**SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions.**

**SHA-3 Стандарт: Хеш-функции на основе перестановок и функции расширенного вывода (XOFs)**

**Аннотация**

Настоящий стандарт определяет семейство функций алгоритма безопасного хеширования (**Secure Hash Algorithm-3, SHA-3**) для двоичных данных. Каждая из функций SHA-3 основана на требованиях, установленных алгоритмом ***KECCAK***, который Национальным институтом стандартов и технологий США (**NIST – National Institute of Standards and Technology**) был выбран в качестве победителя конкурса **SHA-3 Cryptographic Hash Algorithm**. Это стандарт также определяет семейство математических перестановок ***KECCAK-p***, включающих перестановку, лежащую в основе *KECCAK*, для облегчения разработки дополнительных криптографических функций на основе перестановок.

Семейство SHA-3 состоит за четырёх криптографических хеш-функций: SHA3-224, SHA3-256, SHA3-384 и SHA3-512; и двух функций расширенного вывода (**XOFs**): SHAKE128 и SHAKE256.

Хеш-функции являются компонентами многих важных приложений информационной безопасности, в частности: 1) генерация и верификация цифровых подписей; 2) формирование ключа; 3) генерация псевдослучайных битов. Хеш-функции, определенные в настоящем стандарте, дополняют хеш-функцию SHA-1 и семейство хеш-функций SHA-2, определенных в стандарте **FIPS (Federal Information Processing Standards) 180-4, the Secure Hash Standard**.

Функции расширенного вывода отличаются от хеш-функций, но их можно использовать аналогичным образом и гибко адаптировать непосредственно к требованиям конкретных приложений с учетом дополнительных соображений безопасности.

**1. Наименование стандарта:** Стандарт SHA-3: Хеш-функции, основанные на перестановках, и функции расширенного вывода (**FIPSPUB 202**).

**2. Категория стандарта:** Стандарт компьютерной безопасности, криптография.

**3. Объяснение:** Настоящий стандарт (FIPS 202) определяет семейство функций алгоритма криптографического хеширования (SHA-3) для двоичных данных. Каждая из функций SHA-3 базируется на принципах алгоритма *KECCAK*, который NIST выбрало в качестве победителя конкурса SHA-3 Cryptographic Hash Algorithm Competition. Это стандарт также определяет семейство математических перестановок *KECCAK-p*, включающих перестановку, лежащую в основе *KECCAK*. Указанные перестановки могут служить основными компонентами дополнительных криптографических функций, которые могут быть определены в будущем.

Семейство SHA-3 состоит из четырёх криптографических хеш-функций и двух функций расширенного вывода (XOFs). Криптографические хеш-функции: SHA3-224, SHA3-256, SHA3-384 и SHA3-512; функции расширенного вывода: SHAKE128 и SHAKE256.

Данные на входе хеш-функций называются **сообщением** (**message**), a на выходе – **дайджестом** (**digest**) или «хешем» (**hash value**). Сообщение может иметь различную длину, длина дайджеста **фиксирована**. Криптографическая хеш-функция – это хеш-функция, которая обладает специальными свойствами, включающими в себя сопротивление коллизиям (**collision resistance**) и сопротивление прообразу (**preimage resistance**), которые важны для многих приложений в области информационной безопасности. К примеру, криптографическая хеш-функция увеличивает безопасность и эффективность схемы цифровой подписи в том случае, когда вместо сообщения цифровой подписью подписывается дайджест. В этом контексте сопротивление хеш-функции коллизиям обеспечивает уверенность в том, что оригинальное сообщение не могло быть изменено на другое сообщение с тем же значением хеш-функции и, следовательно, с той же подписью. Другое приложение криптографических хеш-функций включает в себя генерацию псевдослучайных чисел (**pseudorandom bit generation**), имитовставки (**message authentication codes**) и функции формирования ключа (**key derivation functions**).

Четыре хеш-функции SHA-3, определенные в настоящем стандарте, дополняют хеш-функции, которые определены в **FIPS 180-4**: семейство SHA-1 и SHA-2. Оба стандарта вместе обеспечивают устойчивость к будущим достижениям в области криптоанализа хеш-функций, поскольку они основаны на принципиально разных принципах проектирования. Помимо разнообразия дизайна, хеш-функции в данном стандарте обеспечивают некоторые дополнительную реализацию и характеристики производительности по сравнению с FIPS 180-4.

Длина выходных данных для XOFs может быть выбрана под требования конкретного приложения. Сами функции расширенного вывода могут быть адаптированы для хеш-функций в связи с дополнительными соображениями безопасности, или использованы во множестве других приложений. Соответствующее использование XOFs будет определено в специальных публикациях NIST.

Перестановки *KECCAK-p* были разработаны для использования в качестве главных компонентов для множества криптографических функций, включая ключевые функции для аутентификации и/или шифрования (**keyed function for authentication and/or encryption**). Шесть функций SHA-3 можно рассматривать как режимы работы (**modes of operation, modes**) перестановки ***KECCAK-p*[1600, 24]**. В будущем дополнительные моды этой перестановки или другие перестановки *KECCAK-p* могут быть определены и одобрены в публикациях FIPS или специальных публикация NIST (**NIST Special Publications**).

**4. Утверждающий орган (Approving Authority):** Министр торговли (Secretary of Commerce).

**5. Подразделение техподдержки (Maintenance Agency):** Департамент торговли США (U.S.. Department of Commerce), Национальный институт стандартов и технологий (NIST), Лаборатория информационной технологии (**Information Technology Laboratory, ITL**).

**6. Применимость (Applicability):** Данный стандарт применим ко всем федеральным департаментам и агентствам (Federal departments and agencies) по защите конфиденциальной не секретной информации, которая подпадает под действие Раздела 10 United States Code Section 2315 (**10 USC 2315**) и не входит в систему национальной безопасности, что определено в Разделе **40 USC 11103(a)(1)**. Данный стандарт или федеральный стандарт обработки информации FIPS 180 должен быть реализован везде, где требуется безопасный алгоритм хеширования для федеральных приложений (Federal applications), в том числе в качестве компонента других криптографических алгоритмов и протоколов. Настоящий Стандарт может быть принят и использоваться неправительственными организациями (non-Federal Government organization).

**7. Спецификации:** Федеральный стандарт обработки информации (FIPS) 202, Стандарт SHA-3: хеш-функции на основе перестановок и функции расширенного вывода.

**8. Реализации:** Федеральные департаменты и агентства должны использовать реализации перестановок *KECCAK-p* только в режимах работы, одобренных FIPS или рекомендованных NIST, таких как SHA-3 функции, указанные в настоящем стандарте. SHA-3 функции могут быть реализованы в программном обеспечении (software), микропрограммном обеспечении (firmware), оборудовании (hardware) или любой их комбинации. Соответствующими настоящему Стандарту считаются только те реализации этих функций, которые проверены Программой валидации криптографических алгоритмов (**Cryptographic Algorithm Validation Program**). Информация о Программе валидации может быть получена по адресу <http://csrc.nist.gov/groups/STM/cavp/index.html>.

**9. График реализации:** Настоящий Стандарт вступает в силу немедленно. Приложения или расширения настоящего Стандарта, которые зависят от выпуска новых или пересмотренных Специальных публикаций NIST, вступают в силу после окончательной публикации поддерживающих Специальных публикаций.

**10. Патенты:** Реализации SHA-3 функций в настоящем Стандарте могут быть защищены иностранными патентами или патентами США.

**11. Экспортный контроль:** Определенные криптографические устройства и технические данные, относящиеся к ним, подлежат федеральному экспортному контролю. Экспорт криптографических модулей реализующих настоящий Стандарт и относящихся к ним технических данных, должен соответствовать федеральным нормам и иметь лицензию Бюро экспортного управления Министерства торговли США (Bureau of Export Administration of the U.S. Department of Commerce). Информация об экспортном регулировании доступна по адресу: <http://www.bis.doc.gov/index.htm>

**12. Квалификация:** Не смотря на то, что настоящий Стандарт определяет математические функции, являющиеся подходящими компонентами для приложений информационной безопасности, соответствие настоящему Стандарту не гарантирует безопасность конкретной реализации. Ответственный орган в каждом агентстве или департаменте должен гарантировать, что общая реализация обеспечивает приемлемый уровень безопасности. Настоящий Стандарт будет пересматриваться каждые пять лет в порядке оценки его адекватности.

**13. Процедура отказа от требования (Waiver Procedure):** Федеральный закон об управлении информационной безопасностью (**FISMA, Federal Information Security Management Act**) не допускает отказов от обязательных требований FIPS, установленных Министром торговли.

**14. Где можно получить копии стандарта:** Настоящая публикация доступна по адресу <http://csrc.nist.gov/publications/>. Другие публикации по компьютерной безопасности, выпущенные NIST, доступны на том же веб-сайте.

**Содержание**

**1. Введение**

**2. Глоссарий**

2.1 Термины и сокращения

2.2 Параметры алгоритма и другие переменные

2.3 Базовые операции и функции

2.4 Специальные функции

**3. *KECCAK-p* перестановки**

3.1 Состояние (State)

3.1.1 Части массива состояний

3.1.2 Конвертация строк в массивы состояний

3.1.3 Конвертация массивов состояний в строки

3.1.4 Соглашение о наименовании массива состояний

3.2 Пошаговое отображение (Step Mapping)

3.2.1 Спецификация***θ***

3.2.2 Спецификация ***ρ***

3.2.3 Спецификация ***π***

3.2.4 Спецификация *χ*

3.2.5 Спецификация ***ι***

3.3 *KECCAK-p*[*b, nr*]

3.4 Сравнение с *KECCAK-f*

**4. Конструкция «губки» (Sponge Construction)**

**5. *KECCAK***

5.1 Спецификация***pad10\*1***

5.2 Спецификация *KECCAK[c]*

**6. SHA-3 Спецификация функций**

6.1 SHA-3 хеш-функции

6.2 SHA-3 функции расширенного вывода

6.3 Альтернативные определения SHA-3 функций расширенного вывода

**7. Соответствие стандарту (Conformance)**

**A. Безопасность**

A.1 Резюме

A.2 Дополнительные соображения о функциях расширенного вывода

**B. Примеры**

B.1 Функции преобразования

B.2 Шестнадцатеричная форма для битов дополнения (padding bits)

**C. Идентификаторы объектов**

**D. Использованная литература**

**Рисунки**

Рисунок 1: Части массива состояния, организованные по размерности

Рисунок 2: ***x*,*y*** и ***z*** координаты для диаграмм пошагового отображения

Рисунок 3: Иллюстрация ***θ***, примененная к одному биту

Рисунок 4: Иллюстрация ***ρ*** для ***b = 200***

Рисунок 5: Иллюстрация ***π***, примененная к одному биту

Рисунок 6: Иллюстрация *χ*, примененная к одному ряду

Рисунок 7: Конструкция «губки»: ***Z=SPONGE*[*f, pad, r*](*N, d*)**

**Таблицы**

Таблица 1: Ширина *KECCAK-p* перестановки и связанные величины

Таблица 2: Смещение **ρ**

Таблица 3: Размеры входных блоков для *HMAC*

Таблица 4: Надежность (security strengths) SHA-1, SHA-2 и SHA-3 функций

Таблица 5: Иллюстрация ***h2b***

Таблица 6: Шестнадцатеричная форма SHA-3 заполнения для сообщений с

байтовым выравниванием

**ВВЕДЕНИЕ**

Настоящий Стандарт определяет новое семейство функций, которое дополняет семейство хеш-функций SHA-1 и SHA-2, определенных в FIPS 180-4 [1]. Это семейство, называемое SHA-3 (Secure Hash Algorithm-3), основано на алгоритме *KECCAK*[2] – победителе конкурса SHA-3 Cryptographic Hash Algorithm Competition[[1]](#footnote-2)[3]. Семейство SHA-3 состоит из четырёх криптографических хеш-функций и двух функций расширенного вывода. Эти шесть функций имеют общую структуру, описанную в [4], а именно, так называемую конструкцию «губки» (***sponge construction***). Функции с данной структурой называются «функциями-губками» (***sponge functions***).

**Хеш-функция**– это функция над двоичными данными (т.е. битовыми строками), для которых длина выходных данных фиксирована[[2]](#footnote-3). Входные данные хеш-функции называются ***сообщением*** (***message***), а выходные данные – ***дайджестом*** (***digest***) или ***хешем*** (**hash value**). Этот дайджест часто служит сжатым представлением сообщения. Четыре SHA-3 хеш-функции называются SHA3-224, SHA3-256, SHA3-384 и SHA3-512; в каждом случае суффикс после тире указывает на фиксированную длину дайджеста, к примеру, SHA3-256 продуцирует 256 битный дайджест. Функции SHA-2, т.e. SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/224 и SHA-512/256, предлагают тот же набор длины дайджеста. Таким образом, SHA-3 хеш-функции могут быть применены как альтернатива функциям SHA-2 и наоборот.

**Функция расширенного вывода**(**XOFs**) – это функции на битовыми строками (также называемыми сообщениями), в которых выходные данные могут быть расширены на любую желаемую длину. Две SHA-3 функции расширенного вывода называются SHAKE128 и SHAKE256[[3]](#footnote-4). Суффиксы «128» и «256» указывают степень безопасности (**security strength**), которую обычно могут поддерживать эти две функции[[4]](#footnote-5), в отличии от суффиксов хеш-функций, которые указывают длину дайджеста. SHAKE128 и SHAKE256 не первые функции расширенного вывода, которые были стандартизированы NIST.

Шесть функций SHA-3 были разработаны для предоставления специальных свойств, таких как сопротивление коллизии (**resistance to collision**), сопротивление атакам нахождения первого и второго прообразов (**preimage and second preimage attacks**). Уровень сопротивления этим трём типам атак кратко изложен в Разделе A.1. Криптографические хеш-функции являются фундаментальными компонентами во многих приложения информационной безопасности, таких как генерация и проверка цифровой подписи (**digital signature generation and verification**), а также генерация псевдослучайных чисел (**pseudorandom bit generation**).

Длина дайджеста в одобренных FIPS хеш-функциях составляет 160, 224, 256, 384 и 512 бит. Когда в приложении требуется криптографическая хеш-функция с нестандартной длиной дайджеста, XOF является естественной альтернативой конструкциям, включающим множественные вызовы хеш-функции и/или усечение выходных бит. Однако, на XOFs распространяются дополнительные соображения безопасности, описанные в Разделе A.2.

Каждая из шести SHA-3 функций использует одну и ту же базовую перестановку в качестве основного компонента конструкции «губки» (**sponge construction**). По сутиSHA-3 функции являются режимами работы (модами) перестановки (**modes of operation, modes**). В настоящем Стандарте перестановка определена как экземпляр семейства перестановок, называемого KECCAK-p, чтобы обеспечить гибкость изменения ее размера и параметров безопасности при разработке любых дополнительных модов в будущих документах.

Четыре SHA-3 хеш-функции немного отличаются от экземпляров KECCAK, предложенных для конкурса SHA-3 [3]. В частности, к сообщениям добавлен двубитный суффикс для того, чтобы отличить SHA-3 хеш-функцию от SHA-3 XOFs и облегчить разработку новых вариантов SHA-3 функций, которые могут быть предназначены для отдельных доменов приложений.

Две функции расширенного вывода SHA-3 определены таким образом, чтобы обеспечить разработку специальных вариантов (**dedicated variants**). Кроме того, функции расширенного вывода SHA-3 совместимы со схемой кодирования Sakura (**Sakura coding scheme**) [6] для дерева хешей (**tree hashing**) [7] для того, чтобы поддерживать разработку параллелизуемых вариантов (**parallelizable variants**) XOFs, определённых в отдельном документе.

Большая часть обозначений и терминологии в настоящем Стандарте согласуется со спецификацией KECCAK в [8].

**2. ГЛОССАРИЙ**

**2.1 Термины и Сокращения**

|  |  |
| --- | --- |
| бит (**bit**) | Двоичная цифра: 0 или 1. |
| байт (**byte**) | Последовательность из 8 бит |
| емкость (**capacity**) | В конструкции «губки»: разница между шириной базовой функции и скорость |
| столбец (**column**) | Для массива состояний: подмассив из пяти бит с константными *x*и *z*координатами |
| дайджест (**digest**) | Данные на выходе криптографической хеш-функции. Также называются хешем. |
| разделение доменов (**domain separation**) | Для функции: разделение данных на разные домены приложений для того, чтобы ни один вход не был назначен более чем одному домену. |
| функция расширенного вывода (**extendable-output function**, **XOF)** | Функция над битовой строкой, в которой выходные данные могут быть расширены на любую желаемую длину |
| **FIPS** | Федеральный стандарт обработки информации (**Federal Information Processing Standard**) |
| **FISMA** | Федеральный закон об управлении информационной безопасностью (**Federal Information Security Management Act**) |
| хеш-функция (**hash function**) | Функция над битовой строкой, в которой длина входных данных фиксирована. Выходные данные часто служат сжатым представлением входных данных |
| хеш (**hash value**) | Смотри «дайджест» |
| **HMAC** | Код аутентификации сообщений, использующий хеш-функцию с ключом (**Keyed-Hash Message Authentication Code**). |
| **KDF** | Функция формирования ключа (**Key derivation function**). |
| **KECCAK** | Семейство всех функций «губок» с перестановкой *KECCAK-f* в качестве базовой функции и мультискоростным заполнением (**multi-rate padding**) в качестве правила заполнения (**padding rule**). KECCAK изначально был указан в [8]. |
| дорожка (**lane**) | Для массива состояния перестановки *KECCAK-p* с шириной *b*, подмассив из *b/25*бит с константными *x* и *y* координатами. |
| сообщение (**message**) | Битовая строка произвольной длины, поступающая на вход SHA-3 функции. |
| мультискоростное заполнение (**multi-rate padding**) | Правило заполнения *pad10\*1*, вывод которого равен 1, за которым следует (возможно, пустая) строка из 0, за которой следует 1 (100…..001). |
| **NIST** | Национальный институт стандартов и технологии (**National Institute of Standards and Technology**) |
| плоскость (**plane**) | Для массива состояния перестановки *KECCAK-p* c шириной *b*, подмассив из *b/5*бит с константной *y* координатой. |
| скорость (**rate**) | В конструкции «губки» количество обработанных входных битов или выходных битов, сгенерированных при каждом вызове базовой функции. |
| раунд (**round**) | Последовательность шагов сопоставлений, которая повторяется при вычислении перестановки *KECCAK-p* |
| константа раунда (**round constant**) | Для каждого раунда перестановки *KECCAK-p* значение дорожки, которая определяется индексом раунда. Константа раунда является вторым входом в пошаговое отображение ***ι*** |
| индекс раунда (**round index**) | Целое значение индекса для раундов перестановки *KECCAK-p*. |
| строка (**row**) | Для массива состояния, подмассив из пяти бит с константными *y* и *z* координатами. |
| **SHA-3** | Алгоритм безопасного хеша 3 (**Secure Hash Algorithm-3**). |
| **SHAKE** | **S**ecure **H**ash **A**lgorithm **KE**CCAK. |
| лист (**sheet**) | Для массива состояния перестановки *KECCAK-p*с длиной *b*подмассив из *b/5*бит с константной *x* координатой. |
| срез (**slice**) | Для массива состояния, подмассив из 25 бит с константной *z* координатой |
| конструкция губки (**sponge construction**) | Метод, первоначально указанный в [4] для определения функции исходя из следующего: 1) базовая функция над битовой строкой фиксированной длины; 2) правило заполнения и 3) скорость (заполнения?). Входные и выходные данные результирующей функции – это битовые строки, которые могут быть сколь угодно длинными. |
| функция губки (**sponge function**) | Функция, которая определяется в соответствии с конструкцией «губки», возможно специализированная для фиксированной длины выходных данных |
| состояние (**state**) | Массив битов, который многократно обновляется в вычислительной процедуре. Для перестановки *KECCAK-p*состояние представляется либо как трёхмерный массив, либо как битовая строка. |
| массив состояний (**state array)** | Для перестановки *KECCAK-p*, массив размера 5\*5\*w бит, которые представляет состояние. Индексы для x, y и z координат находятся в диапазонах [0, 4], [0, 4] и [0, w-1] соответственно. |
| пошаговое отображение (**step mapping**) | Один из пяти компонентов раунда перестановки KECCAK-p: **θ**, **ρ**, **π**, **χ** или **ι** |
| строка (**string**) | Для неотрицательных целых *m*, последовательность из *m* символов |
| ширина (**width)** | В конструкции губки, фиксированная длина входных и выходных данных базовой функции |
| **XOF** | Extendable-Output Function |
| **XOR** | Булева операция **исключающее ИЛИ**, обозначается знаком ⨁ |

**2.2 Параметры алгоритма и другие переменные**

|  |  |
| --- | --- |
| **A** | Массив состояния |
| **A[*x*, *y*, *z*]** | Для массива состояния **A**, бит соответствующий тройке (x, y, z). |
| ***b*** | Длина перестановки *KECCAK-p* в битах |
| ***c*** | Емкость функции «губки» |
| ***d*** | Длина дайджеста хеш-функции или требуемая длина выходных данных XOF в битах |
| ***f*** | Общая базовая функция для конструкции «губки» |
| ***ir*** | Индекс раунда для перестановки *KECCAK-p*. |
| ***J*** | Входная строка для RawSHAKE128 и RawSHAKE256 |
| ***ι*** | Для перестановки KECCAK-p двоичный логарифм от размера дорожки, т.е. log2(*w*). |
| ***Lane*(*i, j*)** | Для массива состояний **A** – строка всех битов дорожки, чьи координаты *x* и *y* равны *i* и *j*. |
| ***M*** | Входная строка для SHA-3 хеш-функции или функции расширенного вывода. |
| ***N*** | Входная строка для функции SPONGE[*f*, *pad*, *r*] KECCAK[*c*]. |
| ***nr*** | Количество рандов для перестановки *KECCAK-p* |
| **pad** | Обобщенное правило заполнения для конструкции «губки». |
| ***Plane*(*j*)** | Для массива состояний **A** – строка всех битов плоскости, у которых координата *y* равна *j*. |
| ***r*** | Скорость функции «губки» |
| ***RC*** | Для раунда перестановки *KECCAK-p* – константа раунда |
| ***w*** | Размер дорожки перестановки *KECCAK-p* в битах, т.е. *b/25*. |

**2.3 Базовые операции и функции**

|  |  |
| --- | --- |
| **0S** | Для положительного целого *s*, 0S это строка, которая состоит из s последовательных нулей. Если *s*=0, тогда это пустая строка |
| **len(*X*)** | Для битовой строки *X*, len(*X*) это длина *X* в битах. |
| ***X*[*i*]** | Для строки *X* и целого *i*, такого что 0 ≤ i < len(*X*), X[*i*] – бит *X* с индексом *i*. Битовый строки изображаются с индексами, возрастающими слева направо, так что *X*[0] появляется слева, затем *X*[1] и т.д. |
| **TruncS(*X*)** | Для положительного целого s и строки X TruncS(*X*) – это строка, состоящая из битов *X*[0]…*X*[*s*-1]. Например, если Trunc2(10100)=10. |
| ***X* ⨁ *Y*** | Для строк *X* и *Y* равной длины, *X*⨁*Y* – это строка, которая является результатом булевой операции «Исключающее ИЛИ» над *X* и *Y*. Например, 1100 ⨁ 1010 = 0110. |
| ***X || Y*** | Для строк *X* и *Y* *X* || *Y* – это конкатенация *X* и *Y*. Например, 11001 || 010 = 11001010. |
| ***m/n*** | Для целых *m* и *n*, *m*/*n* – это частное (**quotient**), т.е. *m* деленное на *n*. |
| ***m* mod *n*** | Для целых *m* и *n*, *m* mod *n* – это целое *r*, для которого  0 ≤ *r* < *n* и *m*–*r* кратно *n*. Например, 11 mod 5 = 1, и –11 mod 5 = 4 |
| **⌈*x*⌉** | Для действительного числа *x*, ⌈*x*⌉ – это наименьшее целое, которое не строго меньше *x*. Например, ⌈3.2⌉=4,  ⌈-3.2⌉ = -3, ⌈6⌉=6 |
| **log2(*x*)** | Для положительного действительного числа *x*, log2(*x*) это действительное число *y*, такое что 2*y*=*x*. |
| **min(*x*, *y*)** | Для действительных чисел *x* и *y*, min(*x*, *y*) является минимумом *x* и *y*. Например, min(9, 33)=9. |

**2.4 Специальные функции**

**3. *KECCAK-p* перестановки**

**3.1 Состояние (State)**

**3.1.1 Части массива состояний**

**3.1.2 Конвертация строк в массивы состояний**

**3.1.3 Конвертация массивов состояний в строки**

**3.1.4 Соглашение о наименовании массива состояний**

**3.2 Пошаговое отображение (StepMapping)**

**3.2.1 Спецификация*θ***

**3.2.2 Спецификация *ρ***

**3.2.3 Спецификация *π***

**3.2.4 Спецификация *χ***

**3.2.5 Спецификация *ι***

**3.3 *KECCAK-p*[*b, nr*]**

**3.4 Сравнениес*KECCAK-f***

**4. Конструкция «губки» (SpongeConstruction)**

**5. *KECCAK***

**5.1 Спецификация*pad10\*1***

**5.2 Спецификация *KECCAK[c]***

**6. SHA-3 Спецификация функций**

**6.1 SHA-3 хеш-функции**

**6.2 SHA-3 функции расширенного вывода**

**6.3 Альтернативные определения SHA-3 функций расширенного вывода**

**7. Соответствие стандарту (Conformance)**

**A. Безопасность**

**A.1 Резюме**

**A.2 Дополнительные соображения о функциях расширенного вывода**

**B. Примеры**

**B.1 Функциипреобразования**

**B.2 Шестнадцатеричная форма для битов дополнения (paddingbits)**

**C. Идентификаторыобъектов**

**D. Использованнаялитература**

1. Точнее, конкурс требовал четыре хеш-функции, а KECCAK – это более широкое семейство функций. [↑](#footnote-ref-2)
2. Для многих хеш-функций, есть ограничение (довольно большое) на длину входных данных. [↑](#footnote-ref-3)
3. Имя «SHAKE» было предложено в [5] как комбинация терминов «**S**ecure **H**ash **A**lgorithm» и «**KE**CCAK». [↑](#footnote-ref-4)
4. За исключением случая, когда длина выходных данных достаточна мала; смотри обсуждение в секции A.1 [↑](#footnote-ref-5)