш-функции. Техническим результатом является повышение быстродействия вычислений, расширение возможности выбора конфигурации устройства. Устройство содержит блок предварительной подготовки, имеющий M входов размерностью k бит, при этом M>1; M блоков конвейерного вычисления, работающих параллельно, каждый из которых содержит модуль памяти, модуль отключения обратной связи, сумматор, конвейерный перемножитель, имеющий L каскадов, блок обратной связи и блок накопления; блок объединения. 2 н. и 4 з.п. ф-лы, 2 ил.

RU 2666303 C1

RU 2666303 C1



Фиг. 1

RU 2666303 C1

RU 2666303 C1

RUSSIAN FEDERATION

FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(19) **RU** (11) **2 666 303**(13) **C1**

(51) Int. Cl.
G06F 17/10 (2006.01)
H04L 9/06 (2006.01)

(12) ABSTRACT OF INVENTION

(52) CPC
G06F 7/724 *(2006.01);* ***G06F 17/10*** *(2006.01);* ***H04L 9/0637*** *(2006.01);* ***H04L 9/0643*** *(2006.01);* ***H04L 9/3236*** *(2006.01)*

(21)(22) Application: **2017143805, 14.12.2017**

(24) Effective date for property rights:
14.12.2017

Registration date:
06.09.2018

Priority:
(22) Date of filing: **14.12.2017**

(45) Date of publication: **06.09.2018** Bull. № **25**

Mail address:
127287, Moskva, Staryj Petrovsko-Razumovskij pr-d, 1/23, str. 1, Otkrytoe aktsionernoe obshchestvo "Informatsionnye tekhnologii i kommunikatsionnye sistemy"

(72) Inventor(s):
Kalistru Ilya Ivanovich (RU)

(73) Proprietor(s):
Otkrytoe aktsionernoe obshchestvo "Informatsionnye tekhnologii i kommunikatsionnye sistemy" (RU)

RU 2666303 C1

(54) METHOD AND DEVICE FOR CALCULATING HASH FUNCTION

(57) Abstract:

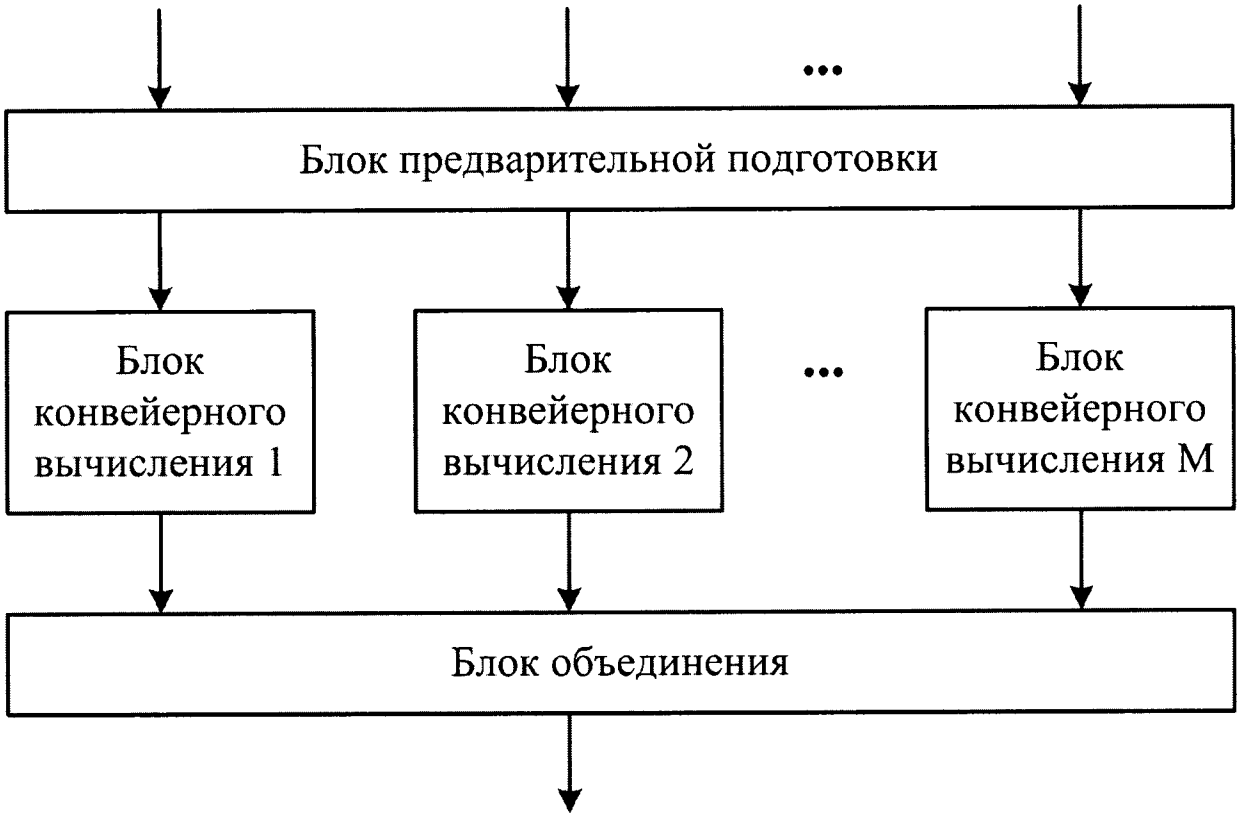
FIELD: computer equipment.

SUBSTANCE: group of inventions refers to computer technology and can be used to calculate a hash function. Device contains a preprocessing unit having M inputs of k bit dimension, with M>1; M of pipelining units operating in parallel, each of which contains a memory module, a feedback disconnect module, an adder, a pipeline multiplier having L stages, a feedback unit, and an accumulation unit; combining unit.

EFFECT: technical result is increased speed of calculations, increased choice of device configuration.

6 cl, 2 dwg

RU 2666303 C1



Фиг. 1

RU 2666303 C1

RU 2666303 C1

Область техники, к которой относится изобретение

Предполагаемое изобретение относится к вычислительной технике и, в частности, к устройствам и способам вычисления хэш-функции.

Уровень техники

В современных цифровых устройствах для обеспечения аутентичности информации используются различные ключевые хэш-функции. Одной из таких хэш-функций является функция GHASH, описанная в стандарте ISO/IEC 19772:2009 A.8.

Данная функция определяется как

$$\mathrm{GHASH}(S, H) = X_l,$$

где S - данные кадра данных,

H - полином хэш-функции, являющийся ключом данной хэш-функции,

l - количество блоков данных в S,

$$X_i = \begin{cases} 0, & \text{при } i = 0 \\ (X_{i-1} \oplus S_i) \cdot H, & \text{при } i \le l \end{cases}$$

Ускорение вычисления данной функции позволяет уменьшить время обработки каждого кадра данных и, как следствие, снижает задержку данных в устройстве обработки данных и увеличивает скорость его работы.

Так, известны способ и устройство вычисления хэш-функции для режима Galois Counter Mode (GCM; патент США №7970130, приоритет от 21.09.2007 г.), причем в способе для входных данных, включающих ассоциированные данные A, шифротекст C, полином H, вычисляют сначала промежуточное значение

$$X_A = A_1 H^{m+1} \oplus A_2 H^m \oplus \ldots \oplus (A_m^* \mid S^{k-v}) \cdot H,$$

затем промежуточное значение

$$X_C = C_1 H^{n+1} \oplus C_2 H^n \oplus \ldots \oplus (C_n^* \| S^{k-u}) H^2 \oplus (len(A) \| len(C)) H,$$

а также значения H^{n+1}, после чего вычисляется величина

$$X_A H^{n+1} \oplus X_C$$

являющаяся искомым значением функции GHASH.

Устройство, осуществляющее вычисления, состоит из первого, второго и третьего вычислительных модулей, вычисляющих значения X_A, X_C и H^{n+1} соответственно, а также четвертого вычислительного модуля, вычисляющего значение $X_A H^{n+1} \oplus X_C$. Это устройство позволяет вычислить значение GHASH (A, C, H) за max(m, n)+1 циклов, где m - количество блоков данных в A, а n - количество блоков данных в C, при этом $l = m + n + 1$ - это количество блоков, подаваемых на вход функции GHASH.

Известные способ и устройство приняты за прототипы.

Недостатками известных устройства и способа являются невысокое быстродействие и отсутствие возможности выбора конфигурации устройства перед его изготовлением. Так, наилучшего быстродействия в известном устройстве и способе удается добиться, если m=n. В этом случае удается получить ускорение в 2 раза по сравнению с последовательным вычислением.

Раскрытие изобретения

Техническим результатом является

1) повышение быстродействия вычислений,

2) расширение возможности выбора конфигурации устройства.

Для этого предлагается устройство для вычисления хэш-функции для кадра цифровых данных, причем кадр данных состоит из блоков данных длиной k бит, включающее

- блок предварительной подготовки, имеющий M входов размерностью k бит, при этом M>1;
- M блоков конвейерного вычисления, входы которых являются соответствующими выходами блока предварительной подготовки, и которые используют конвейерные перемножители, содержащие L каскадов;
- блока объединения, имеющего M входов, причем каждый вход подключен к соответствующему выходу блока конвейерного вычисления, а выход блока объединения является выходом устройства в целом;

при этом блок предварительной подготовки также имеет M буферных регистров FIFO и выполнен с возможностью

- записывать поступающие одновременно на M входов блоки данных в концы соответствующих буферных регистров FIFO;
- определять количество блоков данных, записанных в буферные регистры FIFO блока предварительной подготовки;
- определять наличие в буферных регистрах FIFO блока предварительной подготовки последнего блока данных кадра данных;
- считывать из соответствующих буферных регистров FIFO блоки данных во все M выходов блока предварительной подготовки при условии наличия в буферных регистрах FIFO блока предварительной подготовки L×M блоков данных или наличия в буферных регистрах FIFO блока предварительной подготовки последнего блока данных кадра данных;
- помечать выходы блока, как не имеющие данных, при условии отсутствия в буферных регистрах FIFO блока предварительной подготовки L×M блоков данных и отсутствия в буферных регистрах FIFO блока предварительной подготовки последнего блока данных кадра данных;
- нумеровать считываемые из буферных регистров FIFO блока предварительной подготовки блоки данных, причем
  - при наличии последнего блока данных в буферных регистрах FIFO блока предварительной подготовки, нумерация блоков данных осуществляется, начиная с конца кадра данных и производится, начиная с нуля,
  - при отсутствии последнего блока данных в буферных регистрах FIFO блока предварительной подготовки, каждому считываемому блоку присваивается номер L×M;

каждый из M блоков конвейерного вычисления, работающих параллельно, содержит

- модуль памяти,
- модуль отключения обратной связи,
- сумматор,
- конвейерный перемножитель, имеющий L каскадов,
- блок обратной связи,
- блок накопления;
- при этом модуль памяти выполнен с возможностью

- хранить записанные в него данные;
- выдавать на выход модуля памяти данные, хранящиеся в тех ячейках модуля памяти, номер которых равен номерам блоков данных, поступающих на вход модуля памяти;

сумматор содержит 1-й и 2-й входы и один выход и выполнен с возможностью суммировать в поле $GF(2^k)$ приходящие на 1-й и 2-й входы блоки данных и передавать результат на выход сумматора;

модуль отключения обратной связи содержит 1-й и 2-й входы, счетчик и выход и выполнен с возможностью

- увеличивать значение счетчика, при поступлении блока данных на 2-й вход
- передавать на выход блок данных с 1-го входа, если значение счетчика больше или равно M;
- передавать на выход блок данных, содержащий нули во всех битах, если значение счетчика меньше M;
- обнулять значение счетчика, если номер блока данных, поступившего на 2-й вход меньше M;

конвейерный перемножитель содержит

- 1-й и 2-й входы,
- L каскадов конвейера перемножения, подключенные друг за другом, работающие одновременно и каждый из которых выполняет свою часть вычислений произведения двух блоков, при этом входы первого каскада конвейера перемножения подключены к входам конвейерного перемножителя,
- один выход, который является выходом последнего каскада конвейера перемножения;

при этом конвейерный перемножитель выполнен с возможностью нумеровать выходные блоки так, что выходному блоку данных присваивается номер соответствующего блока данных, взятого с 1-го входа;

блок обратной связи содержит

- 1-й и 2-й входы и один выход,
- буферный регистр FIFO, в который записываются блоки данных, поступающие на 1-й вход блока обратной связи,

при этом блок обратной связи выполнен с возможностью считывать блоки данных из буферного регистра FIFO на выход лишь при поступлении блоков данных на 2-й вход;

блок накопления содержит ячейку памяти накопления и выполнен с возможностью

- получать на вход блоки данных и их номера;
- сравнивать номер входящего блока данных со значением L×M;
- если номер входящего блока данных больше или равен L×M, то обнулять ячейку памяти накопления;
- если номер входящего блока данных меньше L×M, то суммировать входящий блок данных в поле $GF(2^k)$ со значением, содержащемся в ячейке памяти накопления и записывать результат обратно в ячейку памяти накопления;

при этом вход каждого блока конвейерного вычисления, подключен к 1-му входу сумматора, ко входу модуля памяти, 2-му входу модуля отключения обратной связи и 2-му входу блока обратной связи;

при этом 1-й вход конвейерного перемножителя подключен к выходу модуля памяти,

а 2-й вход конвейерного перемножителя подключен к выходу сумматора;

при этом выход конвейерного перемножителя подключен ко входу блока накопления и к 1-му входу блока обратной связи, а выход блока обратной связи подключен к 1-му входу модуля отключения обратной связи;

выход модуля отключения обратной связи подключен ко 2-му входу сумматора;

выход блока накопления является выходом блока конвейерного вычисления;

блок объединения содержит M входов и выполнен с возможностью производить сложение блоков данных в поле $GF(2^k)$ от всех M входов и выдавать результат этого сложения на свой выход.

Также предлагается способ работы этого устройства, заключающийся в том, что

- определяют полином H хэш-функции;
- обнуляют содержимое всех буферных регистров FIFO блока предварительной подготовки;
- обнуляют ячейку памяти блоков накопления всех блоков конвейерного вычисления;
- вычисляют значения степеней полинома H, вычисленные в поле $GF(2^k)$;
- записывают вычисленные значения степеней полинома H в модули памяти всех блоков конвейерного вычисления, причем в ячейку памяти с номером i записывается H^{i+1}, причем $0 \le i < L \times M$, а в ячейку с номером $L \times M$ записывается H^{LM};
- записывают L блоков данных, содержащих нули во всех битах, в блоки обратной связи всех блоков конвейерного вычисления;
- обнуляют значение счетчиков в модулях отключения обратной связи всех блоков конвейерного вычисления;
- подают одновременно на M входов блока предварительной подготовки очередные M блоков данных кадра данных;
- если есть входящие блоки данных, записывают очередные блоки данных в соответствующие M буферные регистры FIFO в блоке предварительной подготовки;
- если в каком либо из буферных регистров FIFO есть последний блок данных кадра данных, то
  - считывают очередные блоки данных из каждого буферного регистра FIFO,
  - передают их на выход блока предварительной подготовки,
  - присваивают каждому блоку данных порядковый номер, начиная с нуля и считая с последнего блока данных кадра, находящегося в буферных регистрах FIFO;
- иначе, если очереди имеют длину L и не содержат последнего блока данных кадра данных, то
  - считывают очередные блоки данных из каждого буферного регистра FIFO,
  - передают их на выход блока предварительной подготовки,
  - присваивают каждому блоку данных номер $L \times M$;
- передают блоки данных с выходов блока предварительной подготовки на входы соответствующих блоков конвейерного вычисления;
- передают входящие блоки данных на вход модуля памяти, на 1-й вход сумматора, на 2-й вход модуля отключения обратной связи, на 2-й вход блока обратной связи в каждом блоке конвейерного вычисления;
- используют номер входящего блока данных как адрес модуля памяти в модуле памяти, извлекают заранее записанную в память модуля памяти значение степени H в

поле $GF(2^k)$;

- подают извлеченное значение на выход модуля памяти;
- выполняют в сумматоре следующие действия
  - суммируют в поле $GF(2^k)$ входящие с 1-го и 2-го входа блоки данных,
  - присваивают результату сложения номер блока данных со 1-го входа,
  - передают этот результат на выход сумматора;
- передают с выхода модуля памяти блок данных на 1-й вход конвейерного перемножителя;
- передают с выхода сумматора очередной блок данных на 2-й вход конвейерного перемножителя;
- вычисляют произведение блоков данных в поле $GF(2^k)$ в конвейерном перемножителе, передают его на выход конвейерного перемножителя, при этом присваивают результирующему блоку номер, равный номеру соответствующего входящего блока данных, взятого со 2-го входа конвейерного перемножителя;
- передают блок данных с выхода конвейерного перемножителя на вход блока накопления и на 1-й вход блока обратной связи;
- если имеется входящий блок данных на 1-м входе блока обратной связи, то записывают его в буферные регистры FIFO блока обратной связи;
- если имеется входящий блок данных на 2-м входе блока обратной связи, то
  - считывают из буферного регистра FIFO блока обратной связи очередной блок данных,
  - передают его на выход блока обратной связи;
- передают блок данных с выхода блока обратной связи на 1-й вход модуля отключения обратной связи;
- выполняют в модуле отключения обратной связи следующие действия
  - если на 2-й вход поступает блок данных, увеличивают значение счетчика;
  - если значение счетчика больше или равно M, передают на выход блок данных с 1-го входа;
  - если значение счетчика меньше M, передают на выход блок данных, содержащий нули во всех битах;
  - если номер блока данных, поступившего на 2-й вход меньше M, обнуляют значение счетчика;
- передают блоки данных с выхода модуля отключения обратной связи на 2-й вход сумматора;
- проверяют наличие данных на входе блока накопления,
  - если на входе имеется блок данных, то
    - если его номер меньше L×M, но больше или равен M, то
      - производят суммирование в поле $GF(2^k)$ входящего блока данных и содержимого ячейки памяти блока накопления,
      - записывают результат обратно в ячейку памяти накопления;
    - иначе, если номер входящего блока данных меньше M, то
      - производят суммирование в поле $GF(2^k)$ входящего блока данных и содержимого

ячейки памяти блока накопления,

- записывают результат в выход блока накопления,
- обнуляют содержимое ячейки памяти блока накопления;

- если на входе блока накопления отсутствует блок данных или если имеется блок данных, но его номер равен L×M, то не передают данные на выход блока накопления;
- передают блок данных с выхода блока накопления на выход блока конвейерного вычисления;
- передают блоки данных с выхода блоков конвейерного вычисления на входы блока объединения,
- проверяют наличие входящих блоков данных в блоке объединения
  - если на входе отсутствуют блоки данных, то помечают выход блока объединения, как не имеющий данных;
  - иначе, если на входе имеются блоки данных, то
    - суммируют в поле $GF(2^k)$ блоки данных со всех входов,
    - передают результат суммирования, являющийся значением хэш-функции, на выход блока объединения.

Следует обратить внимание на то, что выражение для $\mathrm{GHASH}(S,H)=X_l$ можно переписать в следующем виде

$$X_l = ((...(S_0 \cdot H \oplus S_1) \cdot H \oplus ...) \cdot H \oplus S_l) \cdot H$$

Учитывая линейность производимых операций, можно раскрыть скобки и собрать M групп членов, сгруппировав по j все члены содержащие S_{iM+j}, причем $0 \le j < M$. При этом M может быть любым положительным натуральным числом. Применив правило Горнера к каждой из групп отдельно, получим

$$\begin{aligned} X_l = {} & ((((...)H^M \oplus S_{l-2M})H^M \oplus S_{l-M})H^M \oplus S_l)H^1 \\ & \oplus ((((...)H^M \oplus S_{l-2M-1})H^M \oplus S_{l-M-1})H^M \oplus S_{l-1})H^2 \\ & \vdots \\ & \oplus ((((...)H^M \oplus S_{l-2M-(M-1)})H^M \oplus S_{l-M-(M-1)})H^M \oplus S_{l-(M-1)})H^M \end{aligned}$$

Если правильным образом сгруппировать входящие данные, то можно вычислять M слагаемых (частичных сумм) данного выражения независимо друг от друга, при этом умножать нужно не на H, а на H^M, а последний раз в каждом выражении нужно умножать на значения различных степеней H в зависимости от номера блока данных, который был прибавлен перед умножением - после прибавления последнего блока данных нужно умножать на H^1, после прибавления предпоследнего блока данных - H^2 и т.д.

Различные слагаемые данного выражения можно вычислять на различных параллельно работающих вычислителях, что дает прирост в производительности в M раз за счет увеличения количества устройств в M раз.

Также следует обратить внимание, что различные слагаемые данного выражения можно вычислять не только на различных вычислителях, но и на одном конвейерном вычислителе, в котором различные частичные суммы одновременно обрабатываются

в одном вычислителе, но при этом в каждый момент времени находятся на различных каскадах конвейера.

Использование M блоков конвейерной обработки, работающих параллельно, позволяет повысить быстродействие предлагаемого устройства в M раз.

Еще одним методом повышения быстродействия предлагаемого устройства является повышение тактовой частоты работы устройства в L раз за счет применения конвейерного перемножителя, содержащего L каскадов перемножения.

Поскольку сложение в поле Галуа является простейшим побитовым исключающим "или", требует немного ресурсов и быстро выполняется, то основным ограничением на повышение тактовой частоты является перемножитель. Таким образом, если разбить вычисление произведения на несколько каскадов конвейера, тем самым уменьшив длину критических путей распространения сигнала, то можно значительно повысить рабочую тактовую частоту устройства.

Разбиение процедуры перемножения на несколько частей не представляет большой сложности - при перемножении "в столбик" можно разбить суммирование результатов поразрядного перемножения на несколько частей, а при перемножении рекурсивным методом Карацубы, можно, например, завершать очередной каскад конвейера после выполнения сложений алгоритма для более мелких блоков алгоритма и перед началом сложения более крупных. Можно применять и другие разбиения и алгоритмы.

При удачном разбиении критические пути распространения электрических сигналов в схеме будут поделены на L примерно равных частей, и можно добиться повышения частоты в L раз. После разбиения каждому загруженному блоку данных понадобиться L тактов, чтобы пройти всю схему, однако это обстоятельство нивелируется тем фактом, что мы можем загружать в устройство одновременно L блоков данных.

Комбинация описанных приемов повышает быстродействие устройства в L×M раз по сравнению с последовательным вычислением. Таким образом, предложенное устройство вычисляет значение искомой функции за

$$\left\lceil \frac{l}{M} \right\rceil$$

тактов (округление вверх до ближайшего целого значения), каждый из который в L раз короче.

Предложенное устройство допускает возможность выбора целесообразной конфигурации, так, при проектировании устройства для конкретных условий применения, можно выбрать значения M и L, наиболее выгодные для заданных конкретных условий. Например, для достижения заданного быстродействия можно увеличивать количество L каскадов конвейерного перемножителя, увеличивая, тем самым, максимально возможную частоту работы устройства, до тех пор, пока другие технологические факторы не станут ограничивать дальнейший рост частоты. Затем можно увеличивать количество блоков конвейерного вычисления M до достижения требуемой производительности.

Частный случай выполнения устройства предусматривает, что на вход подаются блоки данных, дополнительно содержащие флаг K, причем для блоков данных, принадлежащих первому обрабатываемому кадру данных, флаг K имеет нулевое значение, а для блоков данных всех последующих кадров данных флаг K установлен по следующему правилу:

- если хэш-функция для следующего кадра данных должна быть рассчитана на том же значении полинома H, что и хэш-функция для предыдущего кадра данных, то

значение флага K для всех блоков данных следующего кадра данных установлено в то же значение, что и значение флага K для блоков данных предыдущего кадра данных;

- иначе, значение флага K для всех блоков данных следующего кадра данных установлено в значение, противоположное значению флага K для блоков данных предыдущего кадра данных;

при этом в устройстве

- имеется дополнительный независимый модуль памяти в каждом блоке конвейерного вычисления, подключенный параллельно имеющемуся модулю памяти;
- в каждом блоке конвейерного вычисления блоки данных поступают в модуль памяти, если флаг K равен нулю, или в дополнительный модуль памяти, если флаг K равен единице.

Наличие дополнительного модуля памяти позволяет в предложенном частном случае выполнения устройства обрабатывать различные кадры данных, используя различные значения полинома H хэш-функции без остановки работы устройства для загрузки нового полинома H хэш-функции в модули памяти блока конвейерного вычисления, поскольку, в то время как для обработки блоков данных кадра данных используется модуль памяти блока конвейерного вычисления, значения нового полинома H хэш-функции может быть загружено в дополнительный модуль памяти блока конвейерного вычисления и наоборот.

Это позволяет повысить быстродействие устройства в случае, когда от устройства требуется обработка кадров данных с использованием различных полиномов хэш-функции H.

Разработчик, проектирующий предложенное устройство, может принять решение о целесообразности введения дополнительного модуля памяти на основе данных об увеличении стоимости устройства из-за наличия дополнительного модуля памяти, а также из анализа необходимой частоты смены полинома H хэш-функции и анализа того, насколько простой устройства во время смены полинома H хэш-функции снижает производительность устройства.

При наличии дополнительных модулей памяти в каждом блоке конвейерного вычисления реализуется также частный случай способа, в котором, на этапе передачи входящих блоков данных на вход модуля памяти выполняют следующие действия:

- если входящий блок данных помечен флагом K, равным нулю, то
  - передают входящий блок данных на на вход модуля памяти
  - вычисляют значения степеней следующего полинома H в поле $GF(2^k)$ для следующего кадра данных
  - записывают вычисленные значения степеней полинома H для следующего кадра данных, в дополнительный модуль памяти блока конвейерного вычисления, причем в ячейку памяти с номером i записывается H^{i+1} причем $0 \le i < L \times M$, а в ячейку с номером $L \times M$ записывается H^{LM};
- если входящий блок данных помечен флагом K, равным единице, то
  - передают входящий блок данных на вход дополнительного модуля памяти,
  - вычисляют значения степеней следующего полинома H в поле $GF(2^k)$ для следующего кадра данных
  - записывают вычисленные значения степеней полинома H для следующего кадра данных, в модуль памяти блока конвейерного вычисления, причем в ячейку памяти с

номером i записывают H^{i+1}, причем $0 \le i < L \times M$, а в ячейку с номером $L \times M$ записывают H^{LM}.

Описанный частный случай выполнения способа позволяет повысить быстродействие устройства в случае, когда от устройства требуется обработка кадров данных с использованием различных полиномов хэш-функции H. Блоки данных кадров данных, поступают на вход устройства и содержат флаг K, заранее установленный в соответствии с учетом того, для какого полинома хэш-функции H требуется обработать данный кадр. В то время, когда кадры данных обрабатываются с одним полиномом хэш-функции H, используется один из модулей памяти, а в другой модуль памяти загружают значения степеней полинома H для следующего кадра данных. Это позволяет не останавливать устройство для смены полинома хэш-функции H.

Еще в одном частном случае выполнения устройства выход блока конвейерного перемножителя подключается напрямую к 1-му входу модуля отключения обратной связи, минуя блок обратной связи.

Такой частный случай выполнения устройства целесообразно применять, если заранее известно, что блоки данных кадров данных поступают на предлагаемое устройство на каждый такт тактовой частоты без промежутков. В этом случае также возникает возможность отключать питание от неиспользуемого блока обратной связи, что позволяет снизить энергопотребление устройства.

Если уже на этапе проектирования устройства известно, что блоки данных кадров данных всегда поступают на предлагаемое устройство на каждый такт тактовой частоты без промежутков, то блок обратной связи можно исключить из состава устройства, что экономит аппаратные ресурсы и снижает стоимость устройства.

Для реализации такого частного случая выполнения устройства также предлагается частный случай способа, согласно которому передают блок данных с выхода конвейерного перемножителя на вход блока накопления и на 1-й вход модуля отключения обратной связи вместо 1-го входа блока обратной связи.

Передача блока данных с выхода конвейерного перемножителя на вход блока накопления и на 1-й вход модуля отключения обратной связи вместо 1-го входа блока обратной связи позволяет снизить количество выполняемых операций, что также может снизить энергопотребление и повысить быстродействие устройства.

Краткое описание чертежей

На фиг. 1 показана блок-схема устройства.

На фиг. 2 показана блок-схема блока конвейерного вычисления.

Осуществление изобретения

Для изготовления предложенного устройства необходимо определить исходные данные: количество бит в каждом блоке данных k, количество каскадов в конвейерных перемножителях L, количество блоков конвейерного вычисления M.

Количество бит в каждом блоке данных k выбирают исходя из размера блока данных в используемом для аутентификации алгоритме.

Количество каскадов в конвейерных перемножителях L выбирают достаточным для того, что бы максимальная частота работы конвейерного перемножителя достигла максимальной тактовой частоты F_{max} для других компонентов схемы.

Количество блоков конвейерного вычисления M выбирают достаточным, для достижения требуемого быстродействия, исходя из следующей формулы

$P=F_{max}kM$, бит/с

Если M-1 блоков конвейерного вычисления недостаточно для достижения требуемого

быстродействия, а M блоков конвейерного вычисления обеспечивают быстродействие, превышающее требуемое, то для снижения энергопотребления выбирают частоту работу устройства по формуле

$$F=P/(k\cdot M)$$

Далее проектируют устройство согласно описанию, содержащим все составные элементы, с учетом связей между ними и согласно их назначению.

Предпочтительно выполнять предлагаемое устройство в виде блока, входящего в состав вычислительной системы, выполняющей функции обеспечения аутентичности данных, подготавливающей данные для обработки предлагаемым устройством и использующей вычисленные предлагаемым устройством значения хэш-функции GHASH для целей подтверждения их аутентичности.

Например, предлагаемое устройство может быть выполнено в виде блока интегральной схемы специального назначения, выполняющей функции обеспечения аутентичности данных и содержащей интерфейс для получения и отправки данных, блок подготовки данных для работы предлагаемого устройства, предлагаемое устройство для вычисления хэш-функции, и блок, использующий вычисленную хэш-функцию для обеспечения аутентичности данных. Предлагаемое устройство может также входить в состав других вычислительных систем, в которых требуется вычислять указанную хэш-функцию.

Реализовать предложенное устройство в виде блока интегральной схемы или в виде блока устройства для обеспечения аутентичности данных, выполненного на базе программируемой логической интегральной схемы (ПЛИС) или базового матричного кристалла может специалист в области проектирования цифровых интегральных схем.

Для реализации предложенного способа изготавливают устройство согласно описанию. Затем определяют битовое представление значения полинома H, являющегося ключом хэш-функции и значение которого должно храниться в секрете. Например, 128 бит битового представления значения полинома H может быть получено из генератора случайных чисел. В зависимости, от применяемого алгоритма, значение H может определяться и другими способами.

После этого обнуляют содержимое всех буферных регистров FIFO блока предварительной подготовки, вычисляют в поле $GF(2^k)$ значения степеней полинома H вплоть до H^{LM}, записывают их битовые представления в модули памяти всех блоков конвейерного вычисления, причем в ячейку памяти с номером i записывается H^{i+1} причем $0\le i<L\times M$, а в ячейку с номером $L\times M$ записывают H^{LM}, записывают L блоков данных, содержащих нули во всех битах, в блоки обратной связи всех блоков конвейерного вычисления, обнуляют значение счетчиков в модулях отключения обратной связи всех блоков конвейерного вычисления.

После этого приступают к обработке кадров данных. Для этого разделяют кадры данных на блоки по k бит, причем последний блок снабжен метаинформацией о том, что это последний блок кадра данных. Подают одновременно по M очередных блоков данных на M входов блока предварительной подготовки. В блоке предварительной подготовки нумеруют блоки данных (снабжают метаинформацией). Затем передают блоки данных с выходов блока предварительной подготовки на входы соответствующих M блоков конвейерного вычисления. В каждом блоке конвейерного вычисления, используя хранящиеся в модулях памяти значения степеней H, вычисляют L частичных сумм и суммируют их в блоке накопления блока конвейерного вычисления.

После этого блоки данных с выходов блоков конвейерного вычисления передают

на M входов блока объединения, суммируют в блоке объединения блоки данных, поступивших на его входы, и передают результат, являющийся искомым значением, на выход устройства.

При реализации частного случая выполнения устройства, предусматривающего наличие дополнительного независимого модуля памяти в каждом блоке конвейерного вычисления, подключенный параллельно имеющемуся модулю памяти, возникает возможность ускорить обработку данных за счет исключения простоев устройства во время загрузки значений степеней полинома H во время смены ключа хэш-функции.

Наличие двух модулей памяти в каждом блоке конвейерного вычисления позволяет загружать значения степеней H для нового ключа в один из модулей памяти, без остановки работы устройства, пока происходит обработка данных с использованием степеней H для старого ключа, хранящихся в другом модуле памяти. При этом на вход устройства подаются блоки данных, дополнительно содержащие флаг K, причем для блоков данных, принадлежащих первому обрабатываемому кадру данных, флаг K имеет нулевое значение, а для блоков данных всех последующих кадров данных флаг K установлен по следующему правилу:

- если хэш-функция для следующего кадра данных должна быть рассчитана на том же значении полинома H, что и хэш-функция для предыдущего кадра данных, то значение флага K для всех блоков данных следующего кадра данных установлено в то же значение, что и значение флага K для блоков данных предыдущего кадра данных;
- иначе, значение флага K для всех блоков данных следующего кадра данных установлено в значение, противоположное значению флага K для блоков данных предыдущего кадра данных;

при этом в устройстве в каждом блоке конвейерного вычисления блоки данных поступают в модуль памяти, если флаг K равен нулю, или в дополнительный модуль памяти, если флаг K равен единице.

Для реализации частного случая выполнения устройства блоки данных текущего кадра данных дополнительно снабжают флагом K, при этом значение флага K заранее устанавливают в соответствии с описанным выше правилом. Дополнительно, в каждом блоке конвейерного вычисления, на этапе передачи входящих блоков данных на вход модуля памяти выполняют следующие действия:

- если входящий блок данных помечен флагом K, равным нулю, то
  - передают входящий блок данных на вход модуля памяти,
  - вычисляют значения степеней следующего полинома H в поле $GF(2^k)$ для следующего кадра данных
  - записывают вычисленные значения степеней полинома H для следующего кадра данных, в дополнительный модуль памяти блока конвейерного вычисления, причем в ячейку памяти с номером i записывается H^{i+1} причем $0 \le i < L \times M$, а в ячейку с номером $L \times M$ записывается H^{LM}.
- если входящий блок данных помечен флагом K, равным единице, то
  - передают входящий блок данных на вход дополнительного модуля памяти,
  - вычисляют значения степеней следующего полинома H в поле $GF(2^k)$ для следующего кадра данных
  - записывают вычисленные значения степеней полинома H для следующего кадра данных, в модуль памяти блока конвейерного вычисления, причем в ячейку памяти с

номером i записывается H^{i+1}, причем $0 \le i < L \times M$, а в ячейку с номером $L \times M$ записывается H^{LM};

Возможен еще один частного случая выполнения устройства, согласно которому в устройстве выход конвейерного перемножителя имеет возможность подключаться напрямую к 1-му входу модуля отключения обратной связи, минуя блок обратной связи. При этом устройство может корректно работать, только если блоки данных кадра данных подаются на вход устройства на каждый такт без промежутков до конца каждого кадра.

Данный частный случай возможно осуществить, например, с использованием электронных ключей, отключающих выход конвейерного перемножителя от блока обратной связи и подключающих к 1-му входу модуля отключения обратной связи. При этом возникает возможность для экономии электроэнергии отключить питание блока обратной связи, когда выход конвейерного перемножителя подключен напрямую к 1-му входу модуля отключения обратной связи.

Если же заранее известно, что блоки данных кадра данных подаются на вход устройства на каждый такт без промежутков до конца каждого кадра, то возможно осуществление данного частного случая реализации устройства вообще без изготовления блока обратной связи. В этом случае выход конвейерного перемножителя подключают напрямую к 1-му входу модуля отключения обратной связи. Такое исполнение позволяет экономить электроэнергию и аппаратные ресурсы, например, место на кристалле микросхемы, на которой реализовано устройство.

Для осуществления способа в этом случае на вход устройства подают по M блоков данных на каждый такт до окончания текущего кадра данных. Дополнительно, в каждом блоке конвейерного вычисления передают блок данных с выхода конвейерного перемножителя на вход блока накопления и на 1-й вход модуля отключения обратной связи вместо 1-го входа блока обратной связи.

(57) Формула изобретения

1. Устройство для вычисления хэш-функции для кадра цифровых данных, причем кадр данных состоит из блоков данных длиной k бит, включающее

блок предварительной подготовки, имеющий M входов размерностью k бит, при этом $M>1$;

M блоков конвейерного вычисления, входы которых являются соответствующими выходами блока предварительной подготовки, и которые используют конвейерные перемножители, содержащие L каскадов;

блока объединения, имеющего M входов, причем каждый вход подключен к соответствующему выходу блока конвейерного вычисления, а выход блока объединения является выходом устройства в целом;

при этом блок предварительной подготовки также имеет M буферных регистров FIFO и выполнен с возможностью

записывать поступающие одновременно на M входов блоки данных в концы соответствующих буферных регистров FIFO;

определять количество блоков данных, записанных в буферные регистры FIFO блока предварительной подготовки;

определять наличие в буферных регистрах FIFO блока предварительной подготовки последнего блока данных кадра данных;

считывать из соответствующих буферных регистров FIFO блоки данных во все M выходов блока предварительной подготовки при условии наличия в буферных регистрах

FIFO блока предварительной подготовки L×M блоков данных или наличия в буферных регистрах FIFO блока предварительной подготовки последнего блока данных кадра данных;

помечать выходы блока, как не имеющие данных, при условии отсутствия в буферных регистрах FIFO блока предварительной подготовки L×M блоков данных и отсутствия в буферных регистрах FIFO блока предварительной подготовки последнего блока данных кадра данных;

нумеровать считываемые из буферных регистров FIFO блока предварительной подготовки блоки данных, причем

при наличии последнего блока данных в буферных регистрах FIFO блока предварительной подготовки, нумерация блоков данных осуществляется, начиная с конца кадра данных, и производится, начиная с нуля,

при отсутствии последнего блока данных в буферных регистрах FIFO блока предварительной подготовки, каждому считываемому блоку присваивается номер L×M;

каждый из M блоков конвейерного вычисления, работающих параллельно, содержит

модуль памяти,

модуль отключения обратной связи,

сумматор,

конвейерный перемножитель, имеющий L каскадов,

блок обратной связи,

блок накопления;

при этом модуль памяти выполнен с возможностью

хранить записанные в него данные;

выдавать на выход модуля памяти данные, хранящиеся в тех ячейках модуля памяти, номер которых равен номерам блоков данных, поступающих на вход модуля памяти;

сумматор содержит 1-й и 2-й входы и один выход и выполнен с возможностью суммировать в поле $GF(2^k)$ приходящие на 1-й и 2-й входы блоки данных и передавать результат на выход сумматора;

модуль отключения обратной связи содержит 1-й и 2-й входы, счетчик и выход и выполнен с возможностью

увеличивать значение счетчика, при поступлении блока данных на 2-й вход;

передавать на выход блок данных с 1-го входа, если значение счетчика больше или равно M;

передавать на выход блок данных, содержащий нули во всех битах, если значение счетчика меньше M;

обнулять значение счетчика, если номер блока данных, поступившего на 2-й вход, меньше M;

конвейерный перемножитель содержит

1-й и 2-й входы,

L каскадов конвейера перемножения, подключенные друг за другом, работающие одновременно, и каждый из которых выполняет свою часть вычислений произведения двух блоков, при этом входы первого каскада конвейера перемножения подключены к входам конвейерного перемножителя,

один выход, который является выходом последнего каскада конвейера перемножения;

при этом конвейерный перемножитель выполнен с возможностью нумеровать выходные блоки так, что выходному блоку данных присваивается номер соответствующего блока данных, взятого с 1-го входа;

блок обратной связи содержит

1-й и 2-й входы и один выход,

буферный регистр FIFO, в который записываются блоки данных, поступающие на 1-й вход блока обратной связи;

при этом блок обратной связи выполнен с возможностью считывать блоки данных из буферного регистра FIFO на выход лишь при поступлении блоков данных на 2-й вход;

блок накопления содержит ячейку памяти накопления и выполнен с возможностью получать на вход блоки данных и их номера;

сравнивать номер входящего блока данных со значением L×M;

если номер входящего блока данных больше или равен L×M, то обнулять ячейку памяти накопления;

если номер входящего блока данных меньше L×M, то суммировать входящий блок данных в поле GF (2^k) со значением, содержащемся в ячейке памяти накопления, и записывать результат обратно в ячейку памяти накопления;

при этом вход каждого блока конвейерного вычисления, подключен к 1-му входу сумматора, к входу модуля памяти, 2-му входу модуля отключения обратной связи и 2-му входу блока обратной связи;

при этом 1-й вход конвейерного перемножителя подключен к выходу модуля памяти, а 2-й вход конвейерного перемножителя подключен к выходу сумматора;

при этом выход конвейерного перемножителя подключен ко входу блока накопления и к 1-му входу блока обратной связи, а выход блока обратной связи подключен к 1-му входу модуля отключения обратной связи;

выход модуля отключения обратной связи подключен ко 2-му входу сумматора;

выход блока накопления является выходом блока конвейерного вычисления;

блок объединения содержит M входов и выполнен с возможностью производить сложение блоков данных в поле GF (2^k) от всех M входов и выдавать результат этого сложения на свой выход.

2. Устройство, по п. 1, на вход которого подаются блоки данных, дополнительно содержащие флаг K, причем для блоков данных, принадлежащих первому обрабатываемому кадру данных, флаг K имеет нулевое значение, а для блоков данных всех последующих кадров данных флаг K установлен по следующему правилу:

если хэш-функция для следующего кадра данных должна быть рассчитана на том же значении полинома H, что и хэш-функция для предыдущего кадра данных, то значение флага K для всех блоков данных следующего кадра данных установлено в то же значение, что и значение флага K для блоков данных предыдущего кадра данных;

иначе, значение флага K для всех блоков данных следующего кадра данных установлено в значение, противоположное значению флага K для блоков данных предыдущего кадра данных;

при этом в устройстве

имеется дополнительный независимый модуль памяти в каждом блоке конвейерного вычисления, подключенный параллельно имеющемуся модулю памяти;

в каждом блоке конвейерного вычисления блоки данных поступают в модуль памяти, если флаг K равен нулю, или в дополнительный модуль памяти, если флаг K равен единице.

3. Устройство по п. 1 или 2, в котором выход блока конвейерного перемножителя имеет возможность подключаться напрямую к 1-му входу модуля отключения обратной связи, минуя блок обратной связи.

4. Способ вычисления хэш-функции для кадра цифровых данных, причем кадр данных состоит из блока данных длиной k бит, заключающийся в том, что

определяют полином H хэш-функции;

обнуляют содержимое всех буферных регистров FIFO блока предварительной подготовки;

обнуляют ячейку памяти блоков накопления всех блоков конвейерного вычисления;

вычисляют значения степеней полинома H, вычисленные в поле GF (2^k);

записывают вычисленные значения степеней полинома H в модули памяти всех блоков конвейерного вычисления, причем в ячейку памяти с номером i записывается H^{i+1}, причем $0 \le i < L \times M$, а в ячейку с номером $L \times M$ записывается H^{LM};

записывают L блоков данных, содержащих нули во всех битах, в блоки обратной связи всех блоков конвейерного вычисления;

обнуляют значение счетчиков в модулях отключения обратной связи всех блоков конвейерного вычисления;

подают одновременно на M входов блока предварительной подготовки очередные M блоков данных кадра данных;

если есть входящие блоки данных, записывают очередные блоки данных в соответствующие M буферные регистры FIFO в блоке предварительной подготовки;

если в каком либо из буферных регистров FIFO есть последний блок данных кадра данных, то

считывают очередные блоки данных из каждого буферного регистра FIFO;

передают их на выход блока предварительной подготовки;

присваивают каждому блоку данных порядковый номер, начиная с нуля и считая с последнего блока данных кадра, находящегося в буферных регистрах FIFO;

иначе, если очереди имеют длину L и не содержат последнего блока данных кадра данных, то

считывают очередные блоки данных из каждого буферного регистра FIFO;

передают их на выход блока предварительной подготовки;

присваивают каждому блоку данных номер $L \times M$;

передают блоки данных с выходов блока предварительной подготовки на входы соответствующих блоков конвейерного вычисления;

передают входящие блоки данных на вход модуля памяти, на 1-й вход сумматора, на 2-й вход модуля отключения обратной связи, на 2-й вход блока обратной связи в каждом блоке конвейерного вычисления;

используют номер входящего блока данных как адрес модуля памяти в модуле памяти, извлекают заранее записанную в память модуля памяти значение степени H в поле GF (2^k);

подают извлеченное значение на выход модуля памяти;

выполняют в сумматоре следующие действия

суммируют в поле GF(2^k) входящие с 1-го и 2-го входа блоки данных;

присваивают результату сложения номер блока данных со 1-го входа;

передают этот результат на выход сумматора;

передают с выхода модуля памяти блок данных на 1-й вход конвейерного перемножителя;

передают с выхода сумматора очередной блок данных на 2-й вход конвейерного перемножителя;

вычисляют произведение блоков данных в поле GF (2^k) в конвейерном перемножителе,

передают его на выход конвейерного перемножителя, при этом присваивают результирующему блоку номер, равный номеру соответствующего входящего блока данных, взятого со 2-го входа конвейерного перемножителя;

передают блок данных с выхода конвейерного перемножителя на вход блока накопления и на 1-й вход блока обратной связи;

если имеется входящий блок данных на 1-м входе блока обратной связи, то записывают его в буферные регистры FIFO блока обратной связи;

если имеется входящий блок данных на 2-м входе блока обратной связи, то

считывают из буферного регистра FIFO блока обратной связи очередной блок данных;

передают его на выход блока обратной связи;

передают блок данных с выхода блока обратной связи на 1-й вход модуля отключения обратной связи;

выполняют в модуле отключения обратной связи следующие действия

если на 2-й вход поступает блок данных, увеличивают значение счетчика;

если значение счетчика больше или равно M, передают на выход блок данных с 1-го входа;

если значение счетчика меньше M, передают на выход блок данных, содержащий нули во всех битах;

если номер блока данных, поступившего на 2-й вход меньше M, обнуляют значение счетчика;

передают блоки данных с выхода модуля отключения обратной связи на 2-й вход сумматора;

проверяют наличие данных на входе блока накопления,

если на входе имеется блок данных, то

если его номер меньше L×M, но больше или равен M, то

производят суммирование в поле GF (2^k) входящего блока данных и содержимого ячейки памяти блока накопления;

записывают результат обратно в ячейку памяти накопления;

иначе, если номер входящего блока данных меньше M, то

производят суммирование в поле GF (2^k) входящего блока данных и содержимого ячейки памяти блока накопления;

записывают результат в выход блока накопления;

обнуляют содержимое ячейки памяти блока накопления;

если на входе блока накопления отсутствует блок данных или если имеется блок данных, но его номер равен L×M, то не передают данные на выход блока накопления;

передают блок данных с выхода блока накопления на выход блока конвейерного вычисления;

передают блоки данных с выхода блоков конвейерного вычисления на входы блока объединения;

проверяют наличие входящих блоков данных в блоке объединения

если на входе отсутствуют блоки данных, то помечают выход блока объединения, как не имеющий данных;

иначе, если на входе имеются блоки данных, то

суммируют в поле GF (2^k) блоки данных со всех входов;

передают результат суммирования, являющийся значением хэш-функции, на выход блока объединения.

5. Способ по п. 4 в котором, при наличии дополнительных модулей памяти в каждом

блоке конвейерного вычисления, на этапе передачи входящих блоков данных на вход модуля памяти выполняют следующие действия:

если входящий блок данных помечен флагом K, равным нулю, то

передают входящий блок данных на вход модуля памяти;

вычисляют значения степеней следующего полинома H в поле GF (2^k) для следующего кадра данных;

записывают вычисленные значения степеней полинома H для следующего кадра данных, в дополнительный модуль памяти блока конвейерного вычисления, причем в ячейку памяти с номером i записывается H^{i+1}, причем $0 \le i < L \times M$, а в ячейку с номером $L \times M$ записывается H^{LM};

если входящий блок данных помечен флагом K, равным единице, то

передают входящий блок данных на вход дополнительного модуля памяти;

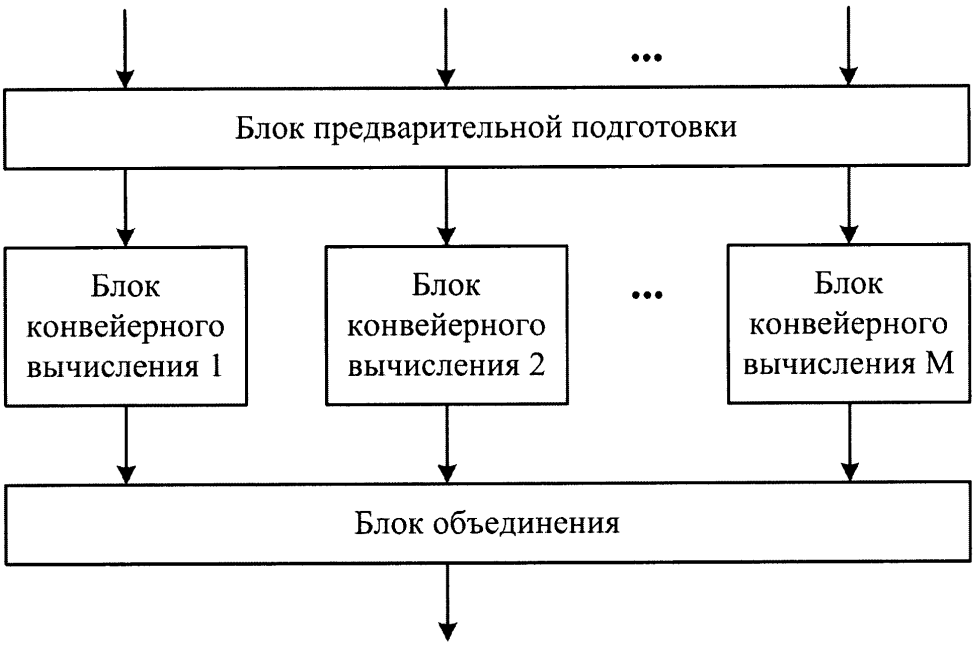
вычисляют значения степеней следующего полинома H в поле GF (2^k) для следующего кадра данных;

записывают вычисленные значения степеней полинома H для следующего кадра данных, в модуль памяти блока конвейерного вычисления, причем в ячейку памяти с номером i записывается H^{i+1}, причем $0 \le i < L \times M$, а в ячейку с номером $L \times M$ записывается H^{LM}.

6. Способ по п. 4 или п. 5, в котором передают блок данных с выхода конвейерного перемножителя на вход блока накопления и на 1-й вход модуля отключения обратной связи вместо 1-го входа блока обратной связи.

1

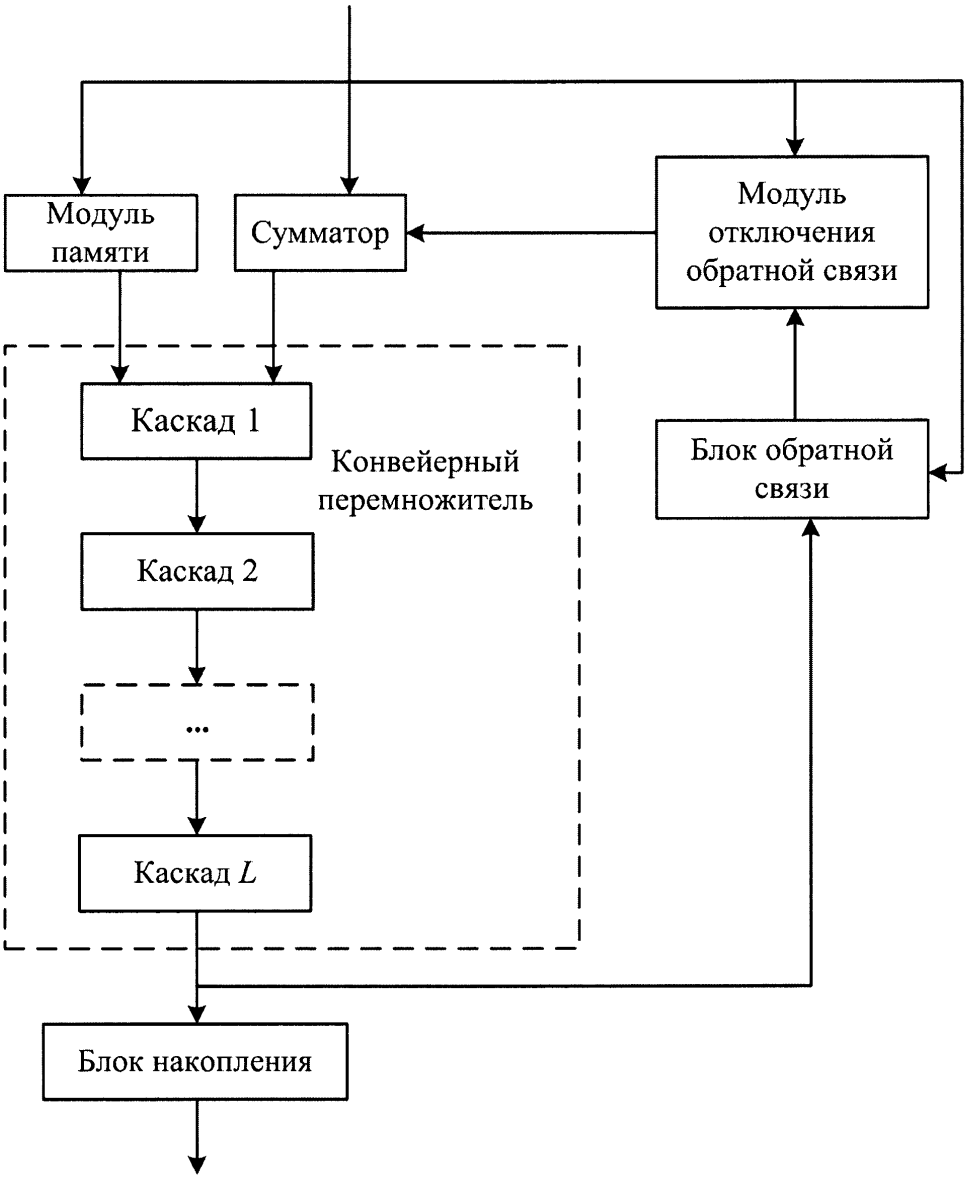
Способ и устройство для вычисления хэш-функции



Фиг. 1

2

Способ и устройство для вычисления хэш-функции



Фиг. 2