**ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ**

# Ασφάλεια και κρυπτογραφικές εφαρμογές σε ενσωματωμένα συστήματα

ΑΡΓΥΡΙΟΣ Σ. ΣΙΔΕΡΗΣ

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ
ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ

ΣΕΠΤΕΜΒΡΙΟΣ 2025
ΚΟΖΑΝΗ

**ΕΠΙΒΛΕΠΩΝ**

Μηνάς Δασυγένης, Αναπληρωτής Καθηγητής Π.Δ.Μ.

**ΜΕΛΗ ΤΡΙΜΕΛΟΥΣ**
**ΣΥΜΒΟΥΛΕΥΤΙΚΗΣ ΕΠΙΤΡΟΠΗΣ**

Μηνάς Δασυγένης, Αναπληρωτής Καθηγητής Π.Δ.Μ.
Παναγιώτης Σαριγιαννίδης, Καθηγητής Π.Δ.Μ.
Αθανάσιος Κακαρούντας, Αναπληρωτής Καθηγητής Π.Θ.

**ΜΕΛΗ ΕΠΤΑΜΕΛΟΥΣ**
**ΕΞΕΤΑΣΤΙΚΗΣ ΕΠΙΤΡΟΠΗΣ**

Μηνάς Δασυγένης, Αναπληρωτής Καθηγητής Π.Δ.Μ.
Παναγιώτης Σαριγιαννίδης, Καθηγητής Π.Δ.Μ.
Αθανάσιος Κακαρούντας, Αναπληρωτής Καθηγητής Π.Θ.
Όνομα Επίθετο, Θέση, Ίδρυμα
Όνομα Επίθετο, Θέση, Ίδρυμα
Όνομα Επίθετο, Θέση, Ίδρυμα
Όνομα Επίθετο, Θέση, Ίδρυμα

Η έγκριση της παρούσας διδακτορικής διατριβής από το τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Δυτικής Μακεδονίας δεν σημαίνει αποδοχή των γνωμών του συγγραφέα.

*Ποτέ δεν είναι αργά για να γίνεις*
*αυτό που θα μπορούσες να είχες γίνει.*

George Eliot, 1819-1880, Αγγλίδα συγγραφέας

# Περίληψη

Στον σύγχρονο κόσμο της τεχνολογίας, όπου η διαχείριση και η προστασία των δεδομένων αποτελούν πρωταρχική ανησυχία, η ασφάλεια και οι κρυπτογραφικές εφαρμογές αποκτούν ολοένα και μεγαλύτερη σημασία. Η εισαγωγή και εξέλιξη νέων κρυπτογραφικών προτύπων, όπως ο SHA-3 (Secure Hash Algorithm 3), σηματοδοτεί μια νέα εποχή στην κρυπτογραφία. Αυτό το πρότυπο προσφέρει αυξημένη ασφάλεια και ανθεκτικότητα σε κυβερνοεπιθέσεις, ξεπερνώντας τα προηγούμενα συστήματα σε αποδοτικότητα και αξιοπιστία. Επιπλέον, η ενσωμάτωση τέτοιων κρυπτογραφικών μηχανισμών σε ενσωματωμένα συστήματα, ειδικά σε Field-Programmable Gate Arrays (FPGAs), ανοίγει νέους δρόμους για την ανάπτυξη ασφαλέστερων και πιο αποδοτικών τεχνολογικών λύσεων.

Τα FPGA, με τη μοναδική τους ικανότητα να προγραμματίζονται και να επαναπρογραμματίζονται στο πεδίο, παρέχουν μια ιδανική πλατφόρμα για την υλοποίηση και την βελτιστοποίηση των κρυπτογραφικών εφαρμογών. Η ευελιξία τους επιτρέπει την προσαρμογή σε συγκεκριμένες ανάγκες και την ανταπόκριση σε διαφορετικές κρυπτογραφικές προκλήσεις. Ο SHA-3, ως ένα από τα πιο πρόσφατα και ασφαλή πρότυπα, προσφέρει βελτιωμένη ασφάλεια συγκριτικά με προηγούμενα πρότυπα, όπως τον SHA-2. Η εφαρμογή του σε FPGA επιτρέπει την ταχύτερη και πιο αποδοτική επεξεργασία των δεδομένων, προσφέροντας έναν αξιόπιστο τρόπο προστασίας της εμπιστευτικότητας και της ακεραιότητας των δεδομένων.

Η υιοθέτηση τέτοιων τεχνολογιών είναι κρίσιμη στη σύγχρονη εποχή, καθώς η ανάγκη για αποδοτικότερη διαχείριση δεδομένων αυξάνεται με ταχύτατους ρυθμούς σε διάφορους τομείς, από το ηλεκτρονικό εμπόριο μέχρι τους κυβερνητικούς οργανισμούς. Η ενσωμάτωση καινοτόμων κρυπτογραφικών λύσεων, όπως ο SHA-3 σε FPGA, όχι μόνο βελτιώνει την ασφάλεια των ψηφιακών συστημάτων, αλλά επίσης αυξάνει την ευελιξία και την ικανότητα προσαρμογής τους σε εξελισσόμενες απαιτήσεις ασφαλείας, διασφαλίζοντας την προστασία των δεδομένων σε ένα γρήγορα μεταβαλλόμενο τεχνολογικό περιβάλλον. Η πρόκληση στον τομέα της υψηλής τεχνολογίας εντοπίζεται στην επίτευξη ενός ιδανικού συνδυασμού μεταξύ

ταχύτητας επεξεργασίας, ενεργειακής αποδοτικότητας και μείωσης του κόστους σε επιφάνεια ολοκλήρωσης, ειδικά στην υλοποίηση προηγμένων κρυπτογραφικών λειτουργιών όπως ο SHA-3.

Στην παρούσα διατριβή, ο κεντρικός στόχος είναι η ανάπτυξη αποδοτικών τεχνικών επιτάχυνσης για τη βελτιστοποίηση της ρυθμαπόδοσης και/ή της αποδοτικότητας σε κρυπτογραφικές εφαρμογές που υλοποιούνται σε FPGA, και ειδικότερα στον κρυπτογραφικό αλγόριθμο SHA-3. Η επιτάχυνση του αλγορίθμου επιδιώκεται με τέτοιο τρόπο ώστε οι προτεινόμενες λύσεις να είναι υλοποιήσιμες σε πραγματικές εφαρμογές, καλύπτοντας τις απαιτήσεις για επεξεργασία μεγάλων όγκων δεδομένων με ταχύτητα και ασφάλεια. Η βελτίωση και επιτάχυνση της ρυθμαπόδοσης αποτελεί κομβικό σημείο της έρευνας, λόγω της αυξανόμενης ανάγκης για ταχύτερη και αποδοτικότερη επεξεργασία δεδομένων σε σύγχρονες πληροφοριακές υποδομές. Έτσι, προτείνονται τεχνικές που αξιοποιούν την ευελιξία των FPGA, αναδεικνύοντας την ικανότητά τους να υποστηρίζουν προσαρμοστικές και κρυπτογραφικές εφαρμογές με έμφαση στην επιτάχυνση και ταυτόχρονα στη διατήρηση υψηλών επιπέδων ρυθμαπόδοσης και αποδοτικότητας.

Οι συνεισφορές που παρουσιάζονται στην παρούσα διατριβή εισάγουν νέες κατευθύνσεις προς πιο προηγμένες, αποδοτικές, επιταχυνόμενες και βιώσιμες λύσεις στον τομέα των κρυπτογραφικών εφαρμογών με SHA-3 που υλοποιούνται σε ενσωματωμένα συστήματα. Ιδιαίτερη έμφαση δίνεται στην ανάπτυξη και αξιολόγηση τεχνικών επιτάχυνσης, οι οποίες στοχεύουν στη βελτιστοποίηση της ρυθμαπόδοσης και/ή της αποδοτικότητας, συμβάλλοντας ουσιαστικά στη δημιουργία ταχύτερων και αποτελεσματικότερων συστημάτων. Η διατριβή αναδεικνύει επίσης τη σημασία της συνεχούς έρευνας και ανάπτυξης στον συγκεκριμένο τομέα, καθώς οι τεχνολογικές προκλήσεις εντείνονται και οι απαιτήσεις για ταχύτερους χρόνους επεξεργασίας γίνονται ολοένα και πιο επιτακτικές.

---

**Λέξεις Κλειδιά:** Επιτάχυνση υλικού, Βελτιστοποίηση υλικού, Ρυθμαπόδοση, Αποδοτικότητα, SHA-3, FPGA

# Abstract

In today's technology world, where data management and protection are primary concerns, security and cryptographic applications are becoming increasingly important. Introducing and evolving new cryptographic standards, such as SHA-3 (Secure Hash Algorithm 3), marks a new era in cryptography. This standard offers increased security and resilience to cyber-attacks, surpassing previous systems' efficiency and reliability. Moreover, integrating such cryptographic mechanisms in embedded systems, especially in Field-Programmable Gate Arrays (FPGAs), opens new avenues for developing safer and more efficient technological solutions.

With their unique ability to be programmed and reprogrammed in the field, the FPGAs provide an ideal platform for implementing and optimizing cryptographic applications. Their flexibility allows adaptation to specific needs and responses to different cryptographic challenges. SHA-3, as one of the latest and most secure standards, offers improved security compared to earlier standards such as SHA-2. Its implementation in FPGA enables faster and more efficient data processing, offering a reliable way to protect data confidentiality and integrity.

Adopting such technologies is critical in the modern era, as the need for faster data management is growing rapidly in various sectors, from e-commerce to government organizations. Integrating innovative cryptographic solutions such as SHA-3 into FPGAs improves the security of digital systems and increases their flexibility and ability to adapt to evolving security requirements, ensuring data protection in a rapidly changing technological environment. The challenge in the high-tech sector is achieving an ideal combination of processing speed, energy efficiency and cost reduction at the integration surface, especially in implementing advanced cryptographic functions such as SHA-3.

In this thesis, the central objective is the development of efficient acceleration techniques for optimizing throughput and/or efficiency in cryptographic applications implemented on FPGAs, with a particular focus on the SHA-3 cryptographic algorithm.

The acceleration of the algorithm is pursued in such a way that the proposed solutions are feasible for real-world applications, addressing the requirements for fast and secure processing of large volumes of data. The improvement and acceleration of throughput constitutes a pivotal aspect of this research, driven by the increasing need for faster and more efficient data processing in modern information infrastructures. Accordingly, the proposed techniques leverage the flexibility of FPGAs, highlighting their capability to support adaptive and cryptographic applications with a strong emphasis on acceleration, while simultaneously maintaining high levels of throughput and efficiency.

The contributions presented in this dissertation introduce new directions toward more advanced, efficient, accelerated, and sustainable solutions in the field of cryptographic applications based on SHA-3 implemented in embedded systems. Special emphasis is placed on the development and evaluation of acceleration techniques, which aim to optimize throughput and/or efficiency, thereby contributing substantially to the creation of faster and more effective systems. The dissertation also highlights the importance of continuous research and development in this field, as technological challenges intensify and the demand for faster processing times becomes increasingly pressing.

---

**Keywords:** Hardware acceleration, Hardware optimization, Throughput, Efficiency, SHA-3, FPGA

# *Ευχαριστίες*

Θα ήθελα να ευχαριστήσω τον Αναπληρωτή Καθηγητή του τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Δυτικής Μακεδονίας Μηνά Δασυγένη, για την καθοδήγησή του, την αμέριστη συμπαράστασή του, αλλά και την εμπιστοσύνη στο πρόσωπό μου. Ευχαριστώ επίσης τα μέλη της διδακτορικής μου συμβουλευτικής επιτροπής, τον καθηγητή Παναγιώτη Σαριγιαννίδη και τον καθηγητή Αθανάσιο Κακαρούντα για τα διορατικά σχόλια, τις συστάσεις και την τεχνογνωσία τους, που βελτίωσαν τη συνολική ποιότητα αυτής της διατριβής. Ακόμα, θα ήθελα να ευχαριστήσω όλους τους συν-συγγραφείς των ερευνητικών μας δημοσιεύσεων, για την καλή μας συνεργασία, τις συζητήσεις ιδεών, αλλά και την καθοδήγησή τους. Τέλος, θα ήθελα να ευχαριστήσω την οικογένεια μου, για τη συμπαράσταση αλλά και κατανόησή τους κατά τη διαδικασία αυτή.

# Δημοσιεύσεις

Δημοσιεύσεις που στηρίζεται η παρούσα Διδακτορική Διατριβή.

### Διεθνή Περιοδικά (με κρίση)

1. Sideris, A., Sanida, T. and Dasygenis, M. (2024). Hardware acceleration design of the SHA-3 for high throughput and low area on FPGA. Journal of Cryptographic Engineering, 1-13. **(Q2)**
2. Sideris, A., Sanida, T. and Dasygenis, M. (2023). A Novel Hardware Architecture for Enhancing the Keccak Hash Function in FPGA Devices. Information, 14(9), 475. **(Q2)**
3. Sideris, A. and Dasygenis, M. (2023). Enhancing the Hardware Pipelining Optimization Technique of the SHA-3 via FPGA. Computation, 11(8), 152. **(Q2)**
4. Sideris, A., Sanida, T., Tsiktsiris, D., and Dasygenis, M. (2022). Acceleration of Image Processing with SHA-3 (Keccak) Algorithm using FPGA. Journal of Engineering Research and Sciences, 1(7), 20–28.
5. Sideris, A., Sanida, T. and Dasygenis, M. (2020). High throughput implementation of the keccak hash function using the nios-ii processor. Technologies, 8(1), 15. **(Q1)**

### Πρακτικά Διεθνών Συνεδρίων (με κρίση)

1. Sideris, A., Sanida, T., Sanida, M. V., Dossis, and M., Dasygenis, M. (2023). Accelerate Processing of Image with the Keccak-512 Algorithm on Cryptoprocessor. In 2023 8th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM) (pp. 1-4). IEEE.

2. Sideris, A., Sanida, T., Chatzisavvas, A., Dossis, M. and Dasygenis, M. (2022). High Throughput of Image Processing with Keccak Algorithm using Microprocessor on FPGA. In 2022 7th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM) (pp. 1-4). IEEE.

3. Sideris, A., Sanida, T., Tsiktsiris, D. and Dasygenis, M. (2022). Image Hashing Based on SHA-3 Implemented on FPGA. In Recent Advances in Manufacturing Modelling and Optimization: Select Proceedings of RAM 2021 (pp. 521-530). Singapore: Springer Nature Singapore.

4. Sideris, A., Sanida, T. and Dasygenis, M. (2020). High throughput pipelined implementation of the SHA-3 cryptoprocessor. In 2020 32nd International Conference on Microelectronics (ICM) (pp. 1-4). IEEE.

5. Sideris, A., Sanida, T. and Dasygenis, M. (2019). Hardware acceleration of SHA-256 algorithm using NIOS-II processor. In 2019 8th International Conference on Modern Circuits and Systems Technologies (MOCAST) (pp. 1-4). IEEE.

# Περιεχόμενα

# Κατάλογος Σχημάτων

# Κατάλογος Πινάκων

# Κατάλογος Αλγορίθμων

# Συντομογραφίες

| | |
|---|---|
| **AES** | Advanced Encryption Standard |
| **ALM** | Adaptive Logic Module |
| **ASIC** | Application Specific Integrated Circuit |
| **CAD** | Computer Aided Design |
| **CPU** | Central Processing Unit |
| **DDR4** | Double Data Rate 4 |
| **DSE** | Design Space Explorer |
| **DSP** | Digital Signal Processor |
| **FPGA** | Field Programmable Gate Arrays |
| **GB** | Gigabyte |
| **Gbps** | Gigabits per second |
| **GHz** | Gigahertz |
| **GPU** | Graphics Processing Unit |
| **HDL** | Hardware Description Language |
| **HMAC** | Hash-based Message Authentication Code |
| **IoT** | Internet of Things |
| **MACs** | Message Authentication Codes |
| **Mbps** | Megabits per second |
| **MD** | Merkle-Damgård |
| **MD-strengthening** | Message Digest strengthening |
| **MD4** | Message Digest 4 |
| **MD5** | Message Digest 5 |
| **MHz** | Megahertz |

| | |
|---|---|
| **NIST** | National Institute of Standards and Technology |
| **PKI** | Public Key Infrastructure |
| **PLL** | Phase Locked Loop |
| **RC** | Round Constant |
| **RSA** | Rivest Shamir Adleman |
| **SDRAM** | Synchronous Dynamic Random-Access Memory |
| **SET** | Secure Electronic Transactions |
| **SHA-3** | Secure Hash Algorithm 3 |
| **SoC** | System on a Chip |
| **SSL** | Secure Sockets Layer |
| **TLS** | Transport Layer Security |
| **VHDL** | Very High Speed Integrated Circuit HDL |

# Ελληνικές Ορολογίες

| | |
|---|---|
| **Authentication** | Αυθεντικοποίηση |
| **Certification** | Πιστοποίηση |
| **Confidentiality** | Εμπιστευτικότητα |
| **Data integrity** | Ακεραιότητα δεδομένων |
| **Efficiency** | Αποδοτικότητα |
| **Hashing** | Κατακερματισμός |
| **Non-repudiation** | Μη αποποίηση |
| **Pipeline** | Διασωλήνωση |
| **Throughput** | Ρυθμαπόδοση |
| **Unrolling** | Ξετύλιγμα |

# Κεφάλαιο 1

# Εισαγωγή

Το παρόν κεφάλαιο εισάγει τον αναγνώστη στον διαρκώς εξελισσόμενο τομέα της κρυπτογραφίας, παρέχοντας μια ολοκληρωμένη επισκόπηση της επιστήμης αυτής και των πολλαπλών εφαρμογών της. Αρχικά, πραγματοποιείται μια σύντομη ιστορική αναδρομή, η οποία παρουσιάζει την εξέλιξη της κρυπτογραφίας διαχρονικά, αναδεικνύοντας τις κομβικές στιγμές και τις σημαντικές μεταβολές που έχει υποστεί ο κλάδος. Στη συνέχεια, το κεφάλαιο εστιάζει στην περιγραφή και ανάλυση των συναρτήσεων κατακερματισμού, καθώς και στον σχεδιασμό τους σε υλικό, υπογραμμίζοντας τη σημασία και τη συμβολή τους στη σύγχρονη κρυπτογραφική τεχνολογία. Επιπλέον, παρουσιάζονται αναλυτικά οι ερευνητικοί στόχοι και τα βασικά ερευνητικά ερωτήματα που θέτει η παρούσα διατριβή, καθορίζοντας τις κατευθύνσεις και τις κύριες προκλήσεις της ερευνητικής αυτής προσπάθειας. Τέλος, παρατίθεται η δομή της διατριβής, με συνοπτική επισκόπηση των επόμενων κεφαλαίων και των κυριότερων θεματικών ενοτήτων που θα αναπτυχθούν στη συνέχεια.

## 1.1 Ιστορική αναδρομή στην κρυπτογραφία

Η δεκαετία του ΄60 σηματοδότησε μια κομβική περίοδο στην εξέλιξη της τεχνολογίας, καθώς η εμφάνιση και η εδραίωση των υπολογιστικών συστημάτων και των τηλεπικοινωνιακών δικτύων επέφεραν ριζικές αλλαγές στον τρόπο διαχείρισης και μετάδοσης της πληροφορίας. Αυτή η τεχνολογική επανάσταση ανέδειξε παράλληλα την ανάγκη για ενισχυμένη ασφάλεια δεδομένων, με αποτέλεσμα η προστασία των

ψηφιακών πληροφοριών και η ασφαλής παροχή ψηφιακών υπηρεσιών να καταστούν αναπόσπαστο μέρος της νέας τεχνολογικής πραγματικότητας [1].

Η ανάγκη για ασφαλή μετάδοση και αποθήκευση δεδομένων, καθώς και η αντιμετώπιση των κυβερνοαπειλών, αναδείχθηκαν σε βασικές προτεραιότητες κατά τις επόμενες δεκαετίες. Αυτή η ανάγκη οδήγησε στην ανάπτυξη καινοτόμων τεχνικών και πρωτοκόλλων ασφάλειας, τα οποία συνεχίζουν να εξελίσσονται μέχρι σήμερα, προσφέροντας ένα ισχυρό θεμέλιο για την πρόοδο της κρυπτογραφίας και της κυβερνοασφάλειας [2, 3]. Ο τομέας των Τεχνολογιών Πληροφορικής και Επικοινωνιών (ΤΠΕ) γνώρισε έκτοτε εκθετική ανάπτυξη, η οποία επέτρεψε την υλοποίηση και εφαρμογή σύνθετων αλγορίθμων, ενώ η πρόοδος στις ασύρματες επικοινωνίες εισήγαγε νέες προκλήσεις στον τομέα της ασφάλειας δεδομένων. Αυτές οι εξελίξεις επηρέασαν καθοριστικά τον κλάδο της κρυπτογραφίας, ο οποίος έχει πλέον αναδειχθεί σε έναν κρίσιμο παράγοντα για τη διασφάλιση της εμπιστευτικότητας και της ακεραιότητας των συστημάτων επικοινωνίας [4, 5].

Η συνεχής πρόοδος και η ενσωμάτωση προηγμένων κρυπτογραφικών τεχνικών και τεχνολογιών αποτελούν αναγκαία προϋπόθεση για τη διασφάλιση των ψηφιακών πληροφοριών. Σε ένα διαρκώς μεταβαλλόμενο και αλληλένδετο τεχνολογικό περιβάλλον, η ανάπτυξη και η υιοθέτηση ισχυρών μεθόδων κρυπτογράφησης καθίστανται θεμελιώδους σημασίας για την αποτελεσματική προστασία έναντι σύγχρονων απειλών, όπως η κυβερνοκλοπή, η κακόβουλη παρεμβολή και η διαδικτυακή απάτη. Η ενίσχυση του επιπέδου ασφάλειας των πληροφοριών ενισχύει την εμπιστοσύνη τόσο μεταξύ των χρηστών όσο και μεταξύ των οργανισμών, διασφαλίζοντας τη βιωσιμότητα και τη λειτουργικότητα των πληροφοριακών συστημάτων σε παγκόσμιο επίπεδο [6, 7].

Στον τομέα της ασφάλειας πληροφοριών, η σημασία της επικοινωνίας, οι απαιτήσεις ασφαλείας και η διαθεσιμότητα των συστημάτων και πόρων από τα εμπλεκόμενα μέρη είναι κρίσιμα στοιχεία. Ένας από τους πυλώνες της ασφάλειας πληροφοριών είναι η ψηφιακή υπογραφή [8–10], η οποία υπηρετεί πολλαπλούς σκοπούς. Αποτελεί βασικό συστατικό σε υπηρεσίες όπως η μη αποποίηση, δηλαδή την αδυναμία αρνήσεως της συμμετοχής σε μια ψηφιακή συναλλαγή, η επικύρωση της προέλευσης των δεδομένων, η αυθεντικοποίηση της ταυτότητας του αποστολέα, καθώς και η διασφάλιση της γνησιότητας και ακεραιότητας των πληροφοριών. Στον κόσμο των ηλεκτρονικών πληροφοριών, η έννοια της υπογραφής χρήζει επαναπροσδιορισμού. Δεν μπορεί απλώς να θεωρηθεί ως κάτι στατικό και αμετάβλητο για τον υπογράφοντα, αλλά πρέπει επίσης να συνδέεται άμεσα με το

περιεχόμενο που υπογράφεται. Σε αντίθεση με μια φυσική υπογραφή, η οποία είναι ένα σταθερό σύμβολο της ταυτότητας ενός ατόμου, η ψηφιακή υπογραφή είναι μια δυναμική σφραγίδα που αλλάζει ανάλογα με τα δεδομένα που συνοδεύει. Αυτή η προσαρμοστικότητα εξασφαλίζει ότι η ψηφιακή υπογραφή παρέχει μια ισχυρή διασφάλιση της ακεραιότητας και της αυθεντικότητας των ψηφιακών πληροφοριών, συμβάλλοντας σημαντικά στην εξέλιξη των μέτρων ασφαλείας στον ψηφιακό κόσμο [11, 12].

Στον κόσμο της κρυπτογραφίας, υπάρχουν ορισμένες θεμελιώδεις έννοιες που καθορίζουν τη σημασία της επιστήμης αυτής:

- Πρώτον, η εμπιστευτικότητα (confidentiality) [13], η οποία συχνά αναφέρεται και ως μυστικότητα, συνιστά θεμελιώδη αρχή και υπηρεσία στην κρυπτογραφία. Η εμπιστευτικότητα αφορά στη διασφάλιση ότι το περιεχόμενο των πληροφοριών παραμένει προστατευμένο από μη εξουσιοδοτημένη πρόσβαση, με στόχο να διαφυλάσσεται η πληροφορία και να αποτρέπεται η αποκάλυψή της σε τρίτα, μη εξουσιοδοτημένα μέρη. Ο απώτερος σκοπός της εμπιστευτικότητας είναι να διασφαλίζει ότι οι πληροφορίες είναι προσβάσιμες αποκλειστικά από τα άτομα ή τα συστήματα στα οποία απευθύνονται.

- Δεύτερον, η έννοια της ακεραιότητας δεδομένων (data integrity) [14] είναι εξίσου σημαντική. Η ακεραιότητα αφορά στη διασφάλιση ότι τα δεδομένα παραμένουν αναλλοίωτα κατά τη μεταφορά ή την αποθήκευσή τους και προστατεύονται αποτελεσματικά από οποιαδήποτε μη εξουσιοδοτημένη τροποποίηση. Ουσιαστικά, η διατήρηση της ακεραιότητας συνεπάγεται την έγκαιρη ανίχνευση και αποτροπή τυχόν μεταβολών, εισαγωγών, διαγραφών ή αντικαταστάσεων στα δεδομένα από μη εξουσιοδοτημένες οντότητες. Η διασφάλιση της ακεραιότητας είναι ζωτικής σημασίας, καθώς παρέχει τη βεβαιότητα ότι τα δεδομένα που παραλαμβάνονται ή ανακτώνται παραμένουν απολύτως ταυτόσημα με αυτά που εστάλησαν ή αποθηκεύτηκαν αρχικά, χωρίς να έχει μεσολαβήσει οποιαδήποτε ανεπιθύμητη ή μη εξουσιοδοτημένη επέμβαση.

- Τρίτον, η αυθεντικοποίηση (authentication) [15, 16] αποτελεί μία από τις βασικότερες έννοιες στην επιστήμη της κρυπτογραφίας και αφορά τη διαδικασία της αναγνώρισης και ταυτοποίησης. Στην ουσία, όταν δύο μέρη επιθυμούν να επικοινωνήσουν, είναι απαραίτητο να μπορούν να αναγνωρίζουν

το ένα το άλλο με ασφάλεια. Αυτή η αναγνώριση δεν αφορά μόνο τις δύο επικοινωνούντες οντότητες, αλλά και την ίδια την πληροφορία που μεταφέρεται μέσω του επικοινωνιακού καναλιού. Έτσι, είναι σημαντικό να επικυρώνεται η προέλευση της πληροφορίας, η ημερομηνία δημιουργίας της, το περιεχόμενό της και άλλα σχετικά στοιχεία. Η αυθεντικοποίηση, λοιπόν, χωρίζεται σε δύο βασικές κλάσεις: την αυθεντικοποίηση οντότητας και την αυθεντικοποίηση προέλευσης δεδομένων [17].

Η αυθεντικοποίηση οντότητας αφορά την ταυτοποίηση και επαλήθευση της ταυτότητας των μερών που συμμετέχουν στην επικοινωνία. Από την άλλη πλευρά, η αυθεντικοποίηση προέλευσης δεδομένων επικεντρώνεται στην επιβεβαίωση της αυθεντικότητας και της προέλευσης των μεταδιδόμενων δεδομένων. Αυτή η κατηγορία της αυθεντικοποίησης μπορεί να παρέχει επίσης την ακεραιότητα δεδομένων, διασφαλίζοντας ότι τα δεδομένα δεν έχουν τροποποιηθεί κατά τη μεταφορά τους. Έτσι, η αυθεντικοποίηση αποτελεί ένα ζωτικό μηχανισμό στην κρυπτογραφία, καθώς εξασφαλίζει ότι η επικοινωνία και η μετάδοση πληροφοριών γίνεται μεταξύ των σωστών μερών και ότι οι πληροφορίες που λαμβάνονται είναι αυθεντικές και αναλλοίωτες. Αυτός ο συνδυασμός ασφαλείας και αξιοπιστίας είναι απαραίτητος σε έναν κόσμο, όπου οι ψηφιακές επικοινωνίες αποτελούν τον κανόνα και όχι την εξαίρεση [18].

- Τέταρτον, μέσα στο πλαίσιο της κρυπτογραφίας και της ασφάλειας πληροφοριών, η μη αποποίηση (non-repudiation) [19] αποτελεί μια θεμελιώδη υπηρεσία που διασφαλίζει ότι μια οντότητα δεν μπορεί να αποποιηθεί την εκτέλεση προηγούμενων συμφωνιών ή ενεργειών. Αυτή η δυνατότητα είναι ιδιαίτερα σημαντική σε σενάρια όπου προκύπτουν διαφωνίες ή διαμάχες λόγω της άρνησης μιας οντότητας να αναγνωρίσει ορισμένες ενέργειες ή συναλλαγές. Για παράδειγμα, ενδέχεται μια οντότητα να προβεί σε μια αγορά ή να συμφωνήσει σε μια συναλλαγή και αργότερα να αρνηθεί αυτές τις ενέργειες. Σε τέτοιες περιπτώσεις, η εξασφάλιση μη αποποίησης γίνεται κρίσιμη για την επίλυση της διαφωνίας.

  Αυτό συχνά επιτυγχάνεται μέσω μηχανισμών που συμπεριλαμβάνουν τη συμμετοχή ενός έμπιστου τρίτου μέρους, το οποίο λειτουργεί ως ανεξάρτητος επιβεβαιωτής των διενεργηθεισών συναλλαγών. Η εφαρμογή της μη αποποίησης μπορεί να περιλαμβάνει την καταγραφή ψηφιακών υπογραφών, την τήρηση εγγράφων ή άλλων τεχνικών επικύρωσης που εγγυώνται

την ακεραιότητα και την αυθεντικότητα των πληροφοριών. Έτσι, η μη αποποίηση αποτελεί έναν κρίσιμο πυλώνα στην ασφάλεια πληροφοριών, καθώς εξασφαλίζει τόσο την ακεραιότητα των συναλλαγών όσο και την ευθύνη των συμμετεχόντων, προσδίδοντας μια επιπλέον διάσταση διαφάνειας και εμπιστοσύνης στην ψηφιακή εποχή [20].

## 1.2 Ιστορική αναδρομή στις κρυπτογραφικές συναρτήσεις κατακερματισμού

Στον τομέα της κρυπτογραφίας, η έννοια της ακεραιότητας είναι αναπόσπαστα συνδεδεμένη με τις συναρτήσεις κατακερματισμού. Οι συναρτήσεις κατακερματισμού διαδραματίζουν καθοριστικό ρόλο, καθώς παράγουν μια μοναδική τιμή σταθερού μήκους για κάθε μήνυμα, καθιστώντας δυνατή την ασφαλή υλοποίηση και επαλήθευση των ψηφιακών υπογραφών. Κατά τις τελευταίες δεκαετίες του 20ού αιώνα, σημειώθηκαν σημαντικές εξελίξεις στον τομέα των συναρτήσεων κατακερματισμού, με καθοριστικό σταθμό την εμφάνιση του προτύπου Message Digest 4 (MD4)[21]. Το MD4, που αναπτύχθηκε από τον Ronald Rivest, αποτέλεσε πρωτοποριακό βήμα για τη διασφάλιση της ακεραιότητας των δεδομένων στη σύγχρονη κρυπτογραφία. Βάσει του πλαισίου που καθιερώθηκε από το MD4, σχεδιάστηκε και το Message Digest 5 (MD5)[22], το οποίο εισήχθη ως ισχυρότερη εκδοχή με στόχο να αντιμετωπίσει ορισμένες αδυναμίες του προκατόχου του. Οι αλγόριθμοι MD4 και MD5 σχεδιάστηκαν για να υπολογίζουν μια σύνοψη μήκους 128 bits, προσφέροντας έτσι μία συμπαγή και μοναδική αναπαράσταση των αρχικών δεδομένων, χωρίς τη δυνατότητα ανακατασκευής τους από τη σύνοψη.

Ωστόσο, εγγενείς αδυναμίες στους αλγόριθμους κατακερματισμού της οικογένειας MD οδήγησαν στην ανάπτυξη της σειράς Secure Hash Algorithm (SHA) από το Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας (National Institute of Standards and Technology, NIST). Ο πρώτος αλγόριθμος της σειράς, ο SHA-0, παρουσιάστηκε το 1993 και σηματοδότησε την αρχή μιας νέας εποχής στους κρυπτογραφικούς κατακερματισμούς. Το 1995 παρουσιάστηκε ο αλγόριθμος SHA-1 [23], ο οποίος παράγει κατακερματισμούς μήκους 160 bit και αρχικά καθιερώθηκε ως διάδοχος του MD5, καθώς οι ευπάθειες του τελευταίου σε επιθέσεις σύγκρουσης (collision attacks) άρχισαν να καθίστανται ανησυχητικά εμφανείς. Η αντοχή σε σύγκρουση (collision resistance) ορίζεται ως η δυσκολία εύρεσης δύο διαφορετικών εισόδων που

να παράγουν την ίδια τιμή κατακερματισμού. Παρά την αρχική αξιοπιστία του, ο ισχυρισμός των Wang et al.[24] ότι είναι εφικτός ο εντοπισμός συγκρούσεων στον SHA-1 με σημαντικά μικρότερη υπολογιστική πολυπλοκότητα απ' ό,τι θεωρούνταν μέχρι τότε, αποτέλεσε σημείο καμπής για την κοινότητα της κρυπτογραφίας. Ο εν λόγω ισχυρισμός τεκμηριώθηκε αργότερα από την εργασία των Marc Stevens et al.[25], οι οποίοι παρουσίασαν πρακτικές συγκρούσεις κατακερματισμού σε διαφορετικά αρχεία, αναδεικνύοντας τα τρωτά σημεία του SHA-1.

Ως απάντηση σε αυτές τις προκλήσεις ασφαλείας, η ανάπτυξη του SHA-2 [26] ξεκίνησε το 2001. Ο SHA-2, σε αντίθεση με τους προκατόχους του, πρόσφερε μια ποικιλία εκδόσεων, συμπεριλαμβανομένων των SHA-256, SHA-384 και SHA-512, που ονομάζονται για τα αντίστοιχα μήκη bit κατακερματισμού τους. Το 2004, μια άλλη παραλλαγή, ο SHA-224, εισήχθη για να ευθυγραμμιστεί με την ισχύ ασφαλείας του 3DES [27], ενός ευρέως χρησιμοποιούμενου προτύπου συμμετρικής κρυπτογράφησης. Ο FIPS-180-3 [28], αναγνώρισε επίσημα αυτούς τους τέσσερις αλγόριθμους ως μέρος του προτύπου SHA-2, υπογραμμίζοντας τη σημασία τους στην κρυπτογραφική ασφάλεια. Η μετάβαση από τον MD4 στον SHA-2 απεικονίζει τη διαρκή εξέλιξη της κρυπτογραφίας με στόχο τη βελτίωση της ακεραιότητας και την ενίσχυση της ασφάλειας των δεδομένων στο σύγχρονο ψηφιακό τοπίο.

Οι αλγόριθμοι SHA-1 και SHA-2 έχουν σχεδιαστεί σύμφωνα με την αρχιτεκτονική Merkle-Damgård (MD) [29]. Στο δομικό αυτό σχήμα, το μήνυμα διαχωρίζεται σε τμήματα (μπλοκ) σταθερού μεγέθους, τα οποία επεξεργάζονται διαδοχικά μέσω μιας ειδικής λειτουργίας συμπίεσης. Ο τελικός κατακερματισμός προκύπτει ως μια συμπαγής, μοναδική αναπαράσταση του αρχικού μηνύματος. Παρά την αποδεδειγμένη αποτελεσματικότητα αυτής της δομής, η εξάρτηση τόσο του SHA-1 όσο και του SHA-2 από το μοντέλο Merkle-Damgård έχει αποτελέσει πηγή ανησυχίας στην επιστημονική κοινότητα. Η κύρια ανησυχία συνίσταται στο ότι ο SHA-2 ενδέχεται να εμφανίζει παρόμοιες ευπάθειες με εκείνες που επέτρεψαν επιθέσεις σύγκρουσης κατά του SHA-1, γεγονός που θα μπορούσε να θέσει σε κίνδυνο την ακεραιότητα των παραγόμενων τιμών κατακερματισμού.

Ως απάντηση στα εντοπισμένα τρωτά σημεία των υπαρχόντων αλγορίθμων κατακερματισμού και στην αυξανόμενη ανάγκη για ένα πιο ασφαλές πρότυπο, το Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας ξεκίνησε το 2007 τη διεξαγωγή ενός διεθνούς διαγωνισμού για την επιλογή του νέου προτύπου κατακερματισμού, που θα ονομαζόταν SHA-3 [30]. Ο διαγωνισμός αυτός αποτέλεσε σημείο αναφοράς για την κρυπτογραφική κοινότητα, προσελκύοντας 64 προτάσεις από ομάδες ερευνητών

παγκοσμίως. Οι υποψήφιοι αλγόριθμοι υπεβλήθησαν σε αυστηρή διαδικασία αξιολόγησης, με ελέγχους ασφάλειας, απόδοσης και θεωρητικής τεκμηρίωσης, σε διαδοχικά στάδια. Στον πρώτο γύρο, η δεξαμενή περιορίστηκε σε 51 υποψήφιους αλγορίθμους. Ο δεύτερος γύρος μείωσε τον αριθμό σε 14, έπειτα από περαιτέρω ανάλυση και έλεγχο [31]. Ο τελικός γύρος περιέλαβε πέντε αλγόριθμους: Grøstl, BLAKE, JH, Keccak και Skein [32].

Μετά από εκτεταμένη διαδικασία ανάλυσης, αξιολόγησης και δοκιμών, τον Οκτώβριο του 2012 σηματοδοτήθηκε ένα ορόσημο στην ιστορία της κρυπτογραφίας: ο αλγόριθμος Keccak ανακηρύχθηκε επίσημα ως ο νέος πρότυπος αλγόριθμος κατακερματισμού, καθιερώνοντας τον ως SHA-3 [33]. Η επιλογή του Keccak δεν βασίστηκε μόνο στα ισχυρά χαρακτηριστικά ασφάλειας που επέδειξε, αλλά και στην καινοτόμο εσωτερική δομή του, η οποία διαφέρει ουσιωδώς από την παραδοσιακή αρχιτεκτονική Merkle-Damgård (MD) και προσφέρει αυξημένη ανθεκτικότητα έναντι πολλών κατηγοριών κρυπτογραφικών επιθέσεων. Η απόφαση αυτή του NIST και της παγκόσμιας κρυπτογραφικής κοινότητας αποτέλεσε σταθμό στην εξέλιξη της ασφάλειας ψηφιακών δεδομένων, εδραιώνοντας την αρχή της συνεχούς προσαρμογής και εξέλιξης έναντι των δυναμικών και διαρκώς μεταβαλλόμενων κυβερνοαπειλών.

Ο SHA-3 υλοποιεί την αρχιτεκτονική της κατασκευής σφουγγαριού (sponge function), η οποία διακρίνεται από δύο διακριτές φάσεις την απορρόφηση και την συμπίεση. Στη φάση απορρόφησης, τα δεδομένα διαβάζονται και επεξεργάζονται, «απορροφώντας» αποτελεσματικά την είσοδο στη δομή του σφουγγαριού. Μετά από αυτό, ξεκινά η φάση συμπίεσης, όπου ο αλγόριθμος «συμπιέζει» την έξοδο. Αυτή η διαδικασία δύο φάσεων επιτρέπει υψηλό βαθμό ευελιξίας και ασφάλειας, καθώς μπορεί να παράγει έξοδο αυθαίρετου μήκους, ένα αξιοσημείωτο πλεονέκτημα έναντι της εξόδου σταθερού μήκους της κατασκευής MD [34]. Πέρα από το λειτουργικό μοντέλο, ο SHA-3 διακρίνεται επίσης ως προς το μέγεθος του μηνύματος και τις δυνατότητες μήκους κατακερματισμού. Έχει σχεδιαστεί για να χειρίζεται ένα ευρύτερο φάσμα μεγεθών εισαγόμενων μηνυμάτων και μπορεί να δημιουργήσει τιμές κατακερματισμού διαφόρων μηκών, προσαρμοσμένες στις συγκεκριμένες απαιτήσεις μιας δεδομένης εφαρμογής. Αυτή η ευελιξία είναι μια σημαντική βελτίωση σε σχέση με τις προηγούμενες εκδόσεις SHA, οι οποίες ήταν πιο άκαμπτες όσον αφορά τις προδιαγραφές μεγέθους εισόδου και εξόδου [35].

Οι συναρτήσεις κατακερματισμού είναι θεμελιώδεις στον τομέα της κρυπτογραφίας και βρίσκουν ένα ευρύ φάσμα εφαρμογών, για τη διασφάλιση της ακεραιότητας

των δεδομένων, την επαλήθευση και την εξουσιοδότηση. Πέρα από το πεδίο της επαλήθευσης μηνυμάτων, οι συναρτήσεις κατακερματισμού διαδραματίζουν κρίσιμο ρόλο στη διαδικασία των ψηφιακών υπογραφών [36], οι οποίες αποτελούν αναπόσπαστο μέρος των μηχανισμών εξουσιοδότησης και επικύρωσης. Οι ψηφιακές υπογραφές χρησιμοποιούν λειτουργίες κατακερματισμού για να δημιουργήσουν μια μοναδική αναπαράσταση του μηνύματος, διασφαλίζοντας την αυθεντικότητα του αποστολέα και την ακεραιότητα του μηνύματος [37].

Επιπλέον, οι συναρτήσεις κατακερματισμού βρίσκουν εφαρμογή τόσο στη δημιουργία ψευδοτυχαίων αριθμών όσο και στην ασφαλή αποθήκευση και προστασία κωδικών πρόσβασης. Η ευελιξία αυτή αναδεικνύει τον καίριο ρόλο τους σε ένα ευρύ φάσμα πρωτοκόλλων και συστημάτων ασφάλειας πληροφοριών, ενισχύοντας τη συνολική αξιοπιστία και ανθεκτικότητα των υπολογιστικών και δικτυακών υποδομών.

## 1.3 Συναρτήσεις κατακερματισμού και κρυπτογραφία

Οι μονόδρομες συναρτήσεις κατακερματισμού [38] λαμβάνουν ως είσοδο μια συμβολοσειρά μεταβλητού μήκους, την οποία αποτελεί το μήνυμα εισόδου και τη μετατρέπουν σε μια συμβολοσειρά εξόδου ορισμένου μήκους. Η διαδικασία αυτή συνήθως παράγει μια συμβολοσειρά εξόδου με μικρότερο μέγεθος από το αρχικό μήνυμα [39]. Η έξοδος των συναρτήσεων κατακερματισμού είναι αποτέλεσμα ενός περίπλοκου μετασχηματισμού που δεν επιδέχεται εύκολης αντιστοίχισης με την αρχική είσοδο. Ένα από τα πιο σημαντικά χαρακτηριστικά των συναρτήσεων κατακερματισμού είναι η ιδιότητα της ευαισθησίας σε αλλαγές εισόδου (avalanche effect), σύμφωνα με την οποία ακόμα και μια μικρή αλλαγή σε ένα μόνο bit του μηνύματος εισόδου προκαλεί σημαντική αλλαγή στην τιμή κατακερματισμού εξόδου. Συγκεκριμένα, μια τέτοια μικρή αλλαγή θα επηρεάσει κατά μέσο όρο τα μισά bit της τιμής κατακερματισμού εξόδου [40]. Με δεδομένη μια συγκεκριμένη τιμή κατακερματισμού, είναι υπολογιστικά ανέφικτο να βρεθεί ένα μήνυμα εισόδου που θα παράγει την ίδια τιμή κατακερματισμού. Αυτή η ιδιότητα καθιστά τις συναρτήσεις κατακερματισμού αξιόπιστα εργαλεία για την επικύρωση της αυθεντικότητας και της ακεραιότητας των δεδομένων σε πληθώρα εφαρμογών ασφαλείας [40].

Η χρήση των μονόδρομων συναρτήσεων κατακερματισμού είναι μια συνηθισμένη πρακτική, διότι επιτρέπει σε οποιονδήποτε να επιβεβαιώσει την τιμή κατακερματισμού ενός μηνύματος. Αντίθετα, όταν ο στόχος είναι να επαληθεύσει την τιμή κατακερματισμού μόνο ένας συγκεκριμένος παραλήπτης, χρησιμοποιούνται οι κώδικες αυθεντικοποίησης μηνυμάτων (Message Authentication Codes, MACs) [41]. Οι κώδικες αυθεντικοποίησης μηνυμάτων συνδυάζουν τις μονόδρομες συναρτήσεις κατακερματισμού με την προσθήκη ενός κρυφού κλειδιού, δημιουργώντας έτσι μία τιμή κατακερματισμού που εξαρτάται τόσο από το μήνυμα εισόδου, όσο και από το κλειδί [42]. Ένας κώδικας αυθεντικοποίησης μηνυμάτων μπορεί να δημιουργηθεί χρησιμοποιώντας είτε μια συμβατική συνάρτηση κατακερματισμού, είτε έναν αλγόριθμο κρυπτογράφησης τμήματος (block cipher). Οι είσοδοι σε μια τέτοια συνάρτηση περιλαμβάνουν ένα τμήμα του μηνύματος εισόδου και την έξοδο των προηγούμενων τμημάτων. Αυτή η διαδικασία, εξασφαλίζει ότι κάθε τμήμα του μηνύματος συμβάλλει στην τελική τιμή κατακερματισμού και παρέχει έναν αξιόπιστο τρόπο για την επαλήθευση της ακεραιότητας και της γνησιότητας του μηνύματος. Συγκεκριμένα, η τιμή κατακερματισμού ενός τμήματος μαζί με το επόμενο τμήμα του μηνύματος αποτελούν τις επόμενες εισόδους στη μονόδρομη συνάρτηση. Τελικά, η τιμή κατακερματισμού ολόκληρου του μηνύματος αποτελείται από την τιμή κατακερματισμού του τελευταίου τμήματος. Για να αποτραπεί το πρόβλημα όπου διαφορετικά μηνύματα με διαφορετικό μήκος μπορούν να έχουν την ίδια τιμή κατακερματισμού, συχνά τα μηνύματα πρέπει να περιέχουν μια δυαδική αναπαράσταση του μήκους τους. Αυτή η τεχνική ονομάζεται Message Digest strengthening (MD-strengthening) και αυξάνει την ασφάλεια της διαδικασίας κατακερματισμού, εξασφαλίζοντας ότι το μήκος του μηνύματος είναι ένας πρόσθετος παράγοντας στην παραγωγή της τελικής τιμής κατακερματισμού. Με αυτόν τον τρόπο, μειώνεται δραστικά η πιθανότητα για τη δημιουργία ίδιων τιμών κατακερματισμού από διαφορετικά μηνύματα, προσθέτοντας ένα σημαντικό στρώμα ασφάλειας στην διαχείριση και την επαλήθευση ψηφιακών δεδομένων [43].

Ο μηχανισμός αυθεντικοποίησης μηνυμάτων έχει ορισμένους κρίσιμους στόχους [44] που επιδιώκει να επιτύχει για να εξασφαλίσει την ασφάλεια και την ακεραιότητα των μεταδιδόμενων δεδομένων:

- Πρώτον, προσπαθεί να χρησιμοποιήσει τις διαθέσιμες συναρτήσεις κατακερματισμού χωρίς να απαιτήσει οποιαδήποτε τροποποίηση σε αυτές, ειδικά σε εκείνες που αποδίδουν καλά σε περιβάλλοντα λογισμικού και είναι ελεύθερα διαθέσιμες.

- Ένας άλλος στόχος είναι η χρησιμοποίηση και διαχείριση των κλειδιών να είναι απλή και απρόσκοπτη. Αυτό εξασφαλίζει ότι οι χρήστες ή τα συστήματα που τον εφαρμόζουν μπορούν εύκολα να διαχειριστούν τα κλειδιά χωρίς περιττή πολυπλοκότητα ή ανάγκη για ειδικές γνώσεις.
- Τρίτον, θα πρέπει να διατηρείται η αρχική απόδοση της συνάρτησης κατακερματισμού χωρίς να προκύπτει σημαντική υποβάθμιση της αποδοτικότητας η οποία θα επιβαρύνει αδικαιολόγητα το σύστημα ή το δίκτυο στο οποίο εφαρμόζεται.
- Τέλος, πρέπει να παρέχει τη δυνατότητα για εύκολη αντικατάσταση της χρησιμοποιούμενης συνάρτησης κατακερματισμού, σε περίπτωση που ανακαλυφθούν νέες, πιο αποδοτικές ή ασφαλέστερες συναρτήσεις.

Αυτή η ευελιξία επιτρέπει την προσαρμογή και την ενημέρωση του συστήματος ασφαλείας για να ανταποκρίνεται στις εξελίξεις και τις αλλαγές στον τομέα της κρυπτογραφίας και της ασφάλειας πληροφοριών [45].

## 1.4 Ενσωματωμένα συστήματα και κρυπτογραφία

Στην εποχή μας, τα ενσωματωμένα συστήματα αποτελούν ένα απαραίτητο δομικό στοιχείο της καθημερινής τεχνολογικής πρακτικής, καθώς παρέχουν αναγκαίες λειτουργίες σε πληθώρα συσκευών καθημερινής χρήσης. Είναι διαμορφωμένα για να εκτελούν ειδικές λειτουργίες, ενσωματώνοντας ιδιαίτερα χαρακτηριστικά σε διάφορα τεχνολογικά προϊόντα, προσδίδοντας έτσι μια σημαντική διάσταση στην ψηφιακή εμπειρία του χρήστη [46]. Αυτά τα συστήματα, που αποτελούν κεντρικό συστατικό μιας σειράς σύγχρονων τεχνολογικών εφαρμογών, διακρίνονται με βάση διάφορες μεταβλητές, όπως το επίπεδο λειτουργικότητας που προσφέρουν, τη διαθέσιμη επεξεργαστική ισχύ, την πολυπλοκότητα της δομής τους, καθώς και το είδος των εφαρμογών στις οποίες αξιοποιούνται [47].

Η βασική αρχιτεκτονική κάθε ενσωματωμένου συστήματος συνίσταται από ένα ή περισσότερα επεξεργαστικά στοιχεία, μνήμη, καθώς και διεπαφές εισόδου/εξόδου για επικοινωνία με το εξωτερικό περιβάλλον. Τα προγραμματιζόμενα ενσωματωμένα συστήματα διακρίνονται σε διάφορες κατηγορίες, οι οποίες εκτείνονται από

απλά συστήματα ελέγχου έως πολύπλοκες πλατφόρμες υψηλής απόδοσης. Οι πλακέτες με μικροελεγκτές (microcontroller boards) αποτελούν τη συνηθέστερη και πιο απλή μορφή ενσωματωμένων συστημάτων, προσφέροντας βασική ελεγκτική λειτουργικότητα σε ευρύ φάσμα συσκευών, από οικιακά ηλεκτρονικά μέχρι βιομηχανικό εξοπλισμό [48].

Τα συστήματα στο ίδιο τσιπ (System on a Chip, SoC) συνιστούν μια από τις πλέον προηγμένες κατηγορίες ενσωματωμένων συστημάτων, καθώς ενσωματώνουν σε ένα ενιαίο ολοκληρωμένο κύκλωμα πολλαπλές λειτουργικές μονάδες και υποσυστήματα. Τυπικά, ένα SoC περιλαμβάνει έναν ή περισσότερους επεξεργαστές (Central Processing Unit, CPU), μονάδες επεξεργασίας γραφικών (Graphics Processing Unit, GPU), επεξεργαστές ψηφιακού σήματος (Digital Signal Processor, DSP), ελεγκτές μνήμης, δικτυακές διεπαφές, καθώς και διάφορα άλλα υποσυστήματα εισόδου-εξόδου και εξειδικευμένες επιταχυντικές μονάδες. Ο συνδυασμός όλων αυτών των στοιχείων σε ένα και μόνο τσιπ προσφέρει υψηλή υπολογιστική απόδοση, ενεργειακή αποδοτικότητα και χαμηλό κόστος παραγωγής. Λόγω αυτών των χαρακτηριστικών, τα SoC κυριαρχούν σε συσκευές με αυξημένες λειτουργικές και υπολογιστικές απαιτήσεις, όπως τα smartphones και οι έξυπνες τηλεοράσεις [49].

Οι DSP αποτελούν εξειδικευμένες κατηγορίες ενσωματωμένων συστημάτων, οι οποίες έχουν βελτιστοποιηθεί αρχιτεκτονικά για την αποδοτική και ταχεία επεξεργασία τόσο αναλογικών όσο και ψηφιακών σημάτων σε πραγματικό χρόνο. Οι DSP διαθέτουν εξειδικευμένες εντολές και υπομονάδες για την εκτέλεση πολύπλοκων μαθηματικών πράξεων — όπως η γρήγορη πολλαπλασιαστική συσσώρευση, οι μετασχηματισμοί Φουριερ και τα φίλτρα Finite Impulse Response (FIR) / Infinite Impulse Response (IIR) — επιτυγχάνοντας έτσι υψηλή απόδοση σε εφαρμογές επεξεργασίας σήματος. Χάρη σε αυτές τις δυνατότητες, οι DSP χρησιμοποιούνται εκτεταμένα σε συστήματα ήχου, ραδιοφωνικούς δέκτες, συσκευές επεξεργασίας εικόνας και βίντεο, καθώς και σε εφαρμογές τηλεπικοινωνιών, όπου απαιτείται υψηλή ακρίβεια και ταχύτητα στην επεξεργασία των σημάτων [50].

Οι συστοιχίες επιτόπια προγραμματιζόμενων πυλών (Field Programmable Gate Arrays, FPGA) αποτελούν εξελιγμένες επεξεργαστικές μονάδες ενσωματωμένων συστημάτων, οι οποίες διακρίνονται για την ευελιξία τους και τη δυνατότητα υλοποίησης προσαρμοσμένων λογικών λειτουργιών και αρχιτεκτονικών απευθείας στο υλικό. Τα FPGA επιτρέπουν στον σχεδιαστή να επαναπρογραμματίσει τη δομή τους μετά την κατασκευή, διευκολύνοντας την ανάπτυξη εφαρμογών που απαιτούν υψηλή απόδοση, χαμηλό λανθάνοντα χρόνο και δυνατότητα παραμετροποίησης

ανάλογα με τις εκάστοτε ανάγκες. Χρησιμοποιούνται ευρέως σε πεδία όπου η προσαρμοστικότητα και η ταχύτητα αποτελούν κρίσιμους παράγοντες, όπως οι τηλεπικοινωνίες, η επεξεργασία σήματος, τα συστήματα ασφάλειας δικτύων και τα ενσωματωμένα συστήματα ελέγχου βιομηχανικών διεργασιών [51].

Τέλος, μια ιδιαίτερη κατηγορία, αφορά τα ενσωματωμένα συστήματα χαμηλής κατανάλωσης ενέργειας, τα οποία συναντώνται κυρίως σε φορητές συσκευές και σε εφαρμογές του Διαδικτύου των Πραγμάτων (Internet of Things, IoT). Τα συστήματα αυτά έχουν σχεδιαστεί με γνώμονα τη βελτιστοποίηση της ενεργειακής αποδοτικότητας και τη διατήρηση επαρκούς επεξεργαστικής ισχύος για την εκτέλεση εξειδικευμένων λειτουργιών. Η ανάγκη για παρατεταμένη αυτονομία και αξιόπιστη λειτουργία σε ποικίλα περιβάλλοντα καθιστά τα εν λόγω ενσωματωμένα συστήματα αναπόσπαστο στοιχείο της σύγχρονης τεχνολογίας, με εφαρμογές που εκτείνονται από έξυπνες συσκευές και αισθητήρες μέχρι συστήματα παρακολούθησης και ελέγχου σε βιομηχανικά ή οικιακά δίκτυα [52].

Στον τομέα της σύγχρονης τεχνολογίας, ο συνδυασμός των ενσωματωμένων συστημάτων με προηγμένες τεχνικές κρυπτογραφίας αποκτά διαρκώς αυξανόμενη σημασία. Σε ένα περιβάλλον όπου η ψηφιακή ασφάλεια δοκιμάζεται από εξελιγμένες επιθέσεις και πολύπλοκους κινδύνους, η ενσωμάτωση ισχυρών και αποτελεσματικών κρυπτογραφικών μηχανισμών στα ενσωματωμένα συστήματα έχει καταστεί αναγκαία [53].

Η εφαρμογή κρυπτογραφικών λύσεων σε τέτοια συστήματα απαιτεί εξειδικευμένες τεχνικές και μεθόδους για να εξασφαλιστεί η ασφάλεια των δεδομένων, ενώ παράλληλα πρέπει να διατηρείται η απαιτούμενη απόδοση των συστημάτων. Η συνεχής ανάπτυξη και βελτίωση των κρυπτογραφικών προσεγγίσεων είναι θεμελιώδης για την προστασία ενάντια σε επιθέσεις και για την διατήρηση της αξιοπιστίας των συστημάτων σε ένα ψηφιακά εξελισσόμενο περιβάλλον [54, 55].

Η κρυπτογραφία στα ενσωματωμένα συστήματα αποτελεί ένα κρίσιμο στοιχείο για τη διασφάλιση της ασφάλειας των δεδομένων τόσο κατά τη μετάδοση, οσο και για την αποθήκευση. Επιπλέον, η αυθεντικοποίηση και η ασφάλεια στην επικοινωνία μεταξύ συσκευών είναι επίσης σημαντικές πτυχές που αντιμετωπίζονται με την κρυπτογραφία. Η κρυπτογραφία περιλαμβάνει την ασφαλή ανταλλαγή κλειδιών και την χρήση πιστοποιήσεων για την εγγύηση της ασφάλειας στις επικοινωνίες [56]. Η πρόκληση στην ενσωμάτωση της κρυπτογραφίας στα ενσωματωμένα συστήματα είναι η εξισορρόπηση μεταξύ ασφάλειας και απόδοσης. Τα ενσωματωμένα

συστήματα συχνά έχουν περιορισμένους πόρους και απαιτούν υλοποιήσεις που είναι ελαφριές και αποδοτικές ως προς την κατανάλωση ενέργειας. Επίσης, η ανάγκη για συνεχή ενημέρωση και προσαρμογή των κρυπτογραφικών λύσεων λόγω της εμφάνισης νέων απειλών και τεχνολογικών προκλήσεων, αποτελεί έναν σημαντικό τομέα εστίασης στον σχεδιασμό αυτών των συστημάτων [57–59].

Τα ενσωματωμένα συστήματα βασισμένα σε FPGA έχουν αναδειχθεί ως μια πολύτιμη αρχιτεκτονική στον τομέα της κρυπτογραφίας, λόγω της υψηλής τους προσαρμοστικότητας και της δυνατότητας για ταχεία ανταπόκριση σε εξελισσόμενες απειλές ασφαλείας. Τα FPGA παρέχουν ένα δυναμικό μέσο για την υλοποίηση και βελτιστοποίηση κρυπτογραφικών αλγορίθμων, επιτρέποντας την εφαρμογή πολύπλοκων λειτουργιών σε ενσωματωμένα συστήματα με ελάχιστη κατανάλωση πόρων [60]. Τα FPGA παρέχουν ευελιξία και μπορούμε να τα προσαρμόσουμε ώστε να συμπεριλάβουν νέες κρυπτογραφικές τεχνικές και αλγορίθμους καθώς αναπτύσσονται, παρέχοντας έναν αξιόπιστο τρόπο για την διασφάλιση της ασφάλειας δεδομένων σε ενσωματωμένα περιβάλλοντα. Ειδικότερα, τα FPGA είναι ιδανικά για την υλοποίηση αλγορίθμων όπως ο Advanced Encryption Standard (AES), ο SHA, και ο Rivest Shamir Adleman (RSA), καθώς παρέχουν την απαιτούμενη υπολογιστική δύναμη και ταυτόχρονα διατηρούν την ενεργειακή απόδοση [61]. Σε συνδυασμό με την ικανότητά τους για παράλληλη επεξεργασία και υψηλή ταχύτητα επεξεργασίας, τα FPGA μπορούν να χειριστούν πολύπλοκες κρυπτογραφικές λειτουργίες, παρέχοντας αυξημένη ασφάλεια σε εφαρμογές όπως η προστασία δεδομένων, η ασφαλής επικοινωνία και η ψηφιακή υπογραφή. Με αυτόν τον τρόπο, τα FPGA βελτιώνουν την απόδοση των ενσωματωμένων συστημάτων κρυπτογραφίας, παρέχοντας μια ισχυρή και ευέλικτη λύση για την αντιμετώπιση των σύγχρονων προκλήσεων στην ασφάλεια δεδομένων [62, 63].

## 1.5 Υλοποιήσεις σε υλικό των συναρτήσεων κατακερματισμού

Για την αποτελεσματική διάκριση μεταξύ των διαφορετικών προτύπων κατακερματισμού και την αξιολόγηση της αποτελεσματικότητάς τους, είναι απαραίτητο να εφαρμοστεί μια δίκαιη και αντικειμενική σύγκριση. Μια προσέγγιση για να επιτευχθεί αυτό είναι μέσω πειραματικών υλοποιήσεων σε υλικό και της σύγκρισης αυτών με άλλες παρόμοιες εφαρμογές. Οι υλοποιήσεις που

βασίζονται σε υλικό μπορούν να προσφέρουν μετρήσεις για την ρυθμαπόδοση, την αποδοτικότητα και την καταλληλότητα διαφόρων προτύπων κατακερματισμού υπό διαφορετικές συνθήκες. Αυτή η προσέγγιση επιτρέπει μια ολοκληρωμένη αξιολόγηση των συναρτήσεων κατακερματισμού, λαμβάνοντας υπόψη παράγοντες όπως η ταχύτητα επεξεργασίας, η κατανάλωση ενέργειας και η ανθεκτικότητα σε διάφορες κρυπτογραφικές επιθέσεις. Τέτοιες συγκριτικές αναλύσεις είναι ζωτικής σημασίας για την επιλογή των κατάλληλων συναρτήσεων κατακερματισμού για συγκεκριμένες εφαρμογές και για την προώθηση του πεδίου της κρυπτογραφικής έρευνας και ανάπτυξης [64, 65].

Τα FPGA ξεχωρίζουν ως η βέλτιστη επιλογή για την εφαρμογή σε υλικό των αλγορίθμων κατακερματισμού, κυρίως λόγω της ευελιξίας και της προσαρμοστικότητάς του. Η προσαρμοστικότητα είναι ιδιαίτερα σημαντική στο ταχέως εξελισσόμενο πεδίο των συναρτήσεων κατακερματισμού κρυπτογράφησης, όπου οι αλγόριθμοι χρειάζονται συχνά ενημερώσεις ή τροποποιήσεις ως απάντηση σε αναδυόμενες απειλές ασφαλείας. Μελέτες έχουν δείξει ότι τα FPGA μπορούν να ξεπεράσουν τις παραδοσιακές μονάδες επεξεργασίας, όπως τις CPU και τις GPU όσον αφορά την ταχύτητα, καθιστώντας τα μια πιο αποτελεσματική επιλογή για επιτάχυνση σε κρυπτογραφικούς υπολογισμούς [66, 67].

Εκτός από τα FPGA, τα ολοκληρωμένα κυκλώματα ειδικής εφαρμογής (Application Specific Integrated Circuit, ASIC) χρησιμοποιούνται επίσης ευρέως σε κρυπτογραφικές εφαρμογές, ειδικά όταν η ταχύτητα επεξεργασίας είναι υψίστης σημασίας. Ένα ASIC έχει σχεδιαστεί για να εκτελεί μια συγκεκριμένη λειτουργία ή ένα σύνολο λειτουργιών, σε αντίθεση με επεξεργαστές γενικής χρήσης, όπως η CPU. Αυτή η εξειδίκευση επιτρέπει στα ASIC να εκτελούν συγκεκριμένες κρυπτογραφικές λειτουργίες, όπως ο AES [68–70], με βελτιωμένη ταχύτητα και αποτελεσματικότητα. Οι μονάδες ασφαλείας υλικού, για παράδειγμα, χρησιμοποιούν συχνά ASIC για να επιταχύνουν τις κρυπτογραφικές διαδικασίες λόγω της ανώτερης ταχύτητας και αποτελεσματικότητάς τους στο χειρισμό συγκεκριμένων εργασιών.

Πέρα από τη χρήση μεμονωμένων τύπων υλικού όπως CPU, GPU, ASIC και FPGA, υπάρχει επίσης η επιλογή ανάπτυξης των υβριδικών συστημάτων υλικού (Hybrid Hardware Systems, HHS) [71]. Ένα HHS συνδυάζει διάφορους τύπους υλικού, αξιοποιώντας τα δυνατά σημεία του καθενός για να επιτύχει μεγαλύτερη συνολική απόδοση. Ωστόσο, είναι σημαντικό να σημειωθεί ότι ενώ το HHS μπορεί να προσφέρει βελτιωμένη απόδοση και ευελιξία, συχνά συνοδεύεται από υψηλότερο κόστος. Αυτή η αύξηση της τιμής μπορεί να είναι ένας σημαντικός παράγοντας όταν

εξετάζεται η εφαρμογή κρυπτογραφικών λύσεων, ιδιαίτερα σε σενάρια όπου η σχέση κόστους-αποτελεσματικότητας είναι εξίσου σημαντική με την απόδοση. Επομένως, η επιλογή του υλικού για την υλοποίηση αλγορίθμου κατακερματισμού πρέπει να εξισορροπείται προσεκτικά μεταξύ των απαιτήσεων απόδοσης και των περιορισμών του προϋπολογισμού.

## 1.6 Μελέτη σκοπιμότητας

Στον σύγχρονο κόσμο της πληροφορικής και της δικτυακής ασφάλειας, υπάρχουν αμέτρητες εφαρμογές που ενσωματώνουν υπηρεσίες αυθεντικοποίησης, οι οποίες είναι απαραίτητες για την προστασία ενός ευρέος φάσματος δικτυακών εφαρμογών και λειτουργιών ασφαλείας. Για παράδειγμα, το IPSec [72] εξασφαλίζει ασφαλείς δικτυακές επικοινωνίες, το Public Key Infrastructure [73] παρέχει ένα σύστημα για τη διαχείριση δημόσιων και ιδιωτικών κλειδιών, ενώ το πρότυπο Secure Electronic Transactions [74] και το IEEE 802.16 [75] διασφαλίζουν για τα τοπικά και μητροπολιτικά δίκτυα την ασφάλεια των δικτυακών συναλλαγών και επικοινωνιών.

Όλες αυτές οι εφαρμογές βασίζονται στην ύπαρξη μιας αξιόπιστης διαδικασίας αυθεντικοποίησης, η οποία συνήθως περιλαμβάνει συναρτήσεις κατακερματισμού ενσωματωμένες στην υλοποίηση της εφαρμογής. Αυτές οι συναρτήσεις είναι απαραίτητες για την εξασφάλιση της επαλήθευσης της ταυτότητας των δικτυακών συμμετεχόντων και της ακεραιότητας των δεδομένων. Η ανάπτυξη και η υλοποίηση αυτών των εφαρμογών και των σχετικών μηχανισμών αυθεντικοποίησης αντιπροσωπεύουν ένα κρίσιμο στοιχείο της σύγχρονης κρυπτογραφίας, παρέχοντας ένα επίπεδο ασφάλειας και εμπιστοσύνης που είναι ζωτικής σημασίας στον σύγχρονο ψηφιακό κόσμο [76].

Επιπρόσθετα, οι συναρτήσεις κατακερματισμού αποτελούν έναν κρίσιμο παράγοντα για την ασφάλεια στα σύγχρονα δίκτυα και τις κινητές υπηρεσίες, όπως το πρωτόκολλο Secure Sockets Layer (SSL) [77]. Το SSL είναι ένα βασικό πρωτόκολλο διαδικτύου που επιτρέπει την ασφαλή και κρυπτογραφημένη επικοινωνία μεταξύ διακομιστών και πελατών στο δίκτυο. Ένας σημαντικός τομέας χρήσης των συναρτήσεων κατακερματισμού στα πλαίσια του SSL είναι η αυθεντικοποίηση των πελατών του δικτύου. Η συνηθισμένη διαδικασία για την αυθεντικοποίηση ενός πελάτη στο δίκτυο είναι να απαιτηθεί η εισαγωγή ενός κωδικού πρόσβασης που έχει προηγουμένως καταχωρηθεί στον διακομιστή. Ωστόσο, η αποθήκευση

των πραγματικών κωδικών πρόσβασης στον διακομιστή θα αποτελούσε ένα σημαντικό κίνδυνο ασφαλείας. Για να αποφευχθεί αυτό, ο διακομιστής μπορεί αντ΄ αυτού να αποθηκεύει τα hashes των κωδικών πρόσβασης. Κατά τη διαδικασία αυθεντικοποίησης, το hash του εισαγόμενου κωδικού συγκρίνεται με το αποθηκευμένο hash, επιτρέποντας την επιβεβαίωση της ταυτότητας του χρήστη χωρίς να αποκαλύπτεται ο πραγματικός κωδικός πρόσβασης [78]. Αυτή η τεχνική εξασφαλίζει ότι, ακόμα και σε περίπτωση που οι κωδικοί πρόσβασης υποκλαπούν ή εκτεθούν, οι πραγματικοί κωδικοί πρόσβασης παραμένουν ασφαλείς και άγνωστοι. Επομένως, οι συναρτήσεις κατακερματισμού αποτελούν ένα ουσιαστικό στοιχείο της ασφάλειας των δικτύων και των κινητών υπηρεσιών, αυξάνοντας την ασφάλεια και την εμπιστοσύνη στις διαδικτυακές συναλλαγές και επικοινωνίες [79].

Παράλληλα, οι συναρτήσεις κατακερματισμού αναδεικνύονται στις εφαρμογές που σχετίζονται με την αυθεντικοποίηση μηνυμάτων, όπως οι εφαρμογές Hash-based Message Authentication Code (HMAC) [80] που παράγουν τους κώδικες αυθεντικοποίησης μηνυμάτων. Οι συναρτήσεις αυτές έχουν γίνει απαραίτητες λόγω της γρήγορης εξέλιξης των προτύπων επικοινωνίας που απαιτούν αυθεντικότητα και ακεραιότητα μηνυμάτων. Εφαρμογές όπως το HMAC χρησιμοποιούν συναρτήσεις κατακερματισμού για την παραγωγή ασφαλών και αξιόπιστων κωδικών αυθεντικοποίησης, ενώ ταυτόχρονα διασφαλίζουν ότι τα μηνύματα δεν έχουν υποστεί τροποποίηση κατά την μετάδοσή τους. Αυτός ο τρόπος εφαρμογής των συναρτήσεων κατακερματισμού καταδεικνύει την ευρεία χρήση τους στα πιο δημοφιλή κρυπτογραφικά περιβάλλοντα. Επιπλέον, το πρότυπο Transport Layer Security (TLS) [81] υποδεικνύει τη χρήση των υλοποιήσεων HMAC που βασίζονται σε συναρτήσεις κατακερματισμού οι οποίες είναι ανθεκτικές σε συγκρούσεις. Η ενσωμάτωση αυτών των συναρτήσεων σε τέτοιες υλοποιήσεις προσδίδει επιπλέον ασφάλεια και ακεραιότητα στην επικοινωνία, ενισχύοντας την προστασία των δεδομένων και των μηνυμάτων σε περιβάλλοντα υψηλής ασφαλείας [82].

Τέλος, οι συναρτήσεις κατακερματισμού αποτελούν ένα απαραίτητο και καίριο στοιχείο στις σύγχρονες διαδικασίες ηλεκτρονικής ψηφοφορίας, μια τάση που γίνεται ολοένα και πιο διαδεδομένη [83]. Η χρησιμότητά τους δεν είναι μόνο στην επέκταση του εύρους των μηνυμάτων που μπορούν να υπογραφούν ψηφιακά, αλλά και στον ρόλο που παίζουν στην ενίσχυση της ασφάλειας του συστήματος. Οι συναρτήσεις κατακερματισμού στις ψηφιακές υπογραφές μετατρέπουν τα μηνύματα οποιουδήποτε μήκους σε συμπαγείς και διαχειρίσιμες τιμές κατακερματισμού, οι

οποίες στη συνέχεια υπογράφονται ψηφιακά. Αυτή η τεχνική εξασφαλίζει ότι τα μηνύματα διατηρούν την ακεραιότητα και την αυθεντικότητά τους καθ΄ όλη τη διάρκεια της διαδικασίας της ψηφοφορίας, μια διαδικασία που απαιτεί υψηλά επίπεδα ασφάλειας και εμπιστοσύνης.

Επιπρόσθετα, οι συναρτήσεις κατακερματισμού διαδραματίζουν σημαντικό ρόλο στην προστασία και την επαλήθευση των ψηφιακών υπογραφών, διασφαλίζοντας ότι οι υπογραφές που χρησιμοποιούνται είναι έγκυρες και δεν έχουν υποστεί καμία τροποποίηση. Η συνεισφορά των συναρτήσεων κατακερματισμού στις διαδικασίες αυτές είναι αποφασιστική για την αξιοπιστία και την ασφάλεια των συστημάτων ηλεκτρονικής ψηφοφορίας και άλλων σχετικών εφαρμογών, παρέχοντας μια ασφαλή και διαφανή μέθοδο για τη διασφάλιση της ακεραιότητας των ψηφιακά διακινούμενων δεδομένων [84].

Έτσι, η σημασία των συναρτήσεων κατακερματισμού στον κόσμο των δικτυακών εφαρμογών και της κυβερνοασφάλειας είναι αδιαμφισβήτητη και αυτό καθίσταται εμφανές μέσω της αυξανόμενης χρήσης τους από πλείστους χρήστες και πελάτες. Με την επέκταση των εφαρμογών που ενσωματώνουν τέτοιες λειτουργίες, γίνεται αναγκαία η αύξηση της απόδοσης, ιδιαίτερα σε ό,τι αφορά τους διακομιστές που υποστηρίζουν αυτές τις υπηρεσίες. Ο λόγος είναι ότι το κρυπτογραφικό σύστημα, και ειδικά ο διακομιστής, πρέπει να είναι σε θέση να επεξεργάζεται και να αποκρίνεται άμεσα στα αιτήματα που προέρχονται από όλους τους χρήστες [85].

Στα περισσότερα σύγχρονα κρυπτογραφικά σχήματα, η απόδοση της ενσωματωμένης λειτουργίας κατακερματισμού είναι καθοριστική για την απόδοση του συνολικού συστήματος ασφάλειας. Επομένως, οι αποδοτικές υλοποιήσεις των συναρτήσεων κατακερματισμού συμβάλλουν σημαντικά στην ομαλή και αποδοτική επικοινωνία μεταξύ αυτών των εφαρμογών. Σε σενάρια όπου οι ρυθμοί μετάδοσης και λήψης είναι ιδιαίτερα υψηλοί, η καθυστέρηση στον υπολογισμό των ψηφιακών υπογραφών ή των τιμών κατακερματισμού μπορεί να οδηγήσει σε σημαντική υποβάθμιση της ποιότητας του δικτύου και της υπηρεσίας [86]. Είναι επομένως ζωτικής σημασίας οι διακομιστές και τα συστήματα που χρησιμοποιούν τέτοιες συναρτήσεις να είναι υψηλής απόδοσης, εξασφαλίζοντας την άμεση και αποδοτική επεξεργασία των αιτημάτων των χρηστών, διατηρώντας ταυτόχρονα την ασφάλεια και την ακεραιότητα των δεδομένων [87].

Η αναζήτηση για αυξημένη απόδοση στις συναρτήσεις κατακερματισμού έχει οδηγήσει σε πρακτικές όπου χρησιμοποιείται επιπλέον επαναλαμβανόμενο υλικό

για την επίτευξη της επιθυμητής αποδοτικότητας (π.χ. παραλληλοποιώντας τους υπολογισμούς). Παρόλα αυτά, αυτές οι υλοποιήσεις συχνά δεν επιτυγχάνουν τη βέλτιστη μέγιστη συχνότητα λειτουργίας και την κορυφαία απόδοση, κυρίως επειδή οι συγκεκριμένοι παράμετροι δεν εστιάζουν στην βελτίωση του κρίσιμου μονοπατιού από σχεδιαστική και αλγοριθμική άποψη [88]. Η προσέγγιση αυτή, αν και μπορεί να προσφέρει ορισμένα πλεονεκτήματα σε όρους απόδοσης, συχνά οδηγεί σε μη αποδοτική χρήση των πόρων και αυξημένο κόστος στην υλοποίηση των συστημάτων. Επιπλέον, οι τεχνικές αυτές μπορεί να μην είναι αρκετά ευέλικτες για να προσαρμοστούν στις γρήγορα μεταβαλλόμενες απαιτήσεις της κρυπτογραφίας και της ασφάλειας δεδομένων. Έτσι, οι μηχανικοί και οι σχεδιαστές αναζητούν εναλλακτικές λύσεις που θα ενσωματώνουν καινοτόμες σχεδιαστικές και αλγοριθμικές προσεγγίσεις, προκειμένου να βελτιστοποιήσουν το κρίσιμο μονοπάτι και να επιτύχουν υψηλότερες αποδόσεις χωρίς υπερβολικό κόστος ή σπατάλη πόρων. Αυτή η στροφή προς την βελτιστοποίηση μπορεί να αποτελέσει το κλειδί για την ανάπτυξη πιο αποδοτικών, ευέλικτων και οικονομικά αποδοτικών συστημάτων ασφαλείας στο μέλλον [89, 90].

## 1.7 Ερευνητικοί στόχοι και ερωτήματα

Η αναγνώριση των υφιστάμενων προκλήσεων στις υλοποιήσεις των συναρτήσεων κατακερματισμού, και ειδικότερα στη συνάρτηση Secure Hash Algorithm 3 (SHA-3), έχει δημιουργήσει ισχυρό κίνητρο για τον σχεδιασμό και την ανάπτυξη νέων μεθοδολογιών. Η παρούσα προσέγγιση επικεντρώνεται στη βελτιστοποίηση και επιτάχυνση του κρίσιμου μονοπατιού της συνάρτησης κατακερματισμού, με απώτερο στόχο την αύξηση της ρυθμαπόδοσης και της αποδοτικότητας. Η προτεινόμενη μεθοδολογία στοχεύει να απαντήσει στις σύγχρονες προκλήσεις της κρυπτογραφίας, καλύπτοντας την ανάγκη για ταχύτερη επεξεργασία δεδομένων και ταυτόχρονη μείωση της κατανάλωσης πόρων, με έμφαση στον αρχιτεκτονικό και αλγοριθμικό σχεδιασμό.

Η επιτάχυνση της ρυθμαπόδοσης συμβάλλει στην υλοποίηση ταχύτερων και πιο αποδοτικών συστημάτων, ενώ η ενίσχυση της αποδοτικότητας οδηγεί σε μείωση του κόστους και της ενεργειακής κατανάλωσης, διασφαλίζοντας παράλληλα την ακεραιότητα και την ασφάλεια των δεδομένων. Η εφαρμογή της προτεινόμενης μεθοδολογίας αναμένεται να αποτελέσει ουσιαστική πρόοδο στην προσαρμογή

των κρυπτογραφικών συστημάτων στις αυξανόμενες και διαρκώς εξελισσόμενες απαιτήσεις της ψηφιακής εποχής.

Οι ερευνητικοί στόχοι της παρούσας διατριβής είναι:

1. Η ανασκόπηση των υφιστάμενων και βέλτιστων υλοποιήσεων του αλγόριθμου SHA-3 σε FPGA. Αυτή η ανασκόπηση στοχεύει στην παρουσίαση των διαφόρων στρατηγικών και τεχνικών που έχουν χρησιμοποιηθεί, εξετάζοντας τις αρχιτεκτονικές τους, τη ρυθμαπόδοση, και τις προκλήσεις που αντιμετωπίζουν.

2. Έρευνα και μελέτη των προηγμένων μεθόδων για την επιτάχυνση και την βελτιστοποίηση της απόδοσης του κρίσιμου μονοπατιού της συνάρτησης κατακερματισμού SHA-3 για απαιτητικές κρυπτογραφικές εφαρμογές.

3. Εστίαση στην ανάπτυξη στρατηγικών επιτάχυνσης και μεθόδων βελτιστοποίησης του κρίσιμου μονοπατιού της συνάρτησης κατακερματισμού SHA-3. Αυτό περιλαμβάνει την υλοποίηση τεχνικών που σχετίζονται με την ταχύτητα και την αποδοτικότητα της επεξεργασίας. Στόχος είναι η δημιουργία ενός συστήματος που να επιτυγχάνει υψηλή ρυθμαπόδοση κατά την επεξεργασία και το χειρισμό των κρυπτογραφικών δεδομένων.

4. Ανάπτυξη υλοποιήσεων της συνάρτησης κατακερματισμού SHA-3 σε FPGA, με στόχο τη μέγιστη επιτάχυνση, την υψηλή ρυθμαπόδοση και/ή αποδοτικότητα. Αυτό περιλαμβάνει τη λεπτομερή αξιολόγηση και σύγκριση των προτεινόμενων υλοποιήσεων με άλλες υπάρχουσες λύσεις, προσφέροντας ένα συγκριτικό πλαίσιο για την επιλογή της βέλτιστης προσέγγισης.

Τα ερευνητικά ερωτήματα της παρούσας διατριβής είναι:

1. Ποιες είναι οι διάφορες υλοποιήσεις του αλγόριθμου SHA-3 σε FPGA και πώς αξιολογούνται ως προς την απόδοση, την αρχιτεκτονική, την επιτάχυνση και τις προκλήσεις που αντιμετωπίζουν;

2. Ποιες μέθοδοι και τεχνικές επιτάχυνσης μπορούν να βελτιώσουν σημαντικά την απόδοση του κρίσιμου μονοπατιού της συνάρτησης κατακερματισμού SHA-3 σε περιβάλλοντα FPGA;

3. Πώς συγκρίνονται οι νέες υλοποιήσεις του SHA-3 σε FPGA από άποψη ρυθμαπόδοσης, αποδοτικότητας και επιτάχυνσης με άλλες υφιστάμενες λύσεις; Ποια είναι τα πλεονεκτήματα και οι περιορισμοί τους σε σχέση με τις υπάρχουσες υλοποιήσεις;

Η παρούσα διατριβή αποσκοπεί στη σε βάθος κατανόηση των κρυπτογραφικών τεχνολογιών που σχετίζονται με το κρίσιμο μονοπάτι της συνάρτησης κατακερματισμού SHA-3. Ιδιαίτερη έμφαση δίνεται στην ανάπτυξη, υλοποίηση και επιτάχυνση προηγμένων μεθοδολογιών βελτιστοποίησης της εν λόγω συνάρτησης σε περιβάλλοντα FPGA. Τα FPGA, χάρη στην ευελιξία και την υψηλή επεξεργαστική τους ικανότητα, αναδεικνύονται ως ζωτικής σημασίας εργαλεία για την υλοποίηση, την επιτάχυνση και την αποδοτική εκτέλεση των προτεινόμενων βελτιστοποιήσεων. Η φιλοδοξία της διατριβής είναι να αναδείξει τον κρίσιμο ρόλο των FPGA στην ανάπτυξη επιταχυνόμενων κρυπτογραφικών λύσεων υψηλής ρυθμαπόδοσης και αποδοτικότητας, οι οποίες μπορούν να εφαρμοστούν ευρέως στο σύγχρονο ψηφιακό περιβάλλον.

## 1.8 Πλάνο έρευνας

Η παρούσα ενότητα αποσκοπεί στην ανάλυση της μεθοδολογίας που υιοθετήθηκε για την επίτευξη των ερευνητικών στόχων της διατριβής. Η μεθοδολογική δομή της έρευνας χαρακτηρίζεται από τέσσερις βασικές φάσεις:

1. Στην εκτενή βιβλιογραφική ανασκόπηση,
2. Στον εντοπισμό ερευνητικών κενών,
3. Στην ανάπτυξη της προτεινόμενης προσέγγισης,
4. Στην επικύρωση μέσω πειραμάτων.

Η αρχική φάση της ερευνητικής διαδικασίας της παρούσας διατριβής ήταν η συστηματική και εκτενής βιβλιογραφική ανασκόπηση. Στόχος αυτής της φάσης ήταν η έρευνα και η μελέτη της υπάρχουσας βιβλιογραφίας σχετικά με τις τεχνικές επιτάχυνσης και βελτιστοποίησης του κρίσιμου μονοπατιού της συνάρτησης κατακερματισμού SHA-3, ειδικά σε περιβάλλοντα FPGA. Για

τη συλλογή των απαραίτητων πληροφοριών, πραγματοποιήθηκε εστιασμένη αναζήτηση σε αναγνωρισμένες βάσεις δεδομένων και ακαδημαϊκές πηγές, όπως το IEEE Xplore, το ScienceDirect και το Google Scholar. Από τις πηγές αυτές συλλέχθηκαν και αναλύθηκαν πολυάριθμες σχετικές επιστημονικές δημοσιεύσεις σε περιοδικά και πρακτικά συνεδρίων, με χρήση συγκεκριμένων κλειδιών αναζήτησης.

Τα ερευνητικά κλειδιά αναζήτησης περιελάμβαναν:

1. SHA-3 optimization,
2. Keccak implementation,
3. critical path analysis of the SHA-3,
4. hardware acceleration for SHA-3 algorithm,
5. high-performance computing on FPGA of the Keccak algorithm,
6. efficiency in FPGA designs of the Keccak,
7. FPGA implementation of the SHA-3, και
8. critical path function efficiency of the SHA-3.

Η αξιολόγηση των συλλεχθέντων πηγών πραγματοποιήθηκε με βάση συγκεκριμένα κριτήρια, όπως η ακαδημαϊκή εγκυρότητα και το κύρος των επιστημονικών περιοδικών και συνεδρίων στα οποία δημοσιεύθηκαν οι εργασίες, η επιρροή τους στον επιστημονικό χώρο, καθώς και η συχνότητα αναφοράς τους σε συναφείς προηγούμενες μελέτες. Ιδιαίτερη έμφαση δόθηκε στην επιλογή πηγών που παρουσιάζουν υψηλό δείκτη απήχησης και αναγνωρισιμότητας στην ερευνητική κοινότητα.

Η επισκόπηση της βιβλιογραφίας σχετικά με την επιτάχυνση του κρίσιμου μονοπατιού στη συνάρτηση κατακερματισμού SHA-3 σε περιβάλλοντα FPGA αποτέλεσε ουσιώδη φάση της ερευνητικής διαδικασίας της διατριβής. Μέσα από αυτήν τη βιβλιογραφική ανασκόπηση, αναδείχθηκαν σημαντικά κενά και περιορισμοί στις υφιστάμενες προσεγγίσεις επιτάχυνσης. Από την ανάλυση προέκυψε ότι, παρά τα αξιοσημείωτα βήματα προόδου στη ρυθμαπόδοση του SHA-3 σε συστήματα FPGA, η πρόκληση της επίτευξης της βέλτιστης επιτάχυνσης και ρυθμαπόδοσης στο κρίσιμο μονοπάτι της επεξεργασίας δεν έχει ακόμη επιλυθεί

επαρκώς. Η αναγνώριση αυτού του ερευνητικού κενού αποτέλεσε το κίνητρο για την υλοποίηση νέων, βελτιωμένων προσεγγίσεων και μεθοδολογιών.

Μετά την πλήρη ανάλυση και κατανόηση της τρέχουσας κατάστασης, διαμορφώθηκε μια νέα ερευνητική προσέγγιση με στόχο τη βελτιστοποίηση και επιτάχυνση της ρυθμαπόδοσης του κρίσιμου μονοπατιού της συνάρτησης κατακερματισμού SHA-3 σε συστήματα FPGA. Για την επικύρωση και αξιολόγηση της αποτελεσματικότητας της προτεινόμενης προσέγγισης, υλοποιήθηκε μια σειρά από πειραματικές δοκιμές. Στο πλαίσιο των πειραμάτων εφαρμόστηκαν τόσο ποσοτικές όσο και ποιοτικές μέθοδοι αξιολόγησης, εξετάζοντας παραμέτρους όπως η επιτάχυνση, η ρυθμαπόδοση και η αποδοτικότητα των προτεινόμενων λύσεων. Τα αποτελέσματα των πειραμάτων αναλύθηκαν λεπτομερώς και συγκρίθηκαν με άλλες υφιστάμενες υλοποιήσεις, προκειμένου να εκτιμηθεί η συνολική απόδοση της νέας προσέγγισης.

## 1.9 Διάρθρωση διατριβής

Η παρούσα διατριβή επιχειρεί να συμβάλει στον τομέα της κρυπτογραφίας μέσω της εξέτασης, ανάπτυξης και εφαρμογής μεθοδολογιών επιτάχυνσης για τη βελτίωση της σχεδίασης και υλοποίησης της λειτουργίας του κρίσιμου μονοπατιού για τον αλγόριθμο κατακερματισμού SHA-3 σε περιβάλλον FPGA. Κεντρικός στόχος της είναι η επιτάχυνση της ρυθμαπόδοσης της συγκεκριμένης κρυπτογραφικής λειτουργίας, αξιοποιώντας τεχνικές που οδηγούν στην ταυτόχρονη βελτίωση της αποδοτικότητας και εξοικονόμησης πόρων. Μέσω ενός ευρέος φάσματος πειραματικών δοκιμών και αναλυτικών μελετών, η διατριβή αυτή επιδιώκει να καθοδηγήσει τις μελλοντικές προσπάθειες στον σχεδιασμό αποδοτικών κρυπτογραφικών συστημάτων, ενισχύοντας την ασφάλεια και την απόδοση σε ψηφιακά συστήματα που εξαρτώνται από την αξιοπιστία και την ακεραιότητα της επεξεργασίας δεδομένων.

Το Κεφάλαιο 2 παρέχει μια συνοπτική αλλά ολοκληρωμένη εισαγωγή στις βασικές αρχές και τις κύριες έννοιες που διαμορφώνουν την οικογένεια των αλγορίθμων κατακερματισμού. Αναλύοντας την ιστορική εξέλιξη και τη συνεχή επέκταση των αλγορίθμων κατακερματισμού, το κεφάλαιο αυτό επιχειρεί να καταδείξει τα σημαντικά χαρακτηριστικά και τις διάφορες τους. Παράλληλα, δίνεται έμφαση στις τεχνικές και στρατηγικές προκλήσεις που ανακύπτουν κατά τη διαδικασία

σχεδιασμού και ανάπτυξης των συστημάτων που ενσωματώνουν το κρίσιμο μονοπάτι αυτών των αλγορίθμων, εξετάζοντας ταυτόχρονα τους περιορισμούς και τις δυνατότητες που προσφέρουν στον τομέα της κρυπτογραφίας.

Στο Κεφάλαιο 3 προτείνεται μια νέα τεχνική επιτάχυνσης και βελτιστοποίησης που βασίζεται στη διασωλήνωση (pipelining). Ειδικότερα, αναλύεται μια μέθοδος η οποία τοποθετεί τον πρόσθετο καταχωρητή μετά το βήμα $\theta$ $(theta)$ στη συνάρτηση $f$, με στόχο την επιτάχυνση της επεξεργασίας και τη βελτιστοποίηση του κρίσιμου μονοπατιού. Παράλληλα, πραγματοποιήθηκε εκτεταμένη αξιολόγηση και ανάλυση, συγκρίνοντας την περιοχή, τη ρυθμαπόδοση, τη συχνότητα, την αποδοτικότητα και κυρίως τον βαθμό επιτάχυνσης της προτεινόμενης αρχιτεκτονικής με αντίστοιχες μεθόδους που έχουν παρουσιαστεί στη διεθνή βιβλιογραφία.

Στο Κεφάλαιο 4 προτείνεται μια νέα τεχνική επιτάχυνσης και βελτιστοποίησης που βασίζεται στη μέθοδο του ξετυλίγματος (unrolling). Σε αυτό το κεφάλαιο αναλύεται η συγκεκριμένη μέθοδος, η οποία στοχεύει στη σημαντική επιτάχυνση της διαδικασίας, μειώνοντας τον συνολικό αριθμό των κύκλων ρολογιού που απαιτούνται για την απόκτηση του αποτελέσματος της συνάρτησης κατακερματισμού. Παράλληλα, διεξάγεται μια σειρά από λεπτομερείς συγκρίσεις της προτεινόμενης αρχιτεκτονικής, εστιάζοντας στους απαιτούμενους υλικούς πόρους, την ταχύτητα, τη συχνότητα λειτουργίας, την αποδοτικότητα και κυρίως στον βαθμό επιτάχυνσης σε σχέση με άλλες σχετικές μεθόδους που έχουν δημοσιευθεί στη διεθνή επιστημονική βιβλιογραφία.

Στο Κεφάλαιο 5 παρουσιάζεται μια προηγμένη τεχνική επιτάχυνσης και βελτιστοποίησης, η οποία συνδυάζει τις στρατηγικές της διασωλήνωσης (pipelining) και του ξετυλίγματος (unrolling), με στόχο τη βελτίωση της αποδοτικότητας του υλικού. Η προτεινόμενη μέθοδος στοχεύει στην επιτάχυνση της επεξεργασίας, μειώνοντας τους απαιτούμενους κύκλους ρολογιού ανά λειτουργία και προσφέροντας ταυτόχρονα υψηλότερη ταχύτητα και αποδοτικότητα στη διαχείριση των δεδομένων. Επιπρόσθετα, διεξάγεται μια εκτενής συγκριτική ανάλυση με άλλες σύγχρονες μεθόδους που έχουν δημοσιευθεί πρόσφατα, αναδεικνύοντας τα πλεονεκτήματα και τις διαφοροποιήσεις της προτεινόμενης προσέγγισης ως προς την επιτάχυνση, την απόδοση και την οικονομία πόρων, σε σχέση με τις υπάρχουσες λύσεις.

Τέλος, στο Κεφάλαιο 6, καταπιανόμαστε με την εξερεύνηση των προοπτικών για μελλοντικές ερευνητικές δραστηριότητες, διατυπώνοντας ειδικές συστάσεις και

προσδιορίζοντας δυνητικές κατευθύνσεις για την επέκταση των ευρημάτων της παρούσας μελέτης. Επιχειρείται μια λεπτομερής ανάλυση των πιθανών εμποδίων και προκλήσεων που μπορεί να αντιμετωπίσουν οι ερευνητές κατά τη διάρκεια των μελλοντικών τους ενασχολήσεων. Αυτή η αναλυτική προσέγγιση καταλήγει στη διατύπωση των κεντρικών συμπερασμάτων που εξήχθησαν από την εν λόγω έρευνα, επισημαίνοντας την προσφορά της στην περαιτέρω εξέλιξη και κατανόηση του αλγορίθμου ασφαλούς κατακερματισμού SHA-3. Μέσω αυτής της διαδικασίας, στοχεύουμε στη δημιουργία μιας γέφυρας μεταξύ των υφιστάμενων ευρημάτων και των μελλοντικών ερευνητικών προσπαθειών, δίνοντας το έναυσμα για την ανάπτυξη νέων ιδεών και προσεγγίσεων στον ευρύτερο τομέα της κρυπτογραφίας και της ασφάλειας δεδομένων.

# Κεφάλαιο 2

# Θεωρητικό υπόβαθρο

Αυτό το κεφάλαιο εμβαθύνει στις θεμελιώδεις αρχές της οικογένειας αλγορίθμων ασφαλούς κατακερματισμού. Αρχικά, η συζήτηση εστιάζει στον αλγόριθμο SHA-1, με ανάλυση της δομής και της λειτουργίας του. Στη συνέχεια, εξετάζονται οι αλγόριθμοι SHA-2 και SHA-3, με έμφαση στις τεχνικές και αρχιτεκτονικές τους διαφορές και ομοιότητες. Η ενότητα αυτή παρέχει το απαραίτητο θεωρητικό υπόβαθρο για τους αλγορίθμους SHA, δημιουργώντας τις προϋποθέσεις για βαθύτερη και πιο εξειδικευμένη μελέτη στα επόμενα κεφάλαια της διατριβής.

## 2.1 Οικογένεια αλγορίθμων ασφαλούς κατακερματισμού

Η οικογένεια αλγορίθμων ασφαλούς κατακερματισμού SHA δέχεται ένα μήνυμα αυθαίρετου μεγέθους και, μέσω συγκεκριμένων υπολογιστικών διαδικασιών, παράγει την αντίστοιχη τιμή κατακερματισμού, όπως φαίνεται στην Εξίσωση (2.1):

$$h = H(M) \tag{2.1}$$

όπου $M$ είναι το εισερχόμενο μήνυμα και $h$ η παραγόμενη τιμή κατακερματισμού μέσω του αλγορίθμου $H$. Στον Πίνακα 2.1 παρουσιάζονται συγκεντρωτικά οι κύριες παράμετροι των διαφόρων μελών της οικογένειας SHA.

Ο SHA-1 δέχεται μηνύματα μεγέθους μικρότερου των $2^{64}$ bit, τα χωρίζει σε ίσα μπλοκ των 512 bit, και μετά από 80 βήματα εσωτερικών υπολογισμών, παράγει κατακερματισμό 160 bit. Η οικογένεια SHA-2 περιλαμβάνει παραλλαγές με κατακερματισμό 224, 256, 384 και 512 bit. Οι παραλλαγές 224 και 256 bit χρησιμοποιούν μπλοκ 512 bit (έως $2^{64}$ bit μήνυμα), ενώ οι 384 και 512 bit χρησιμοποιούν μπλοκ 1024 bit (έως $2^{128}$ bit μήνυμα), με 64 και 80 βήματα εσωτερικών υπολογισμών αντίστοιχα.

Ο SHA-3, σε αντίθεση με τους προκατόχους του, επιτρέπει επεξεργασία μηνυμάτων οποιουδήποτε μεγέθους, διαθέτοντας παραλλαγές 224, 256, 384 και 512 bit, και χρησιμοποιώντας αντίστοιχα μπλοκ μεγέθους 1152, 1088, 832 και 576 bit, με 24 βήματα εσωτερικών υπολογισμών. Αυτή η ευελιξία τον καθιστά ιδιαίτερα κατάλληλο για σύγχρονες εφαρμογές.

Η σύγκριση αυτών των χαρακτηριστικών είναι σημαντική, καθώς κάθε αλγόριθμος SHA έχει σχεδιαστεί για να εξυπηρετεί διαφορετικές ανάγκες ως προς την ασφάλεια, την απόδοση και τη διαχείριση δεδομένων σε διάφορα υπολογιστικά περιβάλλοντα.

Πίνακας 2.1: Οικογένεια αλγορίθμων SHA.

| **Αλγόριθμος** | **Έξοδος σε (bit)** | **Μέγεθος μπλοκ (bit)** | **Μέγιστο μέγεθος μηνύματος (bit)** | **Βήματα εσωτερικών υπολογισμών** |
|---|---|---|---|---|
| SHA-1 | 160 | 512 | $2^{64}-1$ | 80 |
| SHA-2 | 224 | 512 | $2^{64}-1$ | 64 |
| | 256 | 512 | $2^{64}-1$ | 64 |
| | 384 | 1024 | $2^{128}-1$ | 80 |
| | 512 | 1024 | $2^{128}-1$ | 80 |
| SHA-3 | 224 | 1152 | Οτιδήποτε | 24 |
| | 256 | 1088 | Οτιδήποτε | 24 |
| | 384 | 832 | Οτιδήποτε | 24 |
| | 512 | 576 | Οτιδήποτε | 24 |

Για να θεωρηθεί ασφαλής ένας αλγόριθμος κατακερματισμού, απαιτείται να πληροί τρεις θεμελιώδεις κρυπτογραφικές ιδιότητες [91, 92]:

1. Αντίσταση προεικόνας (preimage resistance): Αναφέρεται στη δυσκολία ανεύρεσης του αρχικού μηνύματος, δεδομένης της τιμής του κατακερματισμού. Δηλαδή, ενώ είναι υπολογιστικά εύκολο να παραχθεί ένας κατακερματισμός από ένα δεδομένο μήνυμα, είναι εξαιρετικά δύσκολο να ανακτηθεί το αρχικό μήνυμα από τον κατακερματισμό.

2. Αντίσταση σε δεύτερη προεικόνα (second preimage resistance): Αφορά τη δυσκολία εντοπισμού ενός δεύτερου μηνύματος, διαφορετικού από το αρχικό, που να παράγει τον ίδιο κατακερματισμό με ένα δοθέν μήνυμα. Η ιδιότητα αυτή διασφαλίζει ότι δεν είναι εφικτό να βρεθεί ένα άλλο μήνυμα με την ίδια τιμή κατακερματισμού.

3. Αντοχή σε σύγκρουση (collision resistance): Αναφέρεται στη δυσκολία εύρεσης δύο διαφορετικών μηνυμάτων που παράγουν τον ίδιο κατακερματισμό. Η ιδιότητα αυτή είναι κρίσιμη για την πρόληψη επιθέσεων σύγκρουσης, όπου ένας κακόβουλος μπορεί να δημιουργήσει διαφορετικά μηνύματα με τον ίδιο κατακερματισμό για να παραπλανήσει ή να εξαπατήσει το σύστημα ή τους χρήστες.

## 2.2 SHA-1

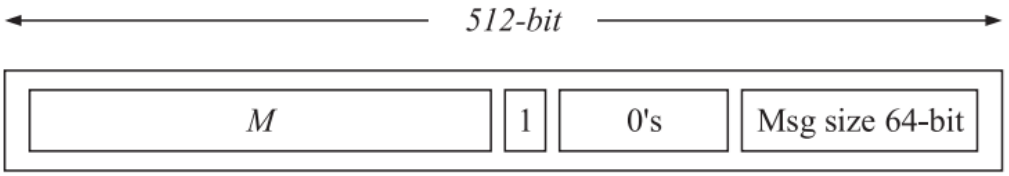
Ο SHA-1, ως παραλλαγή του αρχικού αλγορίθμου κατακερματισμού MD5, αποτελεί μία από τις πρώτες απόπειρες ενίσχυσης της κρυπτογραφικής ασφάλειας στον τομέα των συναρτήσεων κατακερματισμού. Παρά το γεγονός ότι το 2017 οι Stevens et al. [93] ανακοίνωσαν επιτυχημένη επίθεση σύγκρουσης εναντίον του SHA-1, ο αλγόριθμος αυτός εξακολουθεί να χρησιμοποιείται ευρέως σε πληθώρα εφαρμογών και συστημάτων, κυρίως για λόγους συμβατότητας.

Ο SHA-1 βασίζεται στη δομή Merkle-Damgård (MD), μια μεθοδολογία που υιοθετείται ευρέως στους αλγορίθμους κατακερματισμού. Παρέχοντας κατακερματισμό εξόδου 160 bit, ο SHA-1 υποβάλλει το εισερχόμενο μήνυμα σε μια ακολουθία βημάτων συμπίεσης και εσωτερικών υπολογισμών πριν την παραγωγή της τελικής τιμής κατακερματισμού. Η διαδικασία αυτή είναι θεμελιώδης για τη διασφάλιση της ακεραιότητας των δεδομένων και την αντίσταση σε προσπάθειες ανακατασκευής της αρχικής πληροφορίας από τον κατακερματισμό [94].

### 2.2.1 Βασικά στοιχεία

Ο αλγόριθμος SHA-1 είναι ένας κρυπτογραφικός αλγόριθμος κατακερματισμού, που χρησιμοποιείται για την παραγωγή ενός αποτελέσματος σταθερού μεγέθους (160-bit hash) από δεδομένα μεταβλητού μεγέθους. Ο SHA-1 χειρίζεται μηνύματα με μέγιστο

μέγεθος λιγότερο από $2^{128}$ bit. Η διαδικασία αυτή αρχίζει με την προεπεξεργασία του μηνύματος $M$, όπου προστίθεται ένα bit "1" και ένας αριθμός bit "0" έως ότου το μήκος του μηνύματος φτάσει το 448 mod 512 = 448. Στη συνέχεια, το μήκος του αρχικού μηνύματος προστίθεται στο τέλος του μηνύματος ως μια τιμή 64-bit με big-endian σειρά, όπως φαίνεται στο Σχήμα 2.1.




Σχήμα 2.1: Μηχανισμός συμπλήρωσης μηνυμάτων του SHA-1

Το επεξεργασμένο μήνυμα στη συνέχεια διαιρείται σε μπλοκ των 512 bit. Κάθε μπλοκ υποβάλλεται στη λειτουργία συμπίεσης του SHA-1, η οποία αποτελείται από 80 διαδοχικά βήματα, οργανωμένα σε τέσσερις γύρους των 20 βημάτων ο καθένας. Κάθε βήμα ($t$) της λειτουργίας συμπίεσης αξιοποιεί πράξεις όπως αρθρωτή πρόσθεση, αριστερή κυκλική μετατόπιση (left rotation), στρογγυλή συνάρτηση (round function) και στρογγυλοποιημένη σταθερά (round constant), όπως φαίνεται στον Πίνακα 2.2. Οι στρογγυλές συναρτήσεις και οι σταθερές διαφέρουν σε κάθε γύρο, προσδίδοντας στον SHA-1 τα απαραίτητα χαρακτηριστικά κρυπτογραφικής ασφάλειας.

Πίνακας 2.2: SHA-1 γύροι, στρογγυλές συναρτήσεις, βήματα και σταθερές

| **Γύρος** | **Βήματα** | **Στρογγυλές συναρτήσεις** $F(B, C, D)$ | **Στρογγυλοποιημένες σταθερές** $K_t$ |
|---|---|---|---|
| 1 | 0–19 | $(B \wedge C) \vee (\neg B \wedge D)$ | 0x5A827999 |
| 2 | 20–39 | $B \oplus C \oplus D$ | 0x6ED9EBA1 |
| 3 | 40–59 | $(B \wedge C) \vee (B \wedge D) \vee (C \wedge D)$ | 0x8F1BBCDC |
| 4 | 60–79 | $B \oplus C \oplus D$ | 0xCA62C1D6 |

Κάθε μπλοκ μηνύματος διαιρείται σε 16 διαδοχικές λέξεις των 32 bit η καθεμία, από $M_0$ έως $M_{15}$. Στη συνέχεια, οι λέξεις αυτές επεκτείνονται σε 80 λέξεις $W_t$ (για $t = 0, \ldots, 79$) χρησιμοποιώντας την Εξίσωση (2.2). Στην εξίσωση αυτή, η λειτουργία cyclic shift left (csl) 1 αναφέρεται σε κυκλική αριστερή μετατόπιση (ολίσθηση) κατά ένα bit.

$$W_t = \begin{cases} M_t & , \ 0 \leq t \leq 15 \\ (W_{t-16} \ XOR \ W_{t-14} \ XOR \ W_{t-8} \ XOR \ W_{t-3}) \text{ csl } 1 & , \ 16 \leq t \leq 79 \end{cases} \tag{2.2}$$

Ο αλγόριθμος SHA-1 χρησιμοποιεί πέντε μεταβλητές κατάστασης, με ενδείξεις A, B, C, D και E. Η διαδικασία επεξεργασίας περιλαμβάνει τέσσερις γύρους των 20 επαναλήψεων ο καθένας, με στόχο τη σημαντική τροποποίηση των τιμών των A, B, C, D και E, έτσι ώστε να μην διατηρείται συσχέτιση με τις αρχικές τους τιμές. Οι μεταβλητές αυτές ενημερώνονται μετά από κάθε βήμα σύμφωνα με τις Εξισώσεις (2.3)-(2.7).

$$\begin{aligned} A_t = & left\ rotation\ (X)\ by\ 5\ bit(A_{t-1}) \\ & modular\ addition\ F_t(B_{t-1}, C_{t-1}, D_{t-1}) \\ & modular\ addition\ E_{t-1} \\ & modular\ addition\ W_t \\ & modular\ addition\ K_t \end{aligned} \tag{2.3}$$

$$B_t = A_{t-1} \tag{2.4}$$

$$C_t = left\ rotation\ (X)\ by\ 30\ bit(B_{t-1}) \tag{2.5}$$

$$D_t = C_{t-1} \tag{2.6}$$

$$E_t = D_{t-1} \tag{2.7}$$

Μετά την επεξεργασία όλων των μπλοκ του μηνύματος, οι τελικές τιμές των πέντε μεταβλητών κατάστασης A, B, C, D και E συνενώνονται για να παραχθεί ο τελικός κατακερματισμός, ο οποίος αποτελεί το hash του αρχικού μηνύματος. Το παραγόμενο hash είναι μοναδικό για το εκάστοτε μήνυμα και η ανάκτηση του αρχικού μηνύματος από την τιμή κατακερματισμού είναι υπολογιστικά ανέφικτη, ιδιότητα που καθιστά τον SHA-1 κατάλληλο για εφαρμογές ασφάλειας, όπως οι ηλεκτρονικές υπογραφές και ο έλεγχος ακεραιότητας δεδομένων. Ωστόσο, εξαιτίας των πρόσφατων επιθέσεων σύγκρουσης και της συνεχώς αυξανόμενης υπολογιστικής ισχύος, ο SHA-1 δεν θεωρείται πλέον ασφαλής επιλογή για πολλές σύγχρονες εφαρμογές [95].

### 2.2.2 Στάδια υπολογισμών

Ο Αλγόριθμος 2.2.1 παρουσιάζει τα στάδια υπολογισμών του SHA-1. Η διαδικασία ξεκινά με την προεπεξεργασία του αρχικού μηνύματος $M$, η οποία διασφαλίζει ότι το μήκος του μηνύματος θα είναι κατάλληλο για διαίρεση σε μπλοκ των 512 bits. Συγκεκριμένα, αρχικά προστίθεται στο τέλος του μηνύματος ένα bit με τιμή "1". Ακολουθούν όσα bits με τιμή "0" απαιτούνται ώστε το μήκος του μηνύματος να γίνει ακριβώς 64 bits λιγότερο από κάποιο ακέραιο πολλαπλάσιο των 512. Έπειτα, στο τέλος του μηνύματος προσαρτάται μια ακολουθία 64 bits που αναπαριστά το μήκος του αρχικού μηνύματος σε δυαδική μορφή (big-endian). Το big-endian είναι μια σειρά κατάταξης bytes, όπου το πιο σημαντικό byte (το "μεγάλο" άκρο) αποθηκεύεται σε μια χαμηλή διεύθυνση μνήμης, ενώ τα λιγότερο σημαντικά bytes αποθηκεύονται σε υψηλότερες διευθύνσεις μνήμης. Με τον τρόπο αυτό, το τελικό, προεπεξεργασμένο μήνυμα έχει συνολικό μήκος που είναι ακριβές πολλαπλάσιο των 512 bits.

**Algorithm 2.2.1** Ο αλγόριθμος SHA-1

**Require:** Μήνυμα $M$
**Ensure:** 160-bit συμπύκνωμα μηνύματος
1: **function** SHA-1($M$)
2: Αρχικοποίηση των μεταβλητών $H_0, H_1, H_2, H_3, H_4$ στις προκαθορισμένες τιμές
3: Προσθήκη μεταβλητών στο τέλος του $M$
4: Αρχικοποίηση του μετρητή των μπλοκ $N$
5: **for** κάθε μπλοκ $N$ του μηνύματος $M$ **do**
6: Παραγωγή των 80 λέξεων $W[0], \ldots, W[79]$
7: Αρχικοποίηση των τιμών $A, B, C, D, E$ με $H_0, H_1, H_2, H_3, H_4$
8: **for** $i = 0$ μέχρι 79 **do**
9: Υπολογισμός της τιμής $TEMP$
10: Ενημέρωση των μεταβλητών $e = d, d = c, c =$ λογική αριστερή περιστροφή $b$, $b = a, a = TEMP$
11: **end for**
12: Ενημέρωση των $H_0, H_1, H_2, H_3, H_4$ με τις νέες τιμές
13: **end for**
14: **return** Το συμπύκνωμα είναι η συνένωση των $H_0, H_1, H_2, H_3, H_4$
15: **end function**

Στη συνέχεια, το μήνυμα χωρίζεται σε μπλοκ των 512 bits, καθένα από τα οποία επεξεργάζεται ανεξάρτητα. Για κάθε μπλοκ $N$, παράγονται 80 λέξεις $W_0$ έως $W_{79}$ μέσω της διαδικασίας επέκτασης λέξεων. Κατόπιν, οι μεταβλητές κατάστασης $A$, $B$, $C$, $D$ και $E$ αρχικοποιούνται στις τιμές $H_0$, $H_1$, $H_2$, $H_3$ και $H_4$ αντίστοιχα. Για καθένα από τα 80 βήματα του κύριου βρόχου επεξεργασίας, υπολογίζεται μια ενδιάμεση

τιμή $TEMP$, η οποία καθορίζεται από συνάρτηση που εξαρτάται από το εκάστοτε στάδιο και τις τρέχουσες τιμές των μεταβλητών. Αυτή η τιμή χρησιμοποιείται για την ενημέρωση των μεταβλητών κατάστασης μέσω λογικών περιστροφών και αλγεβρικών πράξεων.

Μετά την επεξεργασία κάθε μπλοκ, οι τιμές $H_0$, $H_1$, $H_2$, $H_3$ και $H_4$ ενημερώνονται προσθέτοντας τις τρέχουσες τιμές των αντίστοιχων μεταβλητών, διασφαλίζοντας ότι η εσωτερική κατάσταση του αλγορίθμου εξελίσσεται δυναμικά. Το τελικό κατακερματισμένο αποτέλεσμα προκύπτει από τη συνένωση των τελικών τιμών των $H_0$, $H_1$, $H_2$, $H_3$ και $H_4$, παρέχοντας έναν κατακερματισμό 160 bits που χαρακτηρίζει μοναδικά το αρχικό μήνυμα.

## 2.3 SHA-2

Ο SHA-2 αποτελεί μια οικογένεια αλγορίθμων κατακερματισμού, η οποία περιλαμβάνει διάφορα πρότυπα μεγέθους εξόδου. Συγκεκριμένα, ο SHA-2 προσφέρει τέσσερις βασικές εκδόσεις με διαφορετικά μεγέθη εξόδου: 224, 256, 384 και 512 bit. Επιπλέον, στην οικογένεια περιλαμβάνονται και δύο περικομμένες εκδόσεις, ο SHA-512/224 και ο SHA-512/256, οι οποίες παράγουν αντίστοιχα μικρότερα μεγέθη εξόδου σε σύγκριση με τη βασική έκδοση SHA-512 [96].

Οι εκδόσεις SHA-224 και SHA-256 χρησιμοποιούν μπλοκ των 512 bit, τα οποία αποτελούνται από 16 λέξεις των 32 bit η καθεμία. Συνεπώς, το μήνυμα που πρόκειται να κατακερματιστεί διαιρείται σε τμήματα των 512 bit, με κάθε τμήμα να αντιστοιχεί σε 16 λέξεις. Αντίθετα, οι εκδόσεις SHA-384 και SHA-512 (καθώς και οι περικομμένες τους εκδόσεις) επεξεργάζονται μπλοκ των 1024 bit, τα οποία αποτελούνται από 16 λέξεις των 64 bit. Αυτό συνεπάγεται ότι τόσο το μέγεθος των μπλοκ όσο και των λέξεων είναι διπλάσιο σε σχέση με το SHA-224 και το SHA-256. Η χρήση διαφορετικού μεγέθους λέξεων και μπλοκ στις διάφορες εκδόσεις της οικογένειας SHA-2 προσδίδει στον αλγόριθμο ευελιξία και δυνατότητα προσαρμογής σε ποικίλες απαιτήσεις εφαρμογών, προσφέροντας ταυτόχρονα ένα ευρύ φάσμα επιπέδων ασφάλειας [97, 98].

### 2.3.1 Βασικά στοιχεία

Ο SHA-2 χρησιμοποιεί οκτώ μεταβλητές κατάστασης, οι οποίες συμβολίζονται με τα γράμματα a, b, c, d, e, f, g και h. Οι μεταβλητές αυτές αποτελούν τον πυρήνα της διαδικασίας κατακερματισμού, καθώς κάθε μία αποθηκεύει ενδιάμεσες τιμές κατά τη διάρκεια των διαδοχικών φάσεων του αλγορίθμου. Το μέγεθος κάθε μεταβλητής κατάστασης αντιστοιχεί στο μέγεθος της λέξης της εκάστοτε έκδοσης του SHA-2, προσαρμοζόμενο αναλόγως (32 ή 64 bits). Η αρχικοποίηση των μεταβλητών πραγματοποιείται με προκαθορισμένες σταθερές τιμές, οι οποίες έχουν υπολογιστεί βάσει των δεκαδικών ψηφίων των τετραγωνικών ή κυβικών ριζών των πρώτων αριθμών. Η πρακτική αυτή διασφαλίζει ότι η αρχική κατάσταση του αλγορίθμου είναι μοναδική και μη προβλέψιμη, προσθέτοντας έτσι ένα επιπλέον επίπεδο ασφάλειας στην κρυπτογραφική διαδικασία [99].

Οι αλγόριθμοι SHA-1 και SHA-2 μοιράζονται κοινά χαρακτηριστικά όσον αφορά την προετοιμασία των μηνυμάτων πριν από τον κατακερματισμό. Ένα από αυτά είναι η διαδικασία συμπλήρωσης (padding), η οποία διασφαλίζει ότι το μήκος του μηνύματος γίνεται πολλαπλάσιο του μεγέθους μπλοκ του αλγορίθμου. Συγκεκριμένα, η διαδικασία αρχίζει με την προσθήκη ενός bit "1" στο τέλος του πραγματικού μηνύματος, ακολουθούμενου από μια σειρά από bits "0" μέχρι το μήκος να φτάσει το επιθυμητό μέγεθος, και ολοκληρώνεται με την προσθήκη της δυαδικής αναπαράστασης του μήκους του αρχικού μηνύματος.

Για τον SHA-2, αυτή η διαδικασία διασφαλίζει ότι το τελικό μήνυμα έχει μήκος που είναι πολλαπλάσιο του μεγέθους μπλοκ εισόδου που απαιτεί ο αλγόριθμος: 512 bits για τις εκδόσεις SHA-224 και SHA-256, ή 1024 bits για τις εκδόσεις SHA-384, SHA-512 και τις περικομμένες παραλλαγές τους. Μετά το padding, το μήνυμα διασπάται σε μπλοκ του αντίστοιχου μεγέθους, προετοιμάζοντας το για την επόμενη φάση της επεξεργασίας.

Η επόμενη φάση, η διαδικασία επέκτασης του μηνύματος, μετατρέπει τα αρχικά μπλοκ σε μια σειρά από λέξεις που θα χρησιμοποιηθούν κατά τη διάρκεια της κατακερματιστικής διαδικασίας. Για τον SHA-2, το αρχικό μήνυμα επεκτείνεται σε 64 λέξεις για τις εκδόσεις SHA-224 και SHA-256, ή σε 80 λέξεις για τις εκδόσεις SHA-384 και SHA-512, με τη χρήση λογικών και αριθμητικών πράξεων σε ειδικά διαμορφωμένες εξισώσεις. Αυτή η διαδικασία ενισχύει την πολυπλοκότητα και την ασφάλεια του παραγόμενου κατακερματισμού.

Οι Εξισώσεις επέκτασης (2.8)-(2.10) περιγράφουν το συγκεκριμένο βήμα, όπου κάθε νέα λέξη που παράγεται βασίζεται τόσο στις τιμές προηγούμενων λέξεων όσο και στις αρχικές λέξεις του μηνύματος. Η διαδικασία αυτή ενισχύει την ανθεκτικότητα του αλγορίθμου έναντι κρυπτογραφικών επιθέσεων, καθώς αυξάνει την εντροπία και συμβάλλει ουσιαστικά στη διασφάλιση της ασφάλειας της τελικής τιμής κατακερματισμού [100].

$$\sigma_0 = ROTR^{r1}(W_{t-15})\ XOR\ ROTR^{r2}(W_{t-15})\ XOR\ SHR^{r3}(W_{t-15}) \tag{2.8}$$

$$\sigma_1 = ROTR^{q1}(W_{t-2})\ XOR\ ROTR^{q2}(W_{t-2})\ XOR\ SHR^{q3}(W_{t-2}) \tag{2.9}$$

$$W_t = W_{t-16} + \sigma_0 + W_{t-7} + \sigma_1, \quad 16 \leq t \leq n \tag{2.10}$$

οπου $t$ = 16 εως $n$, $ROTR^n(X)$ περιστρέφει τη λέξη $X$ προς τα δεξιά κατά $n$ $bit$, και $SHR^n(X)$ μετατοπίζει δεξιά τη λέξη $X$ κατά $n$ $bit$. Οι σταθερές $n, r_1, r_2, r_3, q_1, q_2$, και $q_3$ χρησιμοποιούνται στις λειτουργίες $ROTR$ και $SHR$ οι οποιες ποικίλλουν ανάλογα με την έκδοση του $SHA$-2 που εφαρμόζεται.

Για $n = 63$ στους SHA-224 και SHA-256 οι τιμές αυτές έχουν οριστεί ως:

$$r_1 = 7, r_2 = 18, r_3 = 3, q_1 = 7, q_2 = 19, q_3 = 10$$

ενώ για $n = 79$ στους SHA-384 και SHA-512:

$$r_1 = 1, r_2 = 8, r_3 = 7, q_1 = 19, q_2 = 61, q_3 = 6$$

Οι τιμές αυτές είναι προκαθορισμένες και συνδέονται άμεσα με την αρχιτεκτονική και τις απαιτήσεις ασφαλείας της κάθε εκδόσης του SHA-2, επιτρέποντας τον εντοπισμό και την αντιμετώπιση διαφορετικών τύπων κρυπτογραφικών απειλών και επιθέσεων.

Δύο θεμελιώδεις λογικές συναρτήσεις [101] παίζουν κεντρικό ρόλο στη διαδικασία δημιουργίας του κατακερματισμού: η συνάρτηση $Choose$ και η συνάρτηση $Majority$. Η συνάρτηση $Choose$ επιλέγει μεταξύ δύο εισόδων με βάση μια τρίτη: για κάθε θέση $bit$, αν το αντίστοιχο $bit$ της τρίτης εισόδου είναι 1, τότε επιλέγεται

το $bit$ από την πρώτη είσοδο· αν είναι 0, επιλέγεται το $bit$ από τη δεύτερη είσοδο. Αντιθέτως, η συνάρτηση $Majority$ λαμβάνει υπόψη τις τιμές σε κάθε θέση $bit$ για τις τρεις εισόδους και επιστρέφει εκείνη που εμφανίζεται ως πλειοψηφία (δηλαδή, εάν δύο ή περισσότερες είσοδοι έχουν τιμή "1" ή "0", αυτή η τιμή επιστρέφεται ως αποτέλεσμα).

$$Choose(x, y, z) = (x\ AND\ y)\ XOR\ (\ NOT\ x\ AND\ z) \tag{2.11}$$

$$Majority(x, y, z) = (x\ AND\ y)\ XOR\ (x\ AND\ z)\ XOR\ (y\ AND\ z) \tag{2.12}$$

Για την ενίσχυση της ασφάλειας της συνάρτησης κατακερματισμού SHA-2, εισάγονται δύο επιπλέον λογικές συναρτήσεις, οι οποίες είναι απαραίτητες για την περαιτέρω ενίσχυση της πολυπλοκότητας και της δομικής διάχυσης στον αλγόριθμο. Οι συναρτήσεις αυτές προσθέτουν επιπλέον επίπεδα πολυπλοκότητας στην εσωτερική δομή του αλγορίθμου, βελτιώνοντας την ανθεκτικότητά του έναντι επιθέσεων σύγκρουσης κατακερματισμού, όπου δύο διαφορετικές είσοδοι ενδέχεται να παράγουν τον ίδιο κατακερματισμό.

Η συνάρτηση (2.13), υλοποιεί μια σειρά από περιστροφές και λογικές πράξεις XOR πάνω στην είσοδο $x$. Συγκεκριμένα, περιστρέφει το $x$ προς τα δεξιά κατά τρεις διαφορετικές τιμές θέσεων και στη συνέχεια εφαρμόζει την πράξη XOR στα αποτελέσματα αυτών των περιστροφών για να παράγει την τελική τιμή.

$$\sum_{0}(x) = ROTR^{r_1}(x)\ XOR\ ROTR^{r_2}(x)\ XOR\ ROTR^{r_3}(x) \tag{2.13}$$

Αντίστοιχα, η συνάρτηση (2.14), ακολουθεί μια παρόμοια διαδικασία, αλλά με διαφορετικές τιμές περιστροφής. Αυτή η διαφορά στις τιμές περιστροφής διασφαλίζει ότι οι δύο συναρτήσεις παράγουν διαφορετικά μοτίβα από την ίδια είσοδο, προσθέτοντας μια στρώση ασφάλειας στη διαδικασία [102].

$$\sum_{1}(x) = ROTR^{q_1}(x)\ XOR\ ROTR^{q_2}(x)\ XOR\ ROTR^{q_3}(x) \tag{2.14}$$

Οι σταθερές $r_1, r_2, r_3, q_1, q_2$, και $q_3$ ποικίλλουν ανάλογα με την έκδοση του SHA-2 που εφαρμόζεται. Η ακριβής τιμή κάθε σταθεράς επηρεάζει το πώς τα μπλοκ δεδομένων ανακατεύονται και συνδυάζονται, προσδίδοντας στον κατακερματισμό μοναδικότητα και ανθεκτικότητα σε επιθέσεις.

Για SHA-224 και SHA-256 οι τιμές των σταθερών είναι χαμηλότερες:

$$r_1 = 2, r_2 = 13, r_3 = 22, q_1 = 6, q_2 = 11, q_3 = 25$$

Για SHA-384 και SHA-512 οι τιμές των σταθερών είναι σημαντικά υψηλότερες:

$$r_1 = 28, r_2 = 34, r_3 = 39, q_1 = 14, q_2 = 18, q_3 = 41$$

### 2.3.2 Στάδια υπολογισμών

Ο Αλγόριθμος 2.3.1 παρουσιάζει τα στάδια υπολογισμών του SHA-2. Η διαδικασία ξεκινά με την αρχικοποίηση μιας σειράς μεταβλητών κατάστασης, στις θέσεις $a$, $b$, $c$, $d$, $e$, $f$, $g$ και $h$, με προκαθορισμένες σταθερές τιμές που έχουν οριστεί από το πρότυπο του αλγορίθμου.

**Algorithm 2.3.1** Ο αλγόριθμος SHA-2

**Require:** Προεπεξεργασμένο Μήνυμα $M = (M_0, M_1, \ldots, M_n)$ σε Μπλοκ
**Ensure:** Εξαγόμενο Hash (224 ή 256 ή 384 ή 512)

Αρχικοποίηση
$a = H_0^{(i-1)}, b = H_1^{(i-1)},$
$c = H_2^{(i-1)}, d = H_3^{(i-1)},$
$e = H_4^{(i-1)}, f = H_5^{(i-1)},$
$g = H_6^{(i-1)}, h = H_7^{(i-1)}$
**for** $t = 0$ to $n$ do **do**
    **if** $t < 16$ **then**
        $W_t \leftarrow M_t$
    **else**
        $W_t \leftarrow \sigma_1(W_{t-2}) + W_{t-7} + \sigma_0(W_{t-15}) + W_{t-16}$
    **end if**
**end for**
**for** $t = 0$ to $j$ do **do**
    $Temp_1 \leftarrow h + \Sigma_1(e) + Choose(e, f, g) + K_t + W_t$
    $Temp_2 \leftarrow \Sigma_0(a) + Majority(a, b, c)$
    $h \leftarrow g$
    $g \leftarrow f$
    $f \leftarrow e$
    $e \leftarrow d + Temp_1$
    $d \leftarrow c$
    $c \leftarrow b$
    $b \leftarrow a$
    $a \leftarrow Temp_1 + Temp_2$
**end for**
**for** $t = 0$ to $n$ do **do**
    $H_0^{(t)} \leftarrow a + H_0^{(t-1)}$
    $H_1^{(t)} \leftarrow b + H_1^{(t-1)}$
    $H_2^{(t)} \leftarrow c + H_2^{(t-1)}$
    $H_3^{(t)} \leftarrow d + H_3^{(t-1)}$
    $H_4^{(t)} \leftarrow e + H_4^{(t-1)}$
    $H_5^{(t)} \leftarrow f + H_5^{(t-1)}$
    $H_6^{(t)} \leftarrow g + H_6^{(t-1)}$
    $H_7^{(t)} \leftarrow h + H_7^{(t-1)}$
**end for**
**return** Hash
$SHA-224 \leftarrow$ συνένωση των $H_0^{(n)}, H_1^{(n)}, H_2^{(n)}, H_3^{(n)}, H_4^{(n)}, H_5^{(n)}, H_6^{(n)}$
$SHA-256, 512 \leftarrow$ συνένωση των $H_0^{(n)}, H_1^{(n)}, H_2^{(n)}, H_3^{(n)}, H_4^{(n)}, H_5^{(n)}, H_6^{(n)}, H_7^{(n)}$
$SHA-384 \leftarrow$ συνένωση των $H_0^{(n)}, H_1^{(n)}, H_2^{(n)}, H_3^{(n)}, H_4^{(n)}, H_5^{(n)}$

Κάθε ένα από αυτά τα αρχικά στοιχεία προέρχεται από το προηγούμενο block της αλυσίδας hash (παράμετροι $H_j^{(i-1)}$). Η διαδικασία συνεχίζεται με την επεξεργασία του μηνύματος $M$ σε μπλοκ των 512 bits. Στη συνέχεια, κάθε block υφίσταται μια σειρά περίπλοκων λειτουργιών για την παραγωγή της τελικής τιμής κατακερματισμού (hash).

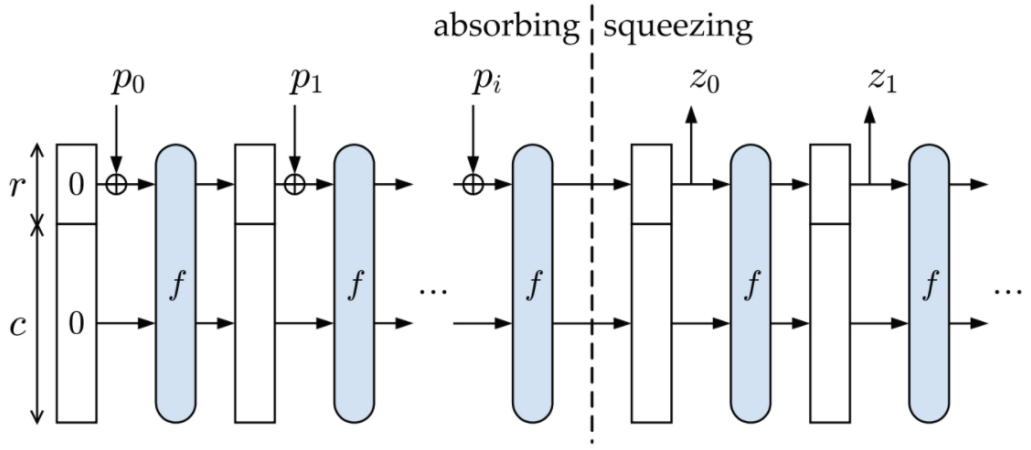
Για κάθε μπλοκ, το μήνυμα χωρίζεται σε λέξεις $W_t$, με $t$ να κυμαίνεται από 0 έως $n$. Οι λέξεις αυτές μετασχηματίζονται μέσω των συναρτήσεων συμπίεσης $\sigma_0$ και $\sigma_1$, καθώς και μέσω διαδοχικών βημάτων (rounds) όπου εφαρμόζονται λειτουργίες όπως οι $\Sigma_0$, $\Sigma_1$, $Choose$ και $Majority$, σε συνδυασμό με σταθερές $K_t$. Οι συγκεκριμένες λειτουργίες ενισχύουν την πολυπλοκότητα της διαδικασίας κατακερματισμού και συμβάλλουν καθοριστικά στην ασφάλεια του αλγορίθμου απέναντι σε κρυπτογραφικές επιθέσεις.

Τέλος, ο αλγόριθμος συνενώνει τις τιμές που προκύπτουν από τις παραπάνω λειτουργίες για να παραγάγει το τελικό hash. Ειδικότερα, για το SHA-224, το hash προκύπτει από τη συνένωση των τιμών $H_0^{(n)}$ έως $H_6^{(n)}$· για το SHA-256 και το SHA-512, από τις τιμές $H_0^{(n)}$ έως $H_7^{(n)}$· ενώ για το SHA-384, από τις τιμές $H_0^{(n)}$ έως $H_5^{(n)}$ [28].

## 2.4 SHA-3

Το 2012, το Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας προχώρησε στη δημοσίευση του προτύπου SHA-3, έπειτα από έναν εκτεταμένο διαγωνισμό σχεδιασμού με στόχο την ανάδειξη του διαδόχου των προηγούμενων αλγορίθμων κατακερματισμού της οικογένειας SHA. Στον τελικό γύρο του διαγωνισμού διακρίθηκαν πέντε τεχνολογίες: Keccak, Grøstl, BLAKE, JH και Skein, με τον Keccak να επικρατεί τελικά και να ανακηρύσσεται ως το νέο πρότυπο για τον SHA-3 [103].

Σε αντίθεση με τα προηγούμενα πρότυπα, SHA-1 και SHA-2, ο SHA-3 βασίζεται κυρίως στη δομή απορρόφησης και συμπίεσης (sponge construction), όπως φαίνεται στο Σχήμα 2.2 μια τεχνική που προσφέρει αυξημένη αντοχή σε επιθέσεις κρυπτανάλυσης και εισάγει νέα προσέγγιση στον σχεδιασμό ασφαλών κρυπτογραφικών λύσεων. Αυτή η εξέλιξη αποτελεί σημαντικό ορόσημο στην ιστορία της κρυπτογραφίας, οδηγώντας την επιστημονική κοινότητα προς πιο ευέλικτες και ασφαλείς λύσεις κατακερματισμού [104, 105].




Σχήμα 2.2: Κατασκευή σφουγγαριού του SHA-3

### 2.4.1 Λειτουργίες σφουγγαριού (Sponge Functions)

Η κρυπτογραφία με λειτουργία σπόγγου (sponge function ή sponge construction) είναι μια καινοτόμος τεχνική που προσφέρει ευελιξία και ασφάλεια στις σύγχρονες κρυπτογραφικές εφαρμογές. Η καρδιά αυτής της δομής είναι η ιδέα του σπόγγου, ο οποίος χειρίζεται δεδομένα μεταβαλλόμενου μήκους με τη μορφή ενός απλού, αλλά ισχυρού μοτίβου λειτουργίας. Η δομή αυτή λειτουργεί με βάση μια κατάσταση $b = r+c$ bits, όπου $r$ αντιπροσωπεύει τον ρυθμό μετάδοσης bit και $c$ τη χωρητικότητα, η οποία είναι αφιερωμένη στην εξασφάλιση της ασφάλειας της διαδικασίας.

Αρχικά, η κατάσταση αρχικοποιείται με μηδενικά και η συμβολοσειρά εισόδου τροποποιείται ώστε το τελικό της μήκος να είναι διαιρετό από τον ρυθμό $r$. Αυτό επιτυγχάνεται μέσω ενός μηχανισμού συμπλήρωσης που εξασφαλίζει ότι κάθε τμήμα της εισόδου μπορεί να διαχειριστεί αποτελεσματικά. Στην πράξη, η εισαγόμενη συμβολοσειρά χωρίζεται σε μπλοκ ίσου μεγέθους $p_0, p_1, ..., p_i$, καθένα από τα οποία είναι $r$ bits.

Η φάση απορρόφησης αποτελεί το επόμενο βήμα, όπου κάθε ένα από αυτά τα μπλοκ υποβάλλεται σε μια διαδικασία XOR με τα πρώτα $r$ bits της κατάστασης $b$. Αυτή η λειτουργία ενσωματώνει το μπλοκ δεδομένων στην κατάσταση με έναν τρόπο που διατηρεί την ασφάλεια και την εντροπία. Μετά το XOR, η συνάρτηση μετάθεσης $f$ στην ενημερωμένη κατάσταση, προσφέρένα περιεκτικό ανακάτεμα των bits που ενισχύει περαιτέρω την ασφάλεια της διαδικασίας. Αυτή η εναλλασσόμενη σειρά απορρόφησης και μετάθεσης συνεχίζεται μέχρι όλα τα μπλοκ εισόδου να έχουν

ενσωματωθεί πλήρως στην κατάσταση, καταλήγοντας σε ένα ισχυρά ανακατεμένο σύνολο δεδομένων που αποτελεί τη βάση για το τελικό συμπύκνωμα [106, 107].

Με την ολοκλήρωση της φάσης απορρόφησης, όπου όλα τα μπλοκ των δεδομένων ενσωματώνονται επιτυχώς στην κατάσταση του σπόγγου, η διαδικασία μεταβαίνει στο επόμενο κρίσιμο στάδιο: τη φάση συμπίεσης. Η στιγμή αυτή σηματοδοτεί την έναρξη της εξαγωγής του τελικού κατακερματισμού από την κατάσταση, που έχει πλέον εμπλουτιστεί με τα δεδομένα εισόδου. Αυτό επιτυγχάνεται με την επιλογή των λιγότερο σημαντικών $r$ bits της κατάστασης $b$, από τα οποία προκύπτουν τα μπλοκ εξόδου $z_0, z_1, ...$, που είναι προορισμένα να συνθέσουν το αποτέλεσμα του κατακερματισμού.

Εάν το μέγεθος της επιθυμητής εξόδου είναι ίσο ή μικρότερο από το $z_0$, τότε η απαιτούμενη έξοδος εξάγεται απευθείας από τα λιγότερο σημαντικά bits αυτού του μπλοκ. Ωστόσο, σε περιπτώσεις όπου το απαιτούμενο μήκος εξόδου υπερβαίνει τα δεδομένα που παρέχονται από το $z_0$, η διαδικασία προχωρά με την εφαρμογή της συνάρτησης μετάθεσης $f$ στην επόμενη έξοδο $z_1$, και ούτω καθεξής. Αυτός ο κυκλικός μηχανισμός παραγωγής επιτρέπει τη δημιουργία μιας συνεχούς ροής εξόδου, η οποία θα συνενωθεί τελικά για να δώσει τον επιθυμητό κατακερματισμό.

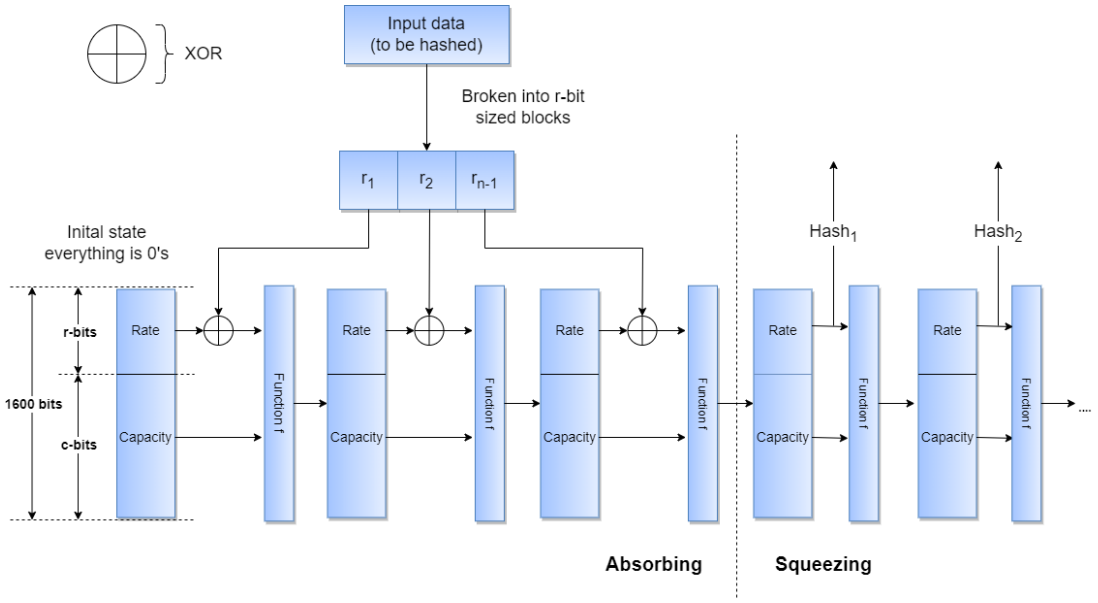
Αυτή η ευέλικτη προσέγγιση εξασφαλίζει ότι ο αλγόριθμος μπορεί να παρέχει κατακερματισμούς με διαφορετικά μήκη εξόδου, όπως 224, 256, 384, ή 512 bits, ανταποκρινόμενος στις ποικίλες ανάγκες των εφαρμογών που τον χρησιμοποιούν. Επιπλέον, η δομή του σπόγγου προσφέρει μια εναλλακτική σε πιο παραδοσιακές μεθόδους, επιτρέποντας την εφαρμογή πρόσθετης μετάθεσης για αυθαίρετα μήκη εξόδου, όπως συμβαίνει με άλλους αλγόριθμους λοπως οι SHAKE128 και SHAKE256, διευρύνοντας το φάσμα των δυνατοτήτων του [33].

Σύμφωνα με τον Πίνακα 2.3, το Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας έχει καθορίσει τέσσερις διαφορετικές παραλλαγές του αλγορίθμου κατακερματισμού SHA-3 με βάση το μήνυμα $M$ και το μέγεθος μήκους εξόδου $d$.

Πίνακας 2.3: Οι τέσσερις μορφές του SHA-3.

| **Μήνυμα ($M$)** | **Μέγεθος μήκους εξόδου ($d$)** | **Τιμή ($r$) $(blocksize)$** | **Χωρητικότητα ($c$)** |
|---|---|---|---|
| 224 | 224 | 1152 | 448 |
| 256 | 256 | 1088 | 512 |
| 384 | 384 | 832 | 768 |
| 512 | 512 | 576 | 1024 |

Στο Σχήμα 2.3 δίνεται ένα παράδειγμα με τις δύο κύριες φάσεις απορρόφησης και συμπίεσης με εισαγώμενο μήνυμα 1600 bit. Το μήνυμα εισόδου απορροφάται στην κατάσταση χρησιμοποιώντας την κατασκευή του σφουγγαριού στη φάση απορρόφησης. Αυτή η φάση προετοιμάζει τα δεδομένα εισόδου εφαρμόζοντας τη συνάρτηση $f$, η οποία ενσωματώνει λειτουργίες bitwise, αρθρωτή προσθήκη και λειτουργίες περιστροφής για την εισαγωγή διάχυσης και σύγχυσης. Στη φάση συμπίεσης, η επιθυμητή έξοδος κατακερματισμού λαμβάνεται με επανειλημμένη συμπίεση μπλοκ δεδομένων από την κατάσταση.




Σχήμα 2.3: Λειτουργία σφουγγαριού του αλγόριθμου SHA-3.

Συγκεκριμένα, ο SHA-3 διακρίνεται για την υποστήριξη δύο βασικών τρόπων λειτουργίας: την παραγωγή κατακερματισμού με σταθερό μήκος εξόδου και την παραγωγή κατακερματισμού με μεταβλητό μήκος εξόδου. Η πρώτη λειτουργία επιτρέπει την παραγωγή κατακερματισμών σε στάνταρ μεγέθη, όπως 224, 256, 384, και 512 bits, προσφέροντας σταθερότητα και προβλεψιμότητα στις εφαρμογές που απαιτούν συγκεκριμένα μήκη hash [108]. Αυτό καθιστά τον SHA-3 ιδανικό για μια πληθώρα κρυπτογραφικών εφαρμογών, από την ψηφιακή υπογραφή μέχρι την ασφάλεια δικτύων [109].

Ο SHA-3 διαθέτει μια μοναδική δομή μετάθεσης, όπου το επίπεδο λειτουργίας της συνάρτησης μετάθεσης καθορίζεται από το $l$, το οποίο μπορεί να λάβει τιμές όπως 25, 50, 100, 200, 400, 800, και 1600. Η πιο συνήθης ρύθμιση χρησιμοποιεί το $b = 1600$ και $l = (0, 1, 2, 3, 4, 5, 6)$, προσφέροντας μια ισχυρή βάση για την ασφαλή επεξεργασία και μετατροπή των δεδομένων [110]. Η τιμή του $b$ υπολογίζεται

μέσω της Εξίσωσης (2.15), παρέχοντας μια δυναμική και ευέλικτη βάση για την κρυπτογράφηση και τον κατακερματισμό δεδομένων.

$$b = 25 \times 2^l \tag{2.15}$$

Η διαδικασία αυτή επιτρέπει στον SHA-3 να προσφέρει προηγμένη ασφάλεια και να αντιμετωπίσει τις σύγχρονες προκλήσεις στο πεδίο της κρυπτογραφίας [111].

Οι δυο βασικές ομάδες παραμέτρων στη λειτουργία σταθερής εξόδου του SHA-3 προορίζονται για την επεξεργασία και παραγωγή κατακερματισμών διαφορετικών μεγεθών. Η πρώτη ομάδα, με ρυθμό μετάδοσης $r = 1344$ και χωρητικότητα $c = 256$, είναι ειδικά σχεδιασμένη για την παραγωγή κατακερματισμών με μήκη 224 και 256 bits. Αυτό επιτρέπει την αποτελεσματική επεξεργασία και ασφαλή κωδικοποίηση δεδομένων, ιδανική για εφαρμογές που απαιτούν μέτριο επίπεδο ασφαλείας αλλά μεγάλη απόδοση.

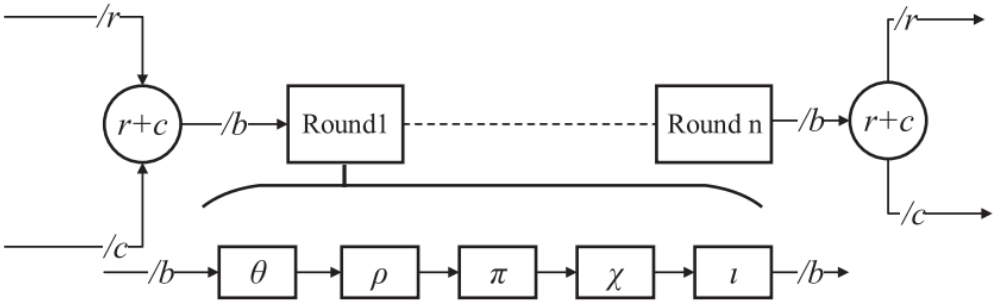
Από την άλλη πλευρά, η δεύτερη ομάδα, με ρυθμό $r = 1088$ και χωρητικότητα $c = 512$, είναι προορισμένη για την παραγωγή κατακερματισμών με μήκη 384 και 512 bits, προσφέροντας μια επιπλέον στρώση ασφάλειας για την προστασία ευαίσθητων δεδομένων [112]. Η διάκριση αυτών των δυο ομάδων υπογραμμίζει την ευελιξία του SHA-3 στην προσαρμογή των κρυπτογραφικών λειτουργιών στις ειδικές ανάγκες κάθε εφαρμογής, επιτρέποντας την ακριβή διαμόρφωση της απόδοσης και της ασφαλείας. Αυτή η προσαρμοστικότητα στις λειτουργικές παραμέτρους ενισχύει την προστασία κατά των κρυπτογραφικών επιθέσεων, καθιστώντας τον SHA-3 μια ισχυρή και αξιόπιστη επιλογή για την ασφάλεια της ψηφιακής εποχής [113].

Στη λειτουργία του σταθερού μεγέθους εξόδου, η εξαγωγή του τελικού κατακερματισμού γίνεται από την αρχή της φάσης συμπίεσης, όπου τα λιγότερο σημαντικά bits της πρώτης εξόδου $z_0$ επιλέγονται βάσει του επιθυμητού μήκους κατακερματισμού (224, 256, 384, και 512 bits). Αυτή η μέθοδος επιτρέπει την παραγωγή μιας προβλεπτικής και ασφαλούς εξόδου, προσαρμοσμένης στις ανάγκες της εκάστοτε εφαρμογής [114].

Όταν η απαίτηση αφορά σε κατακερματισμό μεταβλητού μήκους, ο SHA-3 προσφέρει μια ενδιαφέρουσα προσέγγιση: όλα τα bits της εξόδου $z$ μπορούν να χρησιμοποιηθούν ανάλογα με το επιθυμητό μήκος, παρέχοντας μια προσαρμοστική και ευέλικτη λύση για διάφορες εφαρμογές. Η ευελιξία αυτή επεκτείνεται και στην επιλογή του σημείου εξόδου, καθώς οποιαδήποτε εξαγωγή $z_i$ μπορεί να

επιλεγεί για την παραγωγή του τελικού κατακερματισμού. Αυτός ο προσαρμοστικός μηχανισμός επιτρέπει την εύκολη διαχείριση απαιτήσεων που ποικίλλουν σε περιπλοκότητα και ασφάλεια, εξασφαλίζοντας ταυτόχρονα την ακεραιότητα και την αποτελεσματικότητα των δεδομένων [115]. Κατά συνέπεια, ο SHA-3 διαθέτει ένα ιδιαίτερα ισχυρό και πολυλειτουργικό σύστημα κατακερματισμού που εξυπηρετεί τις ανάγκες τόσο για σταθερή όσο και για μεταβλητή έξοδο, παρέχοντας μια σταθερή βάση για την ασφάλεια στον κυβερνοχώρο. Η ευελιξία του στον καθορισμό της έξοδου και η δυνατότητα προσαρμογής στις εκάστοτε απαιτήσεις κάνουν τον SHA-3 μια προτιμητέα επιλογή για την προστασία κρίσιμων δεδομένων σε μια πληθώρα εφαρμογών [116].

Η συνάρτηση SHA-3, όπως φαίνεται στο Σχήμα 2.4, χαρακτηρίζεται από την καινοτόμο δομή της και την προσαρμοστικότητά της στις ανάγκες ασφαλείας των σύγχρονων κρυπτογραφικών εφαρμογών. Κεντρικό στοιχείο στην αρχιτεκτονική του αλγορίθμου είναι η κατάσταση $b$, η οποία αντιστοιχεί στην προσάρτησει του ρυθμού μετάδοσης $r$ και της χωρητικότητας $c$, αρχικοποιημένη σε μηδέν. Η συνολική δομή και η λειτουργία του αλγορίθμου βασίζεται στην επιλογή του επιπέδου της μετάθεσης, με το $b = 1600$ να αποτελεί την πιο διαδεδομένη επιλογή λόγω της ισορροπημένης απόδοσης και ασφαλείας που προσφέρει [117].




Σχήμα 2.4: Λειτουργία της συνάρτησης SHA-3, όπου επαναλαμβάνονται τα πέντε βήματα για κάθε γύρο $\theta\ (theta), \rho\ (rho), \pi\ (pi), \chi\ (chi)$, και $\iota\ (iota)$.

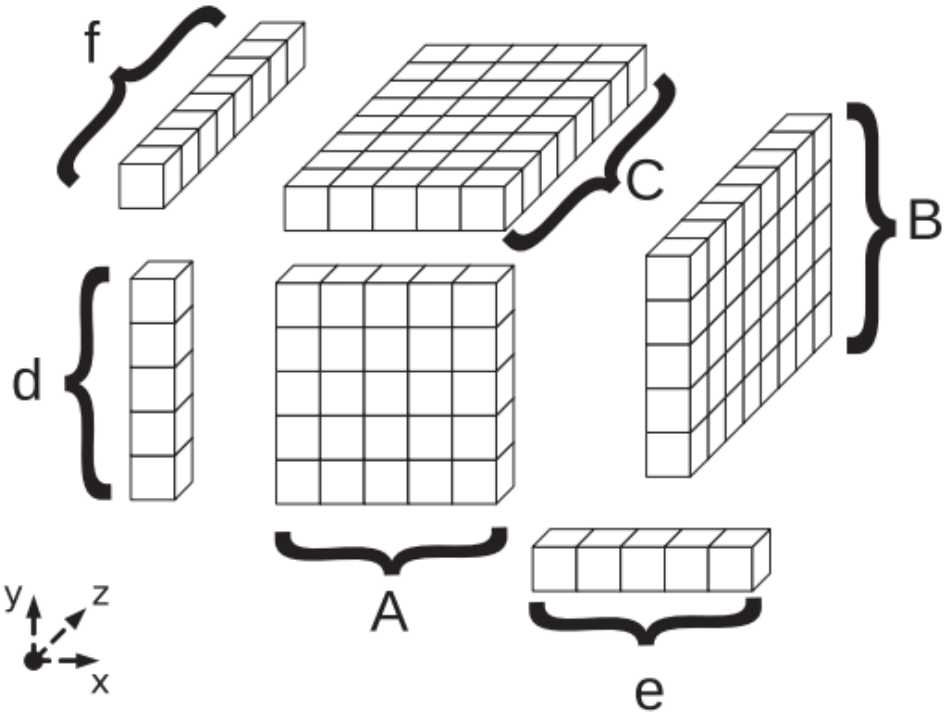
Η επεξεργασία των δεδομένων εντός της συνάρτησης SHA-3 πραγματοποιείται μέσω μιας σειράς γύρων επεξεργασίας, των οποίων ο αριθμός καθορίζεται βάσει της τιμής $l$, σύμφωνα με τη Εξίσωση (2.16).

$$\textit{Rounds} = 12 + 2l \tag{2.16}$$

Αυτό σημαίνει ότι η πολυπλοκότητα και η ασφάλεια της επεξεργασίας μπορούν να προσαρμοστούν ανάλογα με τις ανάγκες της εφαρμογής, με την τιμή $l$ να

επιλέγεται από μια διακριτή σειρά τιμών $(0, 1, 2, 3, 4, 5, 6)$, αυξάνοντας την ευελιξία του κατακερματισμού. Το $b = 1600$, που αποτελεί τη βάση για το πιο συχνά χρησιμοποιούμενο επίπεδο μετάθεσης του SHA-3, επιτρέπει την υψηλή απόδοση και τη συμμόρφωση με τις αυξημένες απαιτήσεις ασφαλείας των σύγχρονων κρυπτογραφικών εφαρμογών. Η ευελιξία στον αριθμό των γύρων επεξεργασίας παρέχει επίσης τη δυνατότητα προσαρμογής του αλγορίθμου σε διάφορα επίπεδα ανάλυσης και επεξεργασίας, εξασφαλίζοντας ταυτόχρονα την ακεραιότητα και την αποδοτικότητα της διαδικασίας κατακερματισμού [118].

Ο αλγόριθμος SHA-3 χρησιμοποιεί μια πρωτοποριακή τρισδιάστατη μητρική δομή για τη διαχείριση των δεδομένων και την εφαρμογή των κρυπτογραφικών λειτουργιών του, όπως φαίνεται στο Σχήμα 2.5.




Σχήμα 2.5: Πίνακας κατάστασης SHA-3 $(A \times B \times C)$, που αντιπροσωπεύεται ως $3D - Matrix$. Κάθε τετράγωνο αντιπροσωπεύει ένα $bit$: (A) φέτα, (B) φύλλο, (C) επίπεδο, (d) στήλη, (e) σειρά, (f) λωρίδα.

Η κατάσταση $S$ στον SHA-3 αναπαριστάται ως ένας πίνακας τριών διαστάσεων $(A \times B \times C)$, προσφέροντας μια ιδιαίτερη ευχέρια στην επεξεργασία και στην ανάλυση των δεδομένων. Αυτή η δομή επιτρέπει την αποτελεσματική και ευέλικτη εφαρμογή των γύρων επεξεργασίας, ενώ ταυτόχρονα διασφαλίζει την ασφάλεια και την ανθεκτικότητα του αλγορίθμου. Κάθε γύρος επεξεργασίας στον SHA-3

διακρίνεται με τη χρήση μιας ξεχωριστής σταθεράς $RC_i$, η οποία ενσωματώνεται στη συνάρτηση μετάθεσης για να ενισχύσει την ασφάλεια και να προσθέσει επιπλέον πολυπλοκότητα στη διαδικασία. Ο αλγόριθμος ακολουθεί μια σταθερή ακολουθία πέντε βημάτων: $\theta\ (theta), \rho\ (rho), \pi\ (pi), \chi\ (chi)$, και $i\ (iota)$, όπου κάθε βήμα εφαρμόζεται στον πίνακα καταστάσεων και συνεισφέρει στον συνολικό μετασχηματισμό των δεδομένων. Η διαδικασία αυτή ενισχύει την ασφάλεια μέσω της περίπλοκης αλληλεπίδρασης μεταξύ των διαφορετικών στοιχείων του πίνακα κατάστασης και την εφαρμογή πολυεπίπεδων μετασχηματισμών [119]. Οι τιμές των $A$ και $B$ καθορίζονται στον αριθμό 5, προσδίδοντας μια σταθερή διάσταση στον πίνακα, ενώ η τιμή του $C$ καθορίζεται από το $w$, βάσει της Εξίσωσης (2.17),

$$w = 2^l \tag{2.17}$$

με το $l$ να προσδιορίζει το επίπεδο της μετάθεσης (απο 0 εως 6) και να επηρεάζει άμεσα τον αριθμό των γύρων και την συνολική ασφάλεια της διαδικασίας.

### 2.4.2 Η συνάρτηση $f$ του SHA-3

Η συνάρτηση $f$ του SHA-3 εκτελείται για 24 γύρους, καθένας από τους οποίους είναι μια διαδοχική εκτέλεση των αντίστοιχων πέντε βημάτων $\theta\ (theta), \rho\ (rho), \pi\ (pi), \chi\ (chi)$, και $i\ (iota)$. Αναλυτικά τα βήματα παρουσιάζονται παρακάτω.

- Βήμα $\theta\ (theta)$: Αυτό το βήμα είναι ζωτικής σημασίας για τη διασφάλιση της διαφορετικότητας και της διασποράς των δεδομένων εισόδου, βοηθώντας έτσι στην ανθεκτικότητα ενάντια σε κρυπτογραφικές επιθέσεις. Αυτή η διαδικασία ενισχύει την αντίσταση του αλγορίθμου σε κρυπτογραφικές επιθέσεις, καθώς μια μικρή αλλαγή στα δεδομένα εισόδου θα έχει μεγάλο και διαφορετικό αντίκτυπο στον πίνακα καταστάσεων, διασπείροντας την επίδραση αυτής της αλλαγής σε ολόκληρο τον πίνακα και βοηθώντας να αποφευχθεί η δημιουργία των ίδιων hash τιμών από διαφορετικά δεδομένα εισόδου.

  Στο βήμα $\theta$, η διαδικασία επεξεργάζεται την κατάσταση των δεδομένων, η οποία αντιπροσωπεύεται ως ένας δισδιάστατος πίνακας $5 \times 5$, με κάθε κελί να περιέχει w bits. Στην ουσία, κάθε γραμμή αυτού του πίνακα αποτελεί μια λωρίδα και η στοιχειώδης μονάδα του πίνακα, το κελί, είναι ένα slice. Η

λειτουργία $\theta$ είναι υπεύθυνη για την ανάμειξη των bits εντός των λωρίδων σε όλο τον πίνακα. Αυτό γίνεται με τον υπολογισμό της ισοτιμίας XOR κάθε λωρίδας και τη συνέχεια μιας περιστροφής και εφαρμογής XOR με άλλα bits του πίνακα, προκειμένου να προκύψει αυξημένη διαφοροποίηση και κρυπτογραφική ανθεκτικότητα.

Η Εξισώση (2.18) που περιγράφει τη λειτουργία του βήματος $\theta$ χρησιμοποιεί δύο πίνακες: $C[x]$ για την ισοτιμία κάθε στήλης της λωρίδας και $D[x]$ για την περιστροφή της ισοτιμίας. Η ισοτιμία κάθε στήλης συνδυάζεται μετά στην συνέχεια με τα bits της κατάστασης χρησιμοποιώντας XOR, δημιουργώντας μια μη γραμμική ανάμειξη των δεδομένων που αυξάνει την ασφάλεια του αλγορίθμου κατακερματισμού απέναντι σε επιθέσεις κρυπτανάλυσης. Η συνεχής μεταβολή του πίνακα καταστάσεων στοχεύει στη διασφάλιση ότι ακόμη και μικρές αλλαγές στην είσοδο θα προκαλέσουν σημαντικές και απρόβλεπτες αλλαγές στην έξοδο του κατακερματισμού, ενισχύοντας την πολυπλοκότητα και την ασφάλεια του κατακερματισμού.

$$
\begin{aligned}
&C[x] = A[x,0] \oplus A[x,1] \oplus A[x,2] \oplus A[x,3] \oplus A[x,4], \\
&\text{για } x = 0,1,2,3,4 \\
&D[x] = C[x-1] \oplus (\text{ROTATE}(C[x+1],1)), \\
&\text{για } x = 0,1,2,3,4 \\
&A[x,y] = A[x,y] \oplus D[x], \\
&\text{για } x = 0,1,2,3,4
\end{aligned}
\tag{2.18}
$$

- Βήμα $\rho$ $(rho)$: Αυτό το βήμα ασχολείται με την περιστροφή (ROTATE) των bits στον πίνακα κατάστασης. Συγκεκριμένα, κάθε bit στον πίνακα κατάστασης περιστρέφεται αριστερά για έναν καθορισμένο αριθμό θέσεων. Η τιμή μετατόπισης περιστροφής συμβολίζεται με $r[x,y]$ η οποια ειναι μια σταθερή τιμή που εκχωρείται σύμφωνα με τον Πίνακα 2.4.

Πίνακας 2.4: Η σταθερή τιμή $r[x,y]$ στο βήμα $r$ $(rho)$.

| | $x=3$ | $x=4$ | $x=0$ | $x=1$ | $x=2$ |
|---|---|---|---|---|---|
| $y=2$ | 25 | 39 | 3 | 10 | 13 |
| $y=1$ | 55 | 20 | 36 | 44 | 6 |
| $y=0$ | 28 | 27 | 0 | 1 | 62 |
| $y=4$ | 56 | 14 | 18 | 2 | 61 |
| $y=3$ | 21 | 8 | 41 | 54 | 15 |

Η λειτουργία αυτή είναι απαραίτητη για να εξασφαλίσει ότι το μοτίβο των bits θα διασπαρεί κατά τον επόμενο γύρο λειτουργιών. Η περιστροφή βοηθά στην δημιουργία μιας πιο σύνθετης δομής μέσα στον πίνακα κατάστασης, αυξάνοντας την εντροπία και δυσκολεύοντας την ανάλυση πιθανών επιθέσεων κρυπτανάλυσης. Επομένως, όπως και το βήμα $\theta$, το $\rho$ είναι κρίσιμο για την ασφάλεια του SHA-3, βοηθώντας στο να καταστεί ασφαλέστερος από επιθέσεις που θα μπορούσαν να εκμεταλλευτούν την ομοιομορφία ή την προβλεψιμότητα στην διαδικασία κατακερματισμού. Η Εξισώση (2.19) περιγράφει τη λειτουργία του βήματος $\rho$.

$$A[i, j] = \text{ROTATE}\left(A'[i, j], r[i, j]\right), \quad [i, j] \leq 4 \tag{2.19}$$

- Βήμα $\pi$ $(pi)$: Αυτό το βήμα είναι κρίσιμο για τη διασφάλιση της διασποράς των δεδομένων μέσα στον πίνακα κατάστασης. Μετά την εφαρμογή του βήματος $\rho$, όπου τα bits στον πίνακα κατάστασης περιστρέφονται, το βήμα $\pi$ αναδιατάσσει αυτά τα bits σε νέες θέσεις. Η λογική πίσω από το βήμα $\pi$ είναι να πετύχει ακόμα μεγαλύτερη ανάμειξη των bits, μετακινώντας τα κάθετα και οριζόντια σε όλο τον πίνακα.

  Η αναδιάταξη στο βήμα $\pi$ γίνεται με βάση την Εξίσωση (2.20). Συγκεκριμένα, το νέο στοιχείο στη θέση $B[j, 2i + 3j]$ προκύπτει από το στοιχείο $A[i, j]$ του προηγούμενου πίνακα. Με αυτό τον τρόπο, το βήμα $\pi$ συμβάλλει στην αύξηση της εντροπίας μέσα στον πίνακα κατάστασης, διασφαλίζοντας ότι μια μικρή αλλαγή σε ένα bit μπορεί να επηρεάσει πολλά άλλα bits μετά από τους γύρους του αλγορίθμου, κάτι που αυξάνει την ασφάλεια κατά την παραγωγή του τελικού hash.

$$\text{B[j, 2i + 3j]} = \text{A[i, j]}, \quad [i, j] \leq 4 \tag{2.20}$$

- Βήμα $\chi$ $(chi)$: Αυτό το βήμα παίζει κεντρικό ρόλο στη μη-γραμμική τροποποίηση του πίνακα κατάστασης. Αυτό το βήμα επιδιώκει να προσθέσει επιπλέον ασφάλεια στον αλγόριθμο με το να εισαγάγει μια διαδικασία που εξασφαλίζει πως κάθε bit επηρεάζεται από τα άλλα bits της ίδιας γραμμής του πίνακα κατάστασης, μέσω μιας συνδυαστικής λογικής λειτουργίας.

  Στο βήμα $\chi$, κάθε bit $A[i, j]$ στον πίνακα κατάστασης $A$ ανανεώνεται χρησιμοποιώντας μια λογική λειτουργία XOR με τον συνδυασμό των bits στην ίδια γραμμή. Συγκεκριμένα, το νέο bit προκύπτει από την εφαρμογή του XOR

στο παλιό bit $A[i, y]$ με το αποτέλεσμα μιας λογικής AND λειτουργίας ανάμεσα στο αντίστροφο του bit $A[(i + 1)mod5, j]$ και το bit $A[(i + 2)mod5, j]$. Αυτό οδηγεί σε μια σύνθετη μη γραμμική αλλαγή που βελτιώνει την ανθεκτικότητα του αλγορίθμου σε επιθέσεις κρυπτανάλυσης.

Επειδή το βήμα $\chi$ είναι μη γραμμικό, προσθέτει στον αλγόριθμο μια δυσκολία πρόβλεψης και ανάλυσης, κάτι που είναι θεμελιώδες για την ασφάλεια ενός κρυπτογραφικού αλγορίθμου. Ανάλογα με την εφαρμογή των σταθερών και των λειτουργιών σε κάθε γύρο της μετάθεσης, το βήμα $\chi$ βοηθάει να διασφαλιστεί ότι κάθε bit του τελικού hash θα έχει επηρεαστεί από πολλαπλά bits του αρχικού μηνύματος, προσθέτοντας σημαντικο επίπεδο ασφάλειας. Η Εξισώση (2.20) περιγράφει τη λειτουργία του βήματος $\chi$.

$$A[i, j] = B[i, j] \,\mathrm{XOR}((-B[i + 1, j])ANDB[i + 2, j]), \quad [i, j] \leq 4 \tag{2.21}$$

- Βήμα $\iota$ $(iota)$: Είναι το τελευταίο βήμα σε κάθε γύρο της λειτουργίας μετάθεσης και εκτελείται με βάση την Εξίσωση (2.22). Στόχος του είναι να εισάγει μια μικρή ποσότητα ασυμμετρίας στη διαδικασία, βοηθώντας στην αποφυγή επιθέσεων που εκμεταλλεύονται την συμμετρία των προηγούμενων βημάτων. Αυτό πραγματοποιείται με την προσθήκη μιας στρογγυλοποιημένης σταθεράς (rounding constant) στο πρώτο στοιχείο του πίνακα κατάστασης $A[0, 0]$.

$$A[0, 0] = A[0, 0] \,\mathrm{XOR}\, RC_i \tag{2.22}$$

Η στρογγυλοποιημένη σταθερά $RC_i$ είναι μοναδική για κάθε γύρο της λειτουργίας μετάθεσης, διασφαλίζοντας ότι κάθε γύρος είναι μοναδικός και προσθέτει μια διαφορετική διάσταση στην κρυπτογραφική δύναμη του αλγορίθμου. Οι σταθερές αυτές είναι προκαθορισμένες και συντίθενται από 64-bit τιμές οπως φαινονται στον Πίνακα 2.5. Η προσθήκη της στρογγυλοποιημένης σταθεράς στον πίνακα κατάστασης ενισχύει την ασφάλεια ενάντια σε δυνατές επιθέσεις επαναλαμβανόμενων σχημάτων που θα μπορούσαν να εμφανιστούν αν οι γύροι ήταν πλήρως συμμετρικοί. Το βήμα $\iota$ προσθέτει ένα επίπεδο προστασίας από επιθέσεις που στοχεύουν στην εύρεση αδυναμιών στη συμμετρική δομή του αλγορίθμου, επιτρέποντας στον SHA-3

να αντιστέκεται σε μια ευρύτερη γκάμα επιθέσεων και να παρέχει μια πιο αξιόπιστη κρυπτογραφική συνάρτηση κατακερματισμού.

Πίνακας 2.5: Σταθερές τιμές $RC_i$

| | | | |
|---|---|---|---|
| $RC_0$ | 0x0000000000000001 | $RC_{12}$ | 0x000000008000808B |
| $RC_1$ | 0x0000000000008082 | $RC_{13}$ | 0x800000000000008B |
| $RC_2$ | 0x800000000000808A | $RC_{14}$ | 0x8000000000008089 |
| $RC_3$ | 0x8000000080008000 | $RC_{15}$ | 0x8000000000008003 |
| $RC_4$ | 0x000000000000808B | $RC_{16}$ | 0x8000000000008002 |
| $RC_5$ | 0x0000000080000001 | $RC_{17}$ | 0x8000000000000080 |
| $RC_6$ | 0x8000000080008081 | $RC_{18}$ | 0x000000000000800A |
| $RC_7$ | 0x8000000000008009 | $RC_{19}$ | 0x800000008000000A |
| $RC_8$ | 0x000000000000008A | $RC_{20}$ | 0x8000000080008081 |
| $RC_9$ | 0x0000000000000088 | $RC_{21}$ | 0x8000000000008080 |
| $RC_{10}$ | 0x0000000080008009 | $RC_{22}$ | 0x0000000080000001 |
| $RC_{11}$ | 0x000000008000000A | $RC_{23}$ | 0x8000000080008008 |

Στο πλαίσιο του SHA-3, η βελτιστοποίηση της χρήσης των πόρων είναι κρίσιμη, ειδικά όταν πρόκειται για υλοποιήσεις σε υλικό όπως FPGA ή ASIC. Το βήμα $\iota$ αντί να αποθηκεύεται και να χρησιμοποιείται ολόκληρη η τιμή της στρογγυλοποιημένης σταθεράς, η οποία είναι ένα 64-bit νούμερο, είναι δυνατή η αποθήκευση μόνο των μη μηδενικών bits. Αυτό συμβαίνει επειδή πολλά από τα bits είναι μηδέν και άρα δεν συνεισφέρουν στην τιμή XOR που υπολογίζεται στο βήμα $\iota$. Με την εφαρμογή αυτής της απλοποίησης, μπορούν να εξοικονομηθούν πόροι στον πίνακα κατάστασης και να μειωθεί ο αριθμός των απαιτούμενων πράξεων XOR, οι οποίες είναι ουσιαστικές για την εφαρμογή της κρυπτογραφικής λειτουργίας.

### 2.4.3 Σύνοψη σταδίων υπολογισμών

Ο Αλγόριθμος 2.4.1 παρουσιάζει τα στάδια υπολογισμών του SHA-3. Η διαδικασία ξεκινά με την αρχικοποίηση ενός πίνακα τριών διαστάσεων $5 \times 5$ λέξεων των $w$ bits (κατάσταση $S$), ο οποίος αρχικά περιέχει μόνο μηδενικά. Στη συνέχεια, το εισερχόμενο μήνυμα $M$ υποβάλλεται σε κατάλληλη διαδικασία padding, ώστε να διαμορφωθεί το μήκος του σύμφωνα με τις απαιτήσεις του αλγορίθμου. Το μήνυμα διασπάται σε διαδοχικά μπλοκ των $r$ bits, τα οποία επεξεργάζονται διαδοχικά και κάθε μπλοκ ενσωματώνεται στον πίνακα κατάστασης $S$ μέσω της διαδικασίας απορρόφησης.

**Algorithm 2.4.1** Ο αλγόριθμος SHA-3 (Keccak)

**Require:** Μήνυμα $M$
**Ensure:** Συμπύκνωμα μηνύματος
**function** SHA-3($M$)
    Αρχικοποίηση του πίνακα κατάστασης $S$ με μηδενικά
    Εφαρμογή Padding στο $M$
    Χωρίστε το $M$ σε μπλοκ $r$-bit και επεξεργαστείτε κάθε μπλοκ
    **for** κάθε μπλοκ $m$ του $M$ **do**
        Προσάρτηση του $m$ στο $S$
    **end for**
    **for** $i = 1$ to $12 + 2l$ **do** ▷ $l = \log_2(b)/2$, $b = 1600$ για SHA-3
        Step-θ($S$) ▷ Λειτουργία $\theta$
        **for** $x \leftarrow 0$ **to** $4$ **do**
            **for** $y \leftarrow 0$ **to** $4$ **do**
                $A[x, y] \leftarrow A[x, y] \oplus D[x]$
            **end for**
        **end for**
        Step-ρ($S$) ▷ Λειτουργία $\rho$
        **for** $x \leftarrow 0$ **to** $4$ **do**
            **for** $y \leftarrow 0$ **to** $4$ **do**
                $A[x, y] \leftarrow \text{rot}(A[x, y], r[x, y])$
            **end for**
        **end for**
        Step-π($S$) ▷ Λειτουργία $\pi$
        **for** $x \leftarrow 0$ **to** $4$ **do**
            **for** $y \leftarrow 0$ **to** $4$ **do**
                $B[y, (2x + 3y) \bmod 5] \leftarrow A[x, y]$
            **end for**
        **end for**
        Step-χ($S$) ▷ Λειτουργία $\chi$
        **for** $x \leftarrow 0$ **to** $4$ **do**
            **for** $y \leftarrow 0$ **to** $4$ **do**
                $A[x, y] \leftarrow B[x, y] \oplus (\neg B[x + 1 \bmod 5, y] \wedge B[x + 2 \bmod 5, y])$
            **end for**
        **end for**
        Step-ι($S$, $i$) ▷ Λειτουργία $\iota$
        **for** $x \leftarrow 0$ **to** $4$ **do**
            **for** $y \leftarrow 0$ **to** $4$ **do**
                $A[0, 0] \leftarrow A[0, 0] \oplus RC$
            **end for**
        **end for**
    **end for**
    **return** Εξαγωγή του συμπυκνώματος από το $S$
**end function**

Η επεξεργασία των μπλοκ εισόδου στον SHA-3 ακολουθεί πέντε διαδοχικά βήματα: $\theta$ $(theta)$ $\rightarrow$ $\rho$ $(rho)$ $\rightarrow$ $\pi$ $(pi)$ $\rightarrow$ $\chi$ $(chi)$ $\rightarrow$ $\iota$ $(iota)$. Κάθε ένα από τα βήματα αυτά εφαρμόζει συγκεκριμένες μαθηματικές και λογικές λειτουργίες στον πίνακα κατάστασης, μετασχηματίζοντας σταδιακά την πληροφορία του μηνύματος με τρόπο που ενισχύει την κρυπτογραφική ασφάλεια της τελικής τιμής κατακερματισμού. Συγκεκριμένα, η λειτουργία $\theta$ επιτυγχάνει τη διάχυση της πληροφορίας σε όλο τον πίνακα, ενώ η $\rho$ εφαρμόζει κυκλικές μετατοπίσεις (rotations) σε κάθε στοιχείο του. Η $\pi$ αναδιατάσσει τα δεδομένα με βάση προκαθορισμένη ακολουθία, η $\chi$ εισάγει μη γραμμικές λογικές πράξεις για τον περαιτέρω μετασχηματισμό της κατάστασης, και τέλος, η $\iota$ προσθέτει μία στρογγυλοποιημένη σταθερά (round constant) στον πίνακα κατάστασης, διασφαλίζοντας ότι κάθε γύρος της επεξεργασίας είναι μοναδικός.

Μετά την ολοκλήρωση όλων των γύρων επεξεργασίας, η τελική φάση του αλγορίθμου παράγει το μήνυμα-συμπύκνωμα (digest) εξάγοντας τα απαιτούμενα $bit$ από τον πίνακα κατάστασης $S$. Ο SHA-3 παρέχει μια εξαιρετικά ασφαλή και ευέλικτη λύση για την παραγωγή κρυπτογραφικών κατακερματισμών, επιτρέποντας μεταβλητό μήκος εξόδου και προσφέροντας έναν ισχυρό μηχανισμό ασφάλειας για σύγχρονες κρυπτογραφικές εφαρμογές.

## 2.5 Σύνοψη κεφαλαίου

Σε αυτό το κεφάλαιο αναπτύσσεται η θεωρητική υπόσταση των αλγορίθμων της οικογένειας ασφαλούς κατακερματισμού. Παρέχεται μια σε βάθος ανάλυση της δομής, της λειτουργικότητας και της ασφάλειας των αλγορίθμων SHA-1, SHA-2, και SHA-3, αναλύοντας τα πλεονεκτήματα και τα μειονοκτήματα τους. Δόθηκε ιδιαίτερη έμφαση στα κύρια χαρακτηριστικά του αλγόριθμου SHA-3, αναγνωρίζοντας τη σημασία της καινοτομίας του στην αυξημένη ανθεκτικότητα σε κρυπταναλυτικές επιθέσεις και τη συμβολή του στην ασφάλεια των δεδομένων.

# Κεφάλαιο 3

# Τεχνική επιτάχυνσης διοχετεύσεων υλικού

Στο προηγούμενο κεφάλαιο αναλύθηκε η ιστορική εξέλιξη και η συνεχής ανάπτυξη των αλγορίθμων κατακερματισμού SHA, καθώς και τα σημαντικότερα χαρακτηριστικά και οι διαφορές μεταξύ τους. Στο παρόν κεφάλαιο προτείνεται μια τεχνική επιτάχυνσης βασισμένη στη διασωλήνωση (pipelining) για τον αλγόριθμο SHA-3. Συγκεκριμένα, αναλύεται μια μεθοδολογία κατά την οποία προστίθεται ένας επιπλέον καταχωρητής μετά το βήμα $\theta$ $(theta)$ στη συνάρτηση $f$. Σε αυτό το κεφάλαιο[1], παρουσιάζονται αναλυτικά η σχεδίαση, η υλοποίηση και η αξιολόγηση της εν λόγω τεχνικής.

## 3.1 Περίληψη

Στο σύγχρονο ψηφιακό περιβάλλον, η μετάδοση πληροφοριών σε μορφή κειμένου, εικόνας, βίντεο και ήχου πραγματοποιείται μέσω πολλαπλών, συχνά μη ασφαλών διαδρομών. Η ψηφιακή αυτή μεταφορά δεδομένων καθιστά επιτακτική την ανάγκη για ασφαλή μετάδοση, με έμφαση στην εμπιστευτικότητα και την ακεραιότητα των δεδομένων. Η προστασία των μεταδιδόμενων πληροφοριών μπορεί να επιτευχθεί μέσω της χρήσης αλγορίθμων κατακερματισμού, οι οποίοι διασφαλίζουν την ακεραιότητα και την αυθεντικότητα των δεδομένων κατά τη μεταφορά.

[1]Το κεφάλαιο έχει δημοσιευθεί στο άρθρο [120]

Ο προηγμένος κρυπτογραφικός αλγόριθμος SHA-3 θεωρείται ιδιαίτερα ανθεκτικός σε επιθέσεις κρυπτανάλυσης και προτιμάται ευρέως για τη μακροπρόθεσμη ασφάλεια που παρέχει σε πλήθος εφαρμογών. Ωστόσο, η ραγδαία αύξηση του όγκου των μεταδιδόμενων δεδομένων δημιουργεί την ανάγκη για ακόμη πιο αποτελεσματικές και βέλτιστες υλοποιήσεις, ικανές να ανταποκριθούν σε απαιτήσεις υπολογισμών σε πραγματικό χρόνο.

Στο πλαίσιο αυτό, τα FPGA αναδεικνύονται ως ιδανική τεχνολογική επιλογή για τη βελτίωση των επιδόσεων του αλγορίθμου, προσφέροντας σημαντικά πλεονεκτήματα όσον αφορά τη ρυθμαπόδοση, τη συχνότητα λειτουργίας, την αποδοτικότητα, τη μείωση της απαιτούμενης επιφάνειας υλοποίησης και την κατανάλωση ενέργειας. Η ανάπτυξη αναβαθμισμένων αρχιτεκτονικών υλοποιήσεων του SHA-3 παραμένει ένας ενεργός τομέας έρευνας, με συνεχείς προσπάθειες για τη βελτιστοποίηση της απόδοσης και της ασφάλειας.

Σε αυτό το κεφάλαιο, επικεντρωθήκαμε στη βελτίωση των μετρικών επιδόσεων υλικού του SHA-3, δίνοντας έμφαση στη ρυθμαπόδοση και την αποδοτικότητα, καθώς και στη μείωση του κόστους επιφάνειας σε slices, για όλα τα μήκη μεγέθους εξόδου (224, 256, 384 και 512 bits). Συγκεκριμένα, παρουσιάζεται και αναλύεται μια μεθοδολογία κατά την οποία προστίθεται ένας επιπλέον καταχωρητής μετά το βήμα $\theta$ στη συνάρτηση $f$. Ο προτεινόμενος σχεδιασμός επιτυγχάνει σημαντική αύξηση στη ρυθμαπόδοση και την αποδοτικότητα, προσφέροντας βελτιστοποιημένες υλοποιήσεις για όλες τις εκδόσεις του SHA-3.

## 3.2 Εισαγωγή

Η ανάγκη για αξιόπιστη και ασφαλή μετάδοση ευαίσθητων δεδομένων έχει καταστεί ιδιαίτερα επιτακτική τα τελευταία χρόνια. Η κρυπτογραφία αποτελεί μια θεμελιώδη τεχνική για την αποθήκευση, προστασία και διασφάλιση πληροφοριών έναντι μη εξουσιοδοτημένης πρόσβασης κατά τη μεταφορά τους σε ψηφιακά δίκτυα. Οι τρεις αυτοί στόχοι, εμπιστευτικότητα, ακεραιότητα και διαθεσιμότητα, επιτυγχάνονται μέσω της εφαρμογής κατάλληλων κρυπτογραφικών μεθόδων. Ενδεικτικά, τομείς όπως η υγειονομική περίθαλψη, ο στρατός, η δημόσια διοίκηση, η βιομηχανία, τα εκπαιδευτικά ιδρύματα και οι ιδιωτικές επιχειρήσεις συλλέγουν και διαχειρίζονται τεράστιες ποσότητες προσωπικών ψηφιακών δεδομένων, τα οποία αποθηκεύονται και διακινούνται σε δικτυακά περιβάλλοντα. Ως εκ τούτου, η χρήση

κρυπτογραφικών αλγορίθμων έχει αυξηθεί σημαντικά τα τελευταία χρόνια, λόγω της ικανότητάς τους να διασφαλίζουν υψηλά επίπεδα ασφάλειας για διάφορες μορφές ψηφιακών δεδομένων, όπως φωτογραφίες, κείμενο, βίντεο και ήχο [121, 122].

Ένας θεμελιώδης τομέας της κρυπτογραφίας είναι ο κατακερματισμός. Ο κατακερματισμός αναφέρεται στη διαδικασία υπολογισμού μιας συμβολοσειράς (τιμής κατακερματισμού) σταθερού μήκους, μέσω μιας καθορισμένης κατακερματιστικής συνάρτησης (hash function), ανεξαρτήτως του μεγέθους της εισόδου. Η συμβολοσειρά εξόδου διατηρεί πάντα το ίδιο μήκος για έναν συγκεκριμένο αλγόριθμο κατακερματισμού, ανεξάρτητα από το μήκος ή το περιεχόμενο της εισόδου. Ιδανικά, κάθε τιμή κατακερματισμού αντιστοιχεί μοναδικά σε μια συγκεκριμένη είσοδο, ενώ ακόμη και μία ελάχιστη μεταβολή στην είσοδο (π.χ. η αλλαγή ενός byte) προκαλεί σημαντική μεταβολή της εξόδου. Αυτές οι ιδιότητες καθιστούν τον κατακερματισμό κεντρικό στοιχείο στις υποδομές ΤΠΕ, ενώ οι σχετικοί αλγόριθμοι αποτελούν αναπόσπαστο μέρος της ψηφιακής μας καθημερινότητας [123, 124].

Ο κατακερματισμός αποτελεί αναπόσπαστο στοιχείο κάθε συστήματος ελέγχου ταυτότητας, από τα τοπικά λειτουργικά συστήματα μέχρι προηγμένες υπηρεσίες, όπως το cloud banking ή οι πλατφόρμες email web. Επιπλέον, διαδραματίζει κρίσιμο ρόλο στην επαλήθευση της ακεραιότητας των δεδομένων, είτε πρόκειται για τοπικά συστήματα αρχείων, όπως το Zettabyte File System (ZFS) [125], το οποίο αποθηκεύει έναν κατακερματισμό για κάθε μπλοκ δεδομένων, είτε για λειτουργικά συστήματα που διατηρούν κατακερματισμούς για κάθε κρίσιμο αρχείο. Δημοφιλή λειτουργικά συστήματα και συστήματα αρχείων, όπως τα Windows, Linux και FreeBSD, διατηρούν βάσεις δεδομένων με κατακερματισμούς για θεμελιώδη στοιχεία του συστήματος, ενώ τα συστήματα ανίχνευσης εισβολής (IDS) υπολογίζουν και συγκρίνουν κατακερματισμούς αρχείων με πρότυπα για την ανίχνευση τροποποιήσεων ή κακόβουλων επεμβάσεων. Επιπρόσθετα, ο κατακερματισμός χρησιμοποιείται ευρέως και στη μετάδοση δεδομένων, διασφαλίζοντας την ακεραιότητα σε επίπεδο πρωτοκόλλων, από το IP μέχρι την ασφαλή περιήγηση στο διαδίκτυο με χρήση των πρωτοκόλλων HTTPS/TLS.

Ο κατακερματισμός χρησιμοποιείται επίσης ευρέως για την ευρετηρίαση σε βάσεις δεδομένων, διευκολύνοντας την ταχύτερη εκτέλεση ερωτημάτων, μια ιδιότητα που καθίσταται ολοένα και πιο απαραίτητη στη σύγχρονη εποχή, λόγω των τεράστιων ποσοτήτων δεδομένων που παράγονται και επεξεργάζονται καθημερινά. Αξίζει να σημειωθεί ότι όλοι οι χρήστες έρχονται σε επαφή με

διάφορες τεχνικές κατακερματισμού, συχνά χωρίς να το αντιλαμβάνονται, καθώς αυτές ενσωματώνονται σε πλήθος τεχνολογικών εφαρμογών. Παράλληλα, οι προγραμματιστές συστημάτων ασφάλειας πληροφορικής καλούνται να σχεδιάζουν, να υλοποιούν και να βελτιστοποιούν νέους, ισχυρότερους και ασφαλέστερους αλγορίθμους κατακερματισμού, προκειμένου να ανταποκριθούν στις διαρκώς αυξανόμενες απαιτήσεις των πληροφοριακών συστημάτων [126].

Σήμερα, τα παλαιότερα πρότυπα συναρτήσεων κατακερματισμού έχουν καταστεί ευάλωτα σε διάφορους τύπους επιθέσεων. Μέχρι σήμερα έχουν καταγραφεί πλήθος επιτυχημένων επιθέσεων εναντίον των αλγορίθμων κατακερματισμού SHA-1 [25] και, σε μικρότερο βαθμό, SHA-2 [127, 128]. Σε ανταπόκριση αυτών των ευρημάτων, το Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας (NIST) προχώρησε στη διερεύνηση και υιοθέτηση νέων, πιο ασφαλών αλγορίθμων κατακερματισμού, επιλέγοντας τελικά τον SHA-3 (Keccak), ο οποίος προσφέρει σημαντικά ενισχυμένο επίπεδο ασφάλειας [30, 129]. Οι νεότερες αυτές συναρτήσεις κατακερματισμού κάνουν χρήση μεγαλύτερων μεγεθών εξόδου και πιο σύνθετων αλγοριθμικών δομών, γεγονός που καθιστά πολύ δυσκολότερη την εύρεση συγκρούσεων ή άλλων αδυναμιών από πλευράς επιτιθέμενων. Οι συναρτήσεις κατακερματισμού χρησιμοποιούνται πλέον εκτενώς σε κρίσιμες εφαρμογές ασφάλειας, όπως ο κώδικας ελέγχου ταυτότητας κατακερματισμένων μηνυμάτων [130], η ασφάλεια δικτύου [131], οι ψηφιακές υπογραφές [132], οι ασφαλείς ηλεκτρονικές συναλλαγές [133] και η υποδομή δημόσιου κλειδιού [134].

Το νέο πρότυπο SHA-3 προέκυψε μέσα από διαγωνισμό που διοργάνωσε το Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας (NIST) το 2011, με στόχο την επιλογή ενός νέου ασφαλούς αλγορίθμου κατακερματισμού. Οι ανοιχτοί διαγωνισμοί έχουν καθιερωθεί ως μέθοδοι επιλογής για κρυπτογραφικά πρότυπα διεθνώς. Το 2012, το NIST ανακοίνωσε ότι η συνάρτηση κατακερματισμού Keccak θα αποτελούσε το νέο πρότυπο SHA-3. Ο SHA-3 διακρίνεται για την υψηλή ρυθμαπόδοση και αποδοτικότητα σε υλοποιήσεις υλικού, τόσο σε μονάδες επεξεργασίας γραφικών όσο και σε συστοιχίες επιτόπια προγραμματιζόμενων πυλών (FPGA) [135].

Τα ενσωματωμένα συστήματα επεξεργασίας που βασίζονται σε FPGA προσφέρουν σημαντικούς υπολογιστικούς πόρους για την κάλυψη των αυξανόμενων απαιτήσεων ασφάλειας [136]. Επιπλέον, τα FPGA είναι γνωστά για τις δυνατότητες υψηλής απόδοσης και χαμηλής κατανάλωσης ενέργειας, καθιστώντας τα ιδανικά για ενσωματωμένες εφαρμογές όπου ο διαθέσιμος χώρος και η ισχύς είναι περιορισμένα [137, 138]. Λόγω της διαρκώς αυξανόμενης ανάγκης για ασφαλή

συστήματα, τα FPGA έχουν καταστεί ελκυστική επιλογή για την υλοποίηση χαρακτηριστικών ασφαλείας όπως η κρυπτογράφηση, ο έλεγχος ταυτότητας και η ανίχνευση εισβολής. Συνεπώς, η ερευνητική κοινότητα της κρυπτογραφίας επικεντρώνεται στον SHA-3, ο οποίος προσφέρει ευελιξία και υψηλή απόδοση σε υλοποιήσεις υλικού [139, 140].

Ακολουθεί μια περίληψη των συνεισφορών που δίνονται σε αυτό το κεφάλαιο:

- Προτείνουμε μια νέα τεχνική επιτάχυνσης και βελτιστοποίησης που βασίζεται στη διασωλήνωση (pipelining) για τον αλγόριθμο SHA-3. Η προτεινόμενη μέθοδος τοποθετεί έναν επιπλέον καταχωρητή μετά το βήμα $\theta$ στη συνάρτηση $f$, επιτυγχάνοντας σημαντική επιτάχυνση της επεξεργασίας και βελτίωση στις μετρικές απόδοσης, ενώ ταυτόχρονα μειώνει το κόστος επιφάνειας (σε slices) των συσκευών.
- Η ορθότητα του προτεινόμενου σχεδιασμού επιβεβαιώθηκε με τη χρήση έγκυρων δοκιμαστικών παραδειγμάτων που παρέχονται από το Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας (NIST) του Υπουργείου Εμπορίου των Ηνωμένων Πολιτειών.
- Παράλληλα, πραγματοποιήσαμε εκτεταμένη αξιολόγηση και ανάλυση, συγκρίνοντας το μέγεθος επιφάνειας (slices), τη ρυθμαπόδοση, τη συχνότητα λειτουργίας και τη συνολική αποδοτικότητα της προτεινόμενης αρχιτεκτονικής με αντίστοιχες λύσεις που έχουν δημοσιευθεί στη διεθνή βιβλιογραφία.

Το υπόλοιπο κεφάλαιο οργανώνεται ως εξής: Στην ενότητα 3.3, παρουσιάζονται οι σχετικές εργασίες της βιβλιογραφίας. Στην ενότητα 3.4, περιγράφεται αναλυτικά η νέα προτεινόμενη τεχνική βελτιστοποίησης υλικού του αλγορίθμου SHA-3 σε περιβάλλον FPGA. Στην ενότητα 3.5, παρουσιάζονται τα πειραματικά αποτελέσματα της μελέτης μας. Στην ενότητα 3.6, ακολουθεί συζήτηση των αποτελεσμάτων και σύγκριση με άλλες σχετικές μελέτες. Τέλος, στην ενότητα 3.7 συνοψίζονται τα συμπεράσματα της εργασίας.

## 3.3 Σχετικές εργασίες διοχετεύσεων υλικού

Η ερευνητική κοινότητα της κρυπτογραφίας έχει πραγματοποιήσει σημαντική πρόοδο στη βελτιστοποίηση αρχιτεκτονικών και τεχνικών για τον αλγόριθμο SHA-3 σε συσκευές FPGA[141]. Κάθε μία από αυτές τις προσεγγίσεις στοχεύει στη βελτίωση της ρυθμαπόδοσης του SHA-3, ενώ παράλληλα επιδιώκει τη μείωση της απαιτούμενης επιφάνειας υλοποίησης και της κατανάλωσης ισχύος[142–145]. Παρά ταύτα, εξακολουθεί να παραμένει επιτακτική η ανάγκη για περαιτέρω βελτίωση των μετρικών απόδοσης, ιδιαίτερα όσον αφορά τη ρυθμαπόδοση και την ελαχιστοποίηση της επιφάνειας υλοποίησης. Στην ενότητα που ακολουθεί, παρουσιάζονται και συζητούνται άλλες ερευνητικές μελέτες που είναι συγκρίσιμες με τη δική μας προσέγγιση.

Οι συγγραφείς της εργασίας [146], πρότειναν μια τεχνική διασωλήνωσης για τον SHA-3 512 bit. Ο προτεινόμενος σχεδιασμός εφαρμόστηκε στο FPGA Virtex-5. Η προτεινόμενη αρχιτεκτονική χρειάζεται επιφάνεια σε slices 2326, επιτυγχάνει συχνότητα 306 MHz, ρυθμαποδοση 5,56 Gbps και ρυθμό αποδοτικότητας 2,40 Mbps/slices.

Στο [147], οι συγγραφείς πρότειναν μια τεχνική με διασωλήνωση δύο σταδίων για το SHA-3 256 bit. Ο προτεινόμενος σχεδιασμός εφαρμόστηκε στο FPGA Virtex-5. Πέτυχαν μέγιστη συχνότητα 317,11 MHz, ρυθμαποδοση 12,68 Gbps, επιφάνεια σε slices 4793 και αποδοτικότητα 2,71 Mbps/slices.

Οι συγγραφείς [148] πρότειναν ένα σχέδιο διασωλήνωσης για το SHA-3 512 bit. Η προτεινόμενη μέθοδος εφαρμόστηκε στο FPGA Virtex-5. Η προτεινόμενη αρχιτεκτονική επιτυγχάνει τον ρυθμό συχνότητας 273 MHz, χρειάζεται επιφάνεια σε slices 1163, ρυθμαποδοση 7,80 Gbps και αποδοτικότητα 6,06 Mbps/slices.

Στο [149], οι συγγραφείς πρότειναν έναν σχεδιασμό δύο σταδίων για το SHA-3 512 bit σε τρεις συσκευές FPGA. Η προτεινόμενη αρχιτεκτονική υλοποιήθηκε σε πλακέτες FPGA, Virtex-4, Virtex-5 και Virtex-6. Τα αποτελέσματα δείχνουν ότι η προτεινόμενη μέθοδος για το SHA-3 512 bit έχει πιο πολλά υποσχόμενα αποτελέσματα με το Virtex-6. Πετυχε μέγιστο ρυθμό συχνότητας 391 MHz, ρυθμαποδοση 18,76 Gbps, επιφάνεια σε slices 2296 και αποδοτικότητα 8,17 Mbps/slices.

Στην μελέτη [150], οι συγγραφείς πρότειναν μια τεχνική διασωλήνωσης δύο σταδίων για το SHA-3 256 bit. Η γλώσσα προγραμματισμού που χρησιμοποιηθηκε ηταν η Very High Speed Integrated Circuit Hardware Description Language (VHDL) και

υλοποιείται στις πλακέτες Virtex-5, Virtex-6 και Virtex-7 FPGA. Τα αποτελέσματα εδειξαν ότι ο προτεινόμενος σχεδιασμός επιτυγχάνει καλύτερα αποτελέσματα με την πλακέτα Virtex-7. Η μέθοδος τους πετυχε ρυθμαποδοση 20,8 Gbps, συχνότητα 434 MHz, επιφάνεια σε slices 1618 και αποδοτικότητα 12,90 Mbps/slices.

Οι συγγραφείς στην εργασια [151] πρότειναν ένα σχέδιο διασωλήνωσης για τα SHA-3 256 bit και SHA-3 512 bit. Η προτεινόμενη μέθοδος εφαρμόστηκε στις πλακέτες Virtex-5 και Virtex-6 FPGA. Η προτεινόμενη μέθοδος για το SHA-3 256 bit χρειάζεται 1456 επιφάνεια σε slices, ρυθμαποδοση 14,942 Gbps και αποδοτικότητα 10,26 Mbps/slices με την πλακέτα Virtex-6, και για την υλοποιηση με SHA-3 512 bit χρειάζεται 1263 επιφάνεια σε slices, ρυθμαποδοση 8,114 Gbps και αποδοτικότητα 6,42 Mbps/slices με την πλακέτα Virtex-6.

Στο [152] πρότειναν μια αρχιτεκτονική που υποστηρίζει όλα τα μήκη μεγέθους εξόδου (224, 256, 384 και 512 bit) των κρυπτογραφικών συναρτήσεων κατακερματισμού SHA-2 και SHA-3. Ο προτεινόμενος σχεδιασμός εφαρμόστηκε και επαληθεύτηκε στην πλακέτα Stratix IV, χρησιμοποιώντας τον επεξεργαστή NIOS II. Η προτεινόμενη αρχιτεκτονική για το SHA-3 χρειάζεται 5363 επιφάνεια σε slices και επιτυγχάνει τον υψηλότερο ρυθμό συχνότητας 110 MHz.

Οι συγγραφείς του [153] παρουσίασαν ένα σχέδιο για τον αλγόριθμο SHA-3 512 bit. Αυτός ο σχεδιασμός εφαρμόστηκε στην πλακέτα Virtex-5 FPGA. Στο Virtex-5, η προτεινόμενη αρχιτεκτονική απαιτούσε 1680 επιφάνεια σε slices και συχνότητα 387 MHz. Η προτεινόμενη αρχιτεκτονική επιτυγχάνει ρυθμαποδοση 8,06 Gbps και αποδοτικότητα 4,91 Mbps/slices.

Ο Πίνακας 3.1 συνοψίζει τις υλοποιήσεις που έχουν σχεδιαστεί με τη μέθοδο διασωλήνωσης δύο σταδίων για τον αλγόριθμο SHA-3. Η πλειονότητα των προηγούμενων μελετών εστιάζει στην τοποθέτηση του καταχωρητή μετά το βήμα $\pi$, κάνοντας χρήση της κλασικής γεννήτριας RC 64-bit.

Πίνακας 3.1: Σύνοψη των δημοσιευμένων προσεγγίσεων της τεχνικής διασωλήνωσης για τον αλγόριθμο SHA-3.

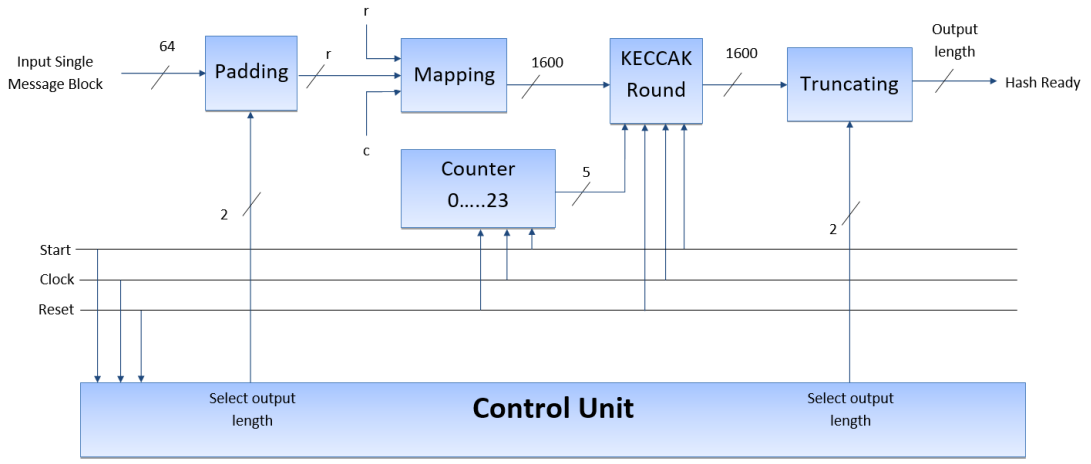
| **Εργασία** | **Μήκος εξόδου** | **Τοποθέτηση διασωλήνωσης** | **RC γεννήτρια** |
|---|---|---|---|
| [146] | SHA-3 512 bit | - | 64-bit |
| [147] | SHA-3 256 bit | μετά το βημα $\pi$ | 64-bit |
| [148] | SHA-3 512 bit | μετά το βημα $\pi$ | 64-bit |
| [149] | SHA-3 512 bit | μετά το βημα $\pi$ | 64-bit |
| [150] | SHA-3 256 bit | μετά το βημα $\pi$ | 64-bit |
| [151] | SHA-3 256 bit | μετά το βημα $\pi$ | 64-bit |
| [151] | SHA-3 512 bit | μετά το βημα $\pi$ | 64-bit |
| [152] | SHA-3 | - | 64-bit |
| [153] | SHA-3 512 bit | - | 64-bit |

Αυτή η εργασία στοχεύει στη συγκριτική αξιολόγηση της ρυθμαπόδοσης (Gbps) και της αποδοτικότητας (Mbps/slices) κατά την εισαγωγή του καταχωρητή είτε μετά το βήμα $\pi$ είτε μετά το βήμα $\theta$ στη διαδικασία κατακερματισμού της συνάρτησης $f$, σε συνδυασμό με τη νέα διαμόρφωση της γεννήτριας RC 7-bit, για όλα τα μήκη εξόδου (224, 256, 384 και 512 bit).

Τα εκτεταμένα πειράματά μας καταδεικνύουν ότι οι επιδόσεις επηρεάζονται άμεσα από το μήκος του κρίσιμου μονοπατιού της συνάρτησης $f$, το οποίο μειώνεται σημαντικά όταν ο καταχωρητής τοποθετείται μετά το βήμα $\theta$, σε συνδυασμό με τη νέα, απλοποιημένη δομή της γεννήτριας RC. Η προτεινόμενη τεχνική βελτιστοποίησης υπερτερεί έναντι των προηγούμενων προσεγγίσεων στα μετρικά απόδοσης και μπορεί να υιοθετηθεί ως βέλτιστη στρατηγική για υλοποιήσεις σε πλακέτες FPGA.

## 3.4 Τεχνικές βελτιστοποίησης αγωγών υλικού

Ο βασικός στόχος της παρούσας εργασίας είναι η επίτευξη υψηλότερης ρυθμαπόδοσης (Gbps) και αποδοτικότητας (Mbps/slices) στο προτεινόμενο σύστημα, με παράλληλη ελαχιστοποίηση των απαιτούμενων υλικών πόρων. Η επίτευξη του στόχου αυτού βασίζεται στην εισαγωγή του καταχωρητή μετά το βήμα $\theta$, καθώς και στη νέα, απλοποιημένη μορφή της προτεινόμενης γεννήτριας RC.




Σχήμα 3.1: Η προτεινόμενη προσέγγιση με τη μέθοδο διασωλήνωσης δύο σταδίων για τον αλγόριθμο SHA-3.

Στο Σχήμα 3.1 παρουσιάζεται η αρχιτεκτονική του συστήματος της προτεινόμενης τεχνικής βελτιστοποίησης διασωλήνωσης. Η σχεδίαση αποτελείται από διακριτές λειτουργικές μονάδες, καθεμία με εξειδικευμένο ρόλο:

1. Μονάδα πλήρωσης (Padding): Διασφαλίζει τη σωστή προεπεξεργασία του μηνύματος εισόδου, εφαρμόζοντας το απαραίτητο παδδινγ ώστε το μήκος του μηνύματος να συμμορφώνεται με τις απαιτήσεις του αλγορίθμου.

2. Μονάδα χαρτογράφησης (Mapping): Χαρτογραφεί το προεπεξεργασμένο μήνυμα σε πίνακα καταστάσεων κατάλληλο για τους γύρους SHA-3.

3. Γύρος SHA-3 (KECCAK Round): Αποτελεί τον πυρήνα της αρχιτεκτονικής, εκτελώντας τη συνάρτηση sponge που μετασχηματίζει το μήνυμα εισόδου στην τελική τιμή κατακερματισμού.

4. Μονάδα περικοπής (Truncating): Αναλαμβάνει την περικοπή του παραγόμενου κατακερματισμού ώστε να προκύπτει το επιθυμητό μήκος εξόδου.

5. Μονάδα ελέγχου (Control Unit): Συντονίζει και διαχειρίζεται τη ροή δεδομένων μεταξύ των επιμέρους μονάδων, διασφαλίζοντας την ορθή αλληλουχία των λειτουργιών.

Το μήκος του μηνύματος εισόδου στο προτεινόμενο σύστημα είναι 64 bits, ενώ το επιθυμητό μήκος εξόδου μπορεί να επιλεγεί ανάλογα με τις απαιτήσεις της

εφαρμογής. Οι δυνατές επιλογές για το μήκος εξόδου παρουσιάζονται συνοπτικά στον Πίνακα 3.2.

Πίνακας 3.2: Επιλογή μήκος εξόδου.

| **Τιμή** | 00 | 01 | 10 | 11 |
|---|---|---|---|---|
| **Έξοδος κατακερματισμού** | 224 | 256 | 384 | 512 |

Η μονάδα πλήρωσης (padding) διασφαλίζει ότι το μήνυμα εισόδου αποκτά κατάλληλο μέγεθος ώστε να είναι επεξεργάσιμο από τον αλγόριθμο. Ειδικά στην περίπτωση του SHA-3, η μονάδα πλήρωσης συνίσταται στην προσάρτηση κατάλληλου αριθμού bit στο τέλος του μηνύματος, ώστε το συνολικό μήκος να καταστεί ακριβές πολλαπλάσιο ενός σταθερού αριθμού bit, συμβολιζόμενου ως $r$ (rate το οποίο για τον SHA-3 μπορεί να είναι 576, 832, 1088 ή 1152, ανάλογα με το επιθυμητό μέγεθος εξόδου).

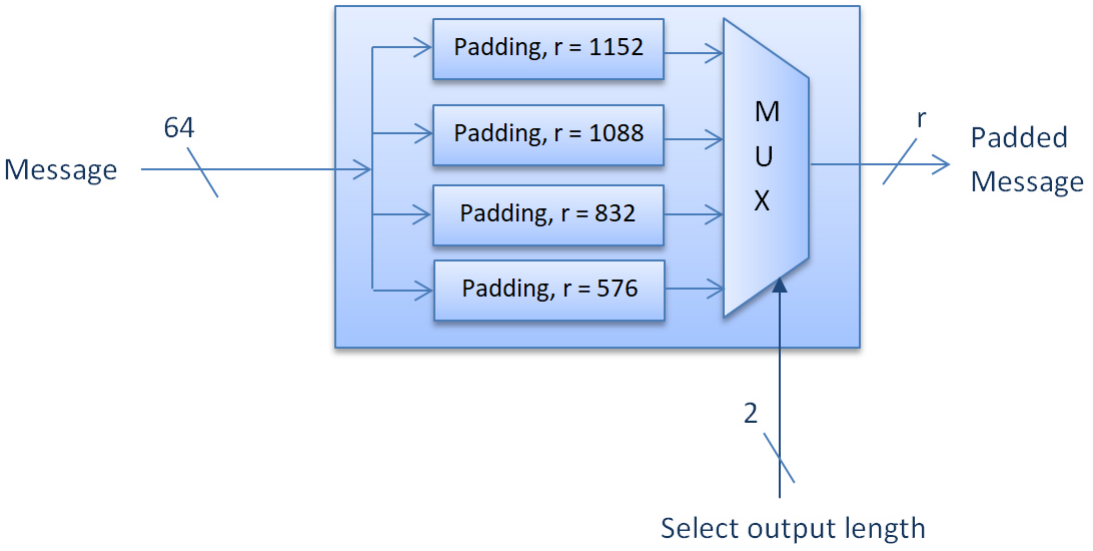
Συγκεκριμένα, η μονάδα πλήρωση που εφαρμόζεται στον SHA-3 περιλαμβάνει [154]:

1. την προσθήκη ενός bit με τιμή "1" στο τέλος του μηνύματος,
2. την προσθήκη τόσων bit "0" όσων απαιτούνται ώστε το συνολικό μήκος να γίνει $r - 1$,
3. και τέλος, την προσθήκη ενός ακόμη bit με τιμή "1".

Με αυτόν τον τρόπο, διασφαλίζεται ότι το τελικό μήκος του μηνύματος είναι ακριβές πολλαπλάσιο των $r$ bits.

Η μονάδα πλήρωση που χρησιμοποιείται στον αλγόριθμο SHA-3 επιλέγεται με βάση το επιθυμητό μήκος εξόδου και υλοποιείται μέσω ενός πολυπλέκτη 4 προς 1. Το μήκος εξόδου καθορίζει την αντίστοιχη τιμή του παραμέτρου $r$, η οποία με τη σειρά της προσδιορίζει το συγκεκριμένο σχήμα πλήρωσης που εφαρμόζεται. Για παράδειγμα, αν το μήκος εξόδου έχει οριστεί σε 224 bits, τότε χρησιμοποιείται σχήμα πλήρωσης για $r = 1152$ bits, όπως απεικονίζεται στο Σχήμα 3.2.

Αφού ολοκληρωθεί η πλήρωση, το μήνυμα περνά στη μονάδα χαρτογράφησης, όπου πραγματοποιείται πράξη XOR μεταξύ των πρώτων $r$ bits του συμπληρωμένου μηνύματος και της τρέχουσας κατάστασης. Αυτή η διαδικασία εξασφαλίζει ότι ο τελικός πίνακας κατάστασης που θα υποβληθεί στον αλγόριθμο κατακερματισμού διαφέρει σημαντικά από την αρχική μορφή του μηνύματος. Το αποτέλεσμα της




Σχήμα 3.2: Μονάδα πλήρωσης του SHA-3.

πράξης αυτής προσαρτάται με τα επόμενα $c$ bits (όπου $c$ είναι σταθερά ίση με $1600 - r$), ολοκληρώνοντας έτσι το σχήμα προετοιμασίας του μηνύματος για επεξεργασία από τη συνάρτηση κατακερματισμού.

Ο μετασχηματισμός δεδομένων που πραγματοποιείται κατά την εξαγωγή της τελικής τιμής κατακερματισμού περιλαμβάνει την περικοπή συγκεκριμένων ψηφίων (bits) από την τελική κατάσταση (state), ώστε να προκύπτει το επιθυμητό μήκος εξόδου. Η διαδικασία αυτή περιγράφεται στην Εξίσωση (3.1), όπου τα bits που επιλέγονται εξαρτώνται από το επιλεγμένο μήκος εξόδου ($r$ = 576, 832, 1088 ή 1152). Η υλοποίηση της περικοπής επιτυγχάνεται μέσω μιας μονάδας περικοπής, η οποία βασίζεται στη χρήση πολυπλέκτη 4 προς 1, επιτρέποντας την ευέλικτη επιλογή των κατάλληλων bits για κάθε εκδοχή του αλγορίθμου.

$$\text{State}[x, y, z] = ((\text{Padded data } r \oplus r)||c)^* \left[64^*(5^* y + x) + z\right] \tag{3.1}$$

Επιπλέον, το βήμα $\iota$ $(iota)$ περιλαμβάνει την τροποποίηση ορισμένων bit του πίνακα κατάστασης $A$ μέσω της προσθήκης (XOR) μιας στρογγυλοποιημένης σταθεράς, όπως περιγράφεται στην Εξίσωση (3.2).

$$A'[x, y, z] = A[x, y, z] \oplus RC\left[i_w\right] \tag{3.2}$$

Η τιμή της στρογγυλοποιημένης σταθεράς RC υπολογίζεται όπως φαίνεται στην Εξίσωση (3.3), σύμφωνα με τις προδιαγραφές του αλγορίθμου SHA-3 [30]. Σε όλες τις υπόλοιπες θέσεις της RC (δηλαδή στις τιμές $RC[i_{\mathrm{w}}][x][y][z]$ εκτός των συγκεκριμένων που ορίζονται στην εξίσωση), το αντίστοιχο bit έχει τιμή μηδέν. Όπως προκύπτει από την Εξίσωση (3.3), μόνο 7 από τα 64 bits της στρογγυλοποιημένης σταθεράς μπορούν να έχουν την τιμή 1, ενώ όλα τα υπόλοιπα διατηρούνται στο μηδέν.

$$\mathrm{RC}\left[i_{\mathrm{w}}\right][0][0]\left[2^{q}-1\right]=\mathrm{wc}\left[q+7i_{\mathrm{w}}\right] \text{ for all } 0\leq q\leq m \tag{3.3}$$

Σύμφωνα με τις προδιαγραφές του SHA-3, ο Πίνακας 3.3 παρουσιάζει λεπτομερώς τις ακριβείς θέσεις των 7 bit με τιμή "1" όταν η παράμετρος $m = 6$. Συγκεκριμένα, οι μοναδικές θέσεις bit στις οποίες εμφανίζεται η τιμή "1" είναι οι 0, 1, 3, 7, 15, 31 και 63, ενώ όλες οι υπόλοιπες θέσεις bit διατηρούν την τιμή "0".

Πίνακας 3.3: Οι θέσεις για καθένα από τα 7-bit όπου έχουν την τιμή 1.

| **q** | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| **[z]** | 0 | 1 | 3 | 7 | 15 | 31 | 63 |

Ο Πίνακας 3.4 παρουσιάζει ένα παράδειγμα της απλοποιημένης μορφής που χρησιμοποιήθηκε για τη στρογγυλοποιημένη σταθερά $RC_6$ του Πίνακα 3.5. Ως αποτέλεσμα, η πράξη XOR που εφαρμόζεται στον πίνακα κατάστασης $A$ επηρεάζει συνολικά επτά συγκεκριμένες θέσεις bit.

Πίνακας 3.4: Παράδειγμα της νέας μορφής του $RC_6$ στο βήμα $\iota$ $(iota)$.

| **Δεκαεξαδικό** | **Δυαδικό** | | | | **Θέσεις με τιμή 1** |
|---|---|---|---|---|---|
| 8081 | 1000 | 0000 | 1000 | 0001 | 0th = 1<br>1st = 0<br>3rd = 0<br>7th = 1<br>15th = 1 |
| 8000 | 1000 | 0000 | 0000 | 0000 | 31st = 1 |
| 0000 | 0000 | 0000 | 0000 | 0000 | - |
| 8000 | 1000 | 1000 | 1000 | 1000 | 63th = 1 |

Η αρχιτεκτονική με διασωλήνωση (pipelining) αποτελεί μια δημοφιλή σχεδιαστική προσέγγιση για την επίτευξη χαμηλής κατανάλωσης ενέργειας, υψηλής ασφάλειας και αυξημένης απόδοσης σε κρυπτογραφικές υλοποιήσεις [155]. Στο προτεινόμενο σύστημα, ο στόχος μας είναι η βελτιστοποίηση μιας αρχιτεκτονικής διασωλήνωσης

Πίνακας 3.5: Η νέα μορφή $RC_i$ του βήματος $\iota$ $(iota)$.

| | | | | | |
|---|---|---|---|---|---|
| $RC_0$ | 1000000 | $RC_8$ | 0111000 | $RC_{16}$ | 0100101 |
| $RC_1$ | 0101100 | $RC_9$ | 0011000 | $RC_{17}$ | 0001001 |
| $RC_2$ | 0111101 | $RC_{10}$ | 1010110 | $RC_{18}$ | 0110100 |
| $RC_3$ | 0000111 | $RC_{11}$ | 0110010 | $RC_{19}$ | 0110011 |
| $RC_4$ | 1111100 | $RC_{12}$ | 1111110 | $RC_{20}$ | 1001111 |
| $RC_5$ | 1000010 | $RC_{13}$ | 1111001 | $RC_{21}$ | 0001101 |
| $RC_6$ | 1001111 | $RC_{14}$ | 1011101 | $RC_{22}$ | 1000010 |
| $RC_7$ | 1010101 | $RC_{15}$ | 1100101 | $RC_{23}$ | 0010101 |

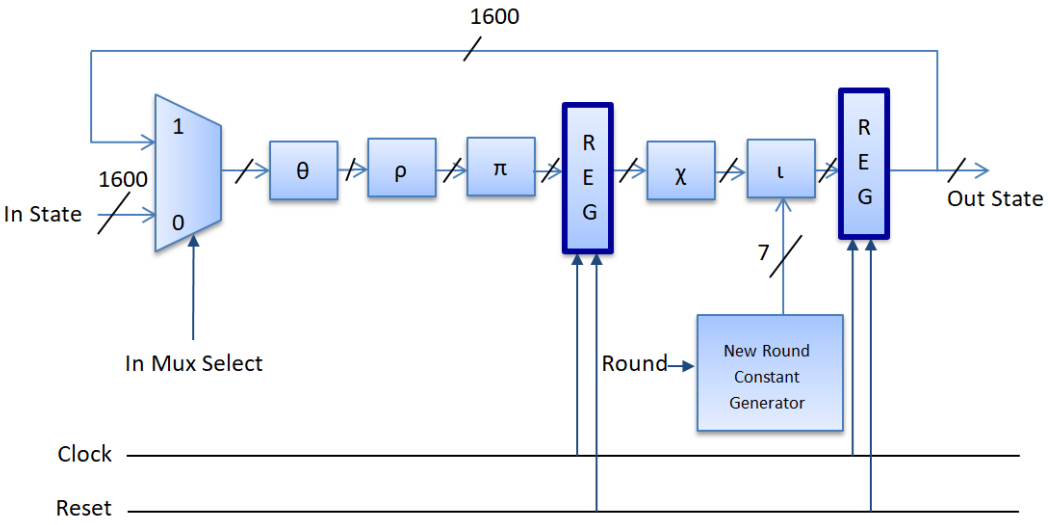
δύο σταδίων, με σκοπό την επίτευξη υψηλότερης συχνότητας λειτουργίας (MHz), αποδοτικότητας (Mbps/slices) και ρυθμαπόδοσης (Gbps) για όλα τα υποστηριζόμενα μήκη εξόδου.

Για την επίτευξη του στόχου αυτού, σχεδιάστηκαν και αξιολογήθηκαν δύο εναλλακτικές στρατηγικές τοποθέτησης της διασωλήνωσης στη ροή επεξεργασίας της συνάρτησης $f$. Η βελτίωση της συνολικής απόδοσης συνδέεται άμεσα με τη μείωση του μήκους του κρίσιμου μονοπατιού της $f$, το οποίο αποτελείται από 24 διαδοχικούς γύρους με πέντε επιμέρους λειτουργίες: $\theta, \rho, \pi, \chi,$ και $\iota$. Επομένως, η κατάλληλη τοποθέτηση των σταδίων διασωλήνωσης εντός της συνάρτησης $f$ είναι ουσιώδους σημασίας για τη μείωση του κρίσιμου μονοπατιού και, κατ' επέκταση, για την επίτευξη των επιθυμητών μετρικών απόδοσης.

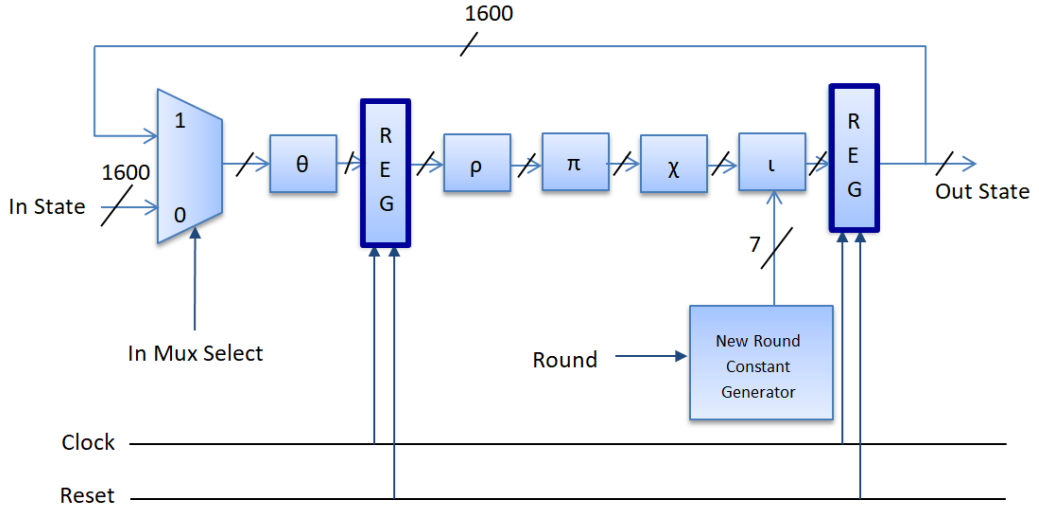
1. Το πρώτο προτεινόμενο αρχιτεκτονικό σχέδιο διασωλήνωσης για τον αλγόριθμο SHA-3 απεικονίζεται στο Σχήμα 3.3. Σε αυτήν την αρχιτεκτονική, ο πρώτος καταχωρητής αγωγού τοποθετείται μεταξύ των βημάτων $\pi$ και $\chi$, ενώ ο δεύτερος καταχωρητής βρίσκεται στο τέλος του κάθε γύρου.

2. Το δεύτερο προτεινόμενο αρχιτεκτονικό σχέδιο με διασωλήνωση του γύρου SHA-3 παρουσιάζεται στο Σχήμα 3.4. Εδώ, ο πρώτος καταχωρητής αγωγού τοποθετείται μεταξύ των βημάτων $\theta$ και $\rho$, ενώ ο δεύτερος καταχωρητής, όπως και πριν, βρίσκεται στο τέλος του γύρου.

Και στις δύο προτεινόμενες αρχιτεκτονικές διασωλήνωσης (Σχήματα 3.3 και 3.4), τα βασικά σήματα ελέγχου για τους δύο καταχωρητές είναι η επαναφορά (reset) και το ρολόι (clock). Ο μετρητής συνιστωσών παρέχει το σήμα ελέγχου για τη στρογγυλοποιημένη σταθερά.




Σχήμα 3.3: Πρώτη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση (σκούρο μπλε) όπου ο πρώτος αγωγός τοποθετείται μετά το βήμα $\pi$.




Σχήμα 3.4: Δεύτερη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση (σκούρο μπλε) όπου ο πρώτος αγωγός τοποθετείται μετά το βήμα $\theta$.

## 3.5 Πειραματικά αποτελέσματα

Για την εξασφάλιση της δίκαιης σύγκρισης της προτεινόμενης στρατηγικής με άλλες σχετικές μελέτες, χρησιμοποιήσαμε πλακέτες FPGA των σειρών Virtex-5, Virtex-6 και Virtex-7. Οι υλοποιήσεις στις πλακέτες Virtex-5 και Virtex-6 πραγματοποιήθηκαν με χρήση του εργαλείου Xilinx ISE, ενώ οι αρχιτεκτονικές στη σειρά Virtex-7 αναπτύχθηκαν με το Xilinx Vivado.

### 3.5.1 Επικύρωση της τροποποιημένης κατασκευής

Η τροποποιημένη κατασκευή βασίζεται στις προδιαγραφές του SHA-3 [30], με ιδιαίτερη έμφαση στην Εξίσωση (3.3), σύμφωνα με την οποία μόνο 7 από τα 64 bits της στρογγυλοποιημένης σταθεράς RC μπορούν να λάβουν τιμή 1. Ακολουθώντας τις προδιαγραφές του αλγορίθμου και αξιοποιώντας τις αναγνωρισμένες ιδιότητες ασφαλείας του, η τροποποιημένη κατασκευή διατηρεί τις εγγυήσεις ασφάλειας που παρέχονται από το πρότυπο SHA-3.

Η επικύρωση της υλοποίησης πραγματοποιείται μέσω προσομοιώσεων με παραδείγματα που παρέχονται από το Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας [156], το οποίο αποτελεί αξιόπιστη πηγή για κρυπτογραφικά πρότυπα. Αυτή η διαδικασία διασφαλίζει ότι η τροποποιημένη κατασκευή λειτουργεί ορθά και παράγει με συνέπεια τα αναμενόμενα αποτελέσματα. Έτσι, ο συνδυασμός της τήρησης των προδιαγραφών του SHA-3 και της επικύρωσης μέσω προσομοίωσης με επίσημα παραδείγματα συμβάλλει στην αξιοπιστία της προτεινόμενης υλοποίησης.

### 3.5.2 Μέτρα αποδοτικότητας και ρυθμαπόδοσης

Τα τυπικά μέτρα αξιολόγησης, όπως η αποδοτικότητα και η ρυθμαπόδοση, χρησιμοποιούνται για τη συγκριτική αξιολόγηση των υλοποιήσεων του αλγορίθμου SHA-3 σε πλατφόρμες FPGA [141, 157]. Ο όρος ρυθμαπόδοση (throughput) αναφέρεται στον αριθμό των bit που υποβάλλονται σε επεξεργασία ανά μονάδα χρόνου, και συνήθως εκφράζεται σε Gbps ή Mbps. Η ρυθμαπόδοση προσδιορίζεται από την Εξίσωση (3.4).

$$Throughput_{pipeline} = \frac{\textit{A message block's bits (r)}}{\textit{Cycles of the clock for each message block}} \times \textit{Frequency} \tag{3.4}$$

Στην Εξίσωση (3.4), το μέγεθος $r$ (A message block's bits) αντιστοιχεί στον αριθμό των bit κάθε μπλοκ μηνύματος και λαμβάνει τιμές 576, 832, 1088 ή 1152, ανάλογα με την επιλεγμένη παραλλαγή του αλγορίθμου. Η παράμετρος συχνότητα (Frequency) αντιπροσωπεύει τη μέγιστη συχνότητα ρολογιού λειτουργίας του συστήματος, ενώ οι κύκλοι ρολογιού ανά μπλοκ (Cycles of the clock for each message block) χαρακτηρίζουν τον αριθμό των επαναλήψεων που απαιτούνται για την επεξεργασία κάθε μπλοκ μηνύματος από τις πέντε βασικές διαδικασίες του αλγορίθμου: $\theta, \rho, \pi, \chi,$ και $\iota$, ώστε να παραχθεί η τελική τιμή κατακερματισμού.

Η αποδοτικότητα ($Efficiency_{pipeline}$) ενός συστήματος με διασωλήνωση ορίζεται ως ο λόγος της ρυθμαπόδοσης ($Throughput_{pipeline}$) προς την επιφάνεια υλοποίησης ($Area_{pipeline}$), όπως αυτή αποτυπώνεται στην Εξίσωση (3.5). Συγκεκριμένα:

$$Efficiency_{pipeline} = \frac{Throughput_{pipeline}}{Area_{pipeline}} \tag{3.5}$$

Ο δείκτης αποδοτικότητας εκφράζει επομένως το πόσα Mbps (ή Gbps) μπορεί να επεξεργαστεί το σύστημα ανά μονάδα επιφάνειας υλικού, προσφέροντας μια συγκριτική μέτρηση για το πόσο αποδοτικά αξιοποιούνται οι διαθέσιμοι υλικοί πόροι για την επίτευξη υψηλής απόδοσης.

### 3.5.3 Αποτελέσματα των δύο αρχιτεκτονικών

Προκειμένου να βελτιωθεί η απόδοση του αλγορίθμου SHA-3, είναι σημαντικό να εντοπιστούν τα πιο δαπανηρά υπολογιστικά βήματα και να εστιάσουμε στη βελτιστοποίηση αυτών των σταδίων. Στην παραδοσιακή αρχιτεκτονική, ο υπολογισμός των bit ισοτιμίας στις στήλες του πίνακα καταστάσεων απαιτεί πρόσβαση σε ολόκληρο τον πίνακα, με αποτέλεσμα σημαντική κίνηση δεδομένων και αυξημένο υπολογιστικό κόστος. Αυτό επιβαρύνει τη χρήση των πόρων και μειώνει τη συνολική ρυθμαπόδοση και αποδοτικότητα του αλγορίθμου.

Για την αντιμετώπιση αυτής της πρόκλησης, εισάγεται καταχωρητής αμέσως μετά το βήμα $\theta$, καθώς αυτό το στάδιο είναι το πιο δαπανηρό υπολογιστικά στη

συνάρτηση μετάθεσης, καταναλώνοντας πάνω από το 50% του συνολικού χρόνου εκτέλεσης. Ο καταχωρητής λειτουργεί ως προσωρινό στοιχείο αποθήκευσης που διατηρεί τα υπολογισμένα bit ισοτιμίας, εξαλείφοντας την ανάγκη επανειλημμένης πρόσβασης στον πίνακα καταστάσεων. Με τον τρόπο αυτό, τα επόμενα στάδια του αλγορίθμου αποκτούν άμεση πρόσβαση στα απαραίτητα δεδομένα χωρίς εκτεταμένη μετακίνηση ή επανυπολογισμό. Η εισαγωγή του καταχωρητή μειώνει σημαντικά το υπολογιστικό φορτίο και τις απαιτήσεις πόρων, βελτιώνοντας τη ρυθμαπόδοση και την αποδοτικότητα. Παράλληλα, απλοποιεί τη ροή δεδομένων εντός του αλγορίθμου, επιτρέποντας ταχύτερη και αποτελεσματικότερη επεξεργασία, ενώ ελαχιστοποιεί το συνολικό κόστος της εφαρμογής SHA-3 μέσω βελτιστοποίησης της χρήσης των πόρων.

Εναλλακτικά, η εισαγωγή καταχωρητή μετά το βήμα $\pi$ μπορεί επίσης να βελτιώσει την απόδοση του αλγορίθμου, αλλά σε μικρότερο βαθμό. Το βήμα $\pi$ είναι υπεύθυνο κυρίως για την αναδιάταξη των bit στον πίνακα καταστάσεων και είναι λιγότερο υπολογιστικά εντατικό από το $\theta$. Συνεπώς, η εισαγωγή καταχωρητή μετά το $\theta$ οδηγεί σε πιο ουσιαστική βελτίωση της υπολογιστικής απόδοσης του SHA-3 σε σύγκριση με την εισαγωγή καταχωρητή μετά το $\pi$.

Ο Πίνακας 3.6 παρουσιάζει τα αποτελέσματα των δύο αρχιτεκτονικών διασωλήνωσης, με καταχωρητή μετά το $\theta$ και μετά το $\pi$, σε πλακέτες FPGA Virtex-5, Virtex-6 και Virtex-7. Στην πλατφόρμα Virtex-5, η πρώτη αρχιτεκτονική απαιτεί 1102 slices και λειτουργεί στα 374 MHz, ενώ η δεύτερη 998 slices στα 402 MHz. Στην πλακέτα Virtex-6, η πρώτη αρχιτεκτονική χρησιμοποιεί 1146 slices με συχνότητα 392 MHz, ενώ η δεύτερη 1042 slices με 422 MHz. Τέλος, στην πλακέτα Virtex-7, η αρχιτεκτονική με καταχωρητή μετά το $\theta$ απαιτεί 1288 slices στα 446 MHz, ενώ εκείνη με καταχωρητή μετά το $\pi$ 1150 slices στα 478 MHz. Τα αποτελέσματα αυτά επιβεβαιώνουν ότι η διασωλήνωση με καταχωρητή μετά το βήμα $\theta$ προσφέρει τη μεγαλύτερη βελτίωση στην απόδοση του αλγορίθμου SHA-3, διατηρώντας παράλληλα υψηλή αποδοτικότητα στη χρήση των πόρων.

Η κατανάλωση ενέργειας των προτεινόμενων σχεδίων μας αξιολογείται χρησιμοποιώντας το εργαλείο ανάλυσης Xilinx XPower [158]. Ο Πίνακας 3.7 εμφανίζει τα αποτελέσματα κατανάλωσης ενέργειας των δύο τεχνικών βελτιστοποίησης με διασωλήνωση με πλακέτες Virtex-5, Virtex-6 και Virtex-7 FPGA. Στην πρώτη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση, η κατανάλωση ενέργειας στα FPGA Virtex-5, Virtex-6 και Virtex-7 ήταν 267 mW, 222 mW και 179 mW, αντίστοιχα. Στη δεύτερη προτεινόμενη τεχνική βελτιστοποίησης

Πίνακας 3.6: Μετρήσεις απόδοσης των δύο μεθόδων βελτιστοποίησης διοχετεύσεων υλικού για τον SHA-3 όταν εφαρμόζονται στα Virtex-5, Virtex-6 και Virtex-7 FPGA.

| **Σχεδίαση** | **Μήκος** | **Πρώτη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση όπου ο πρώτος αγωγός τοποθετείται μετά το βήμα $\pi$** | | | **Δευτερη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση όπου ο πρώτος αγωγός τοποθετείται μετά το βήμα $\theta$** | | |
|---|---|---|---|---|---|---|---|
| **FPGA** | | Virtex-5 | Virtex-6 | Virtex-7 | Virtex-5 | Virtex-6 | Virtex-7 |
| **Επιφάνεια σε (slices)** | | 1102 | 1146 | 1288 | 998 | 1042 | 1150 |
| **Συχνότητα (MHz)** | | 374 | 392 | 446 | 402 | 422 | 478 |
| **Ρυθμαπόδοση (Gbps)** | r = 1152 | 17,952 | 18,816 | 21,408 | 19,296 | 20,256 | 22,944 |
| | r = 1088 | 16,955 | 17,771 | 20,219 | 18,224 | 19,131 | 21,669 |
| | r = 832 | 12,965 | 13,589 | 15,461 | 13,936 | 14,629 | 16,571 |
| | r = 576 | 8,976 | 9,408 | 10,704 | 9,648 | 10,128 | 11,472 |
| **Αποδοτικότητα (Mbps/slices)** | r = 1152 | 16,29 | 16,42 | 16,62 | 19,33 | 19,44 | 19,95 |
| | r = 1088 | 15,39 | 15,51 | 15,70 | 18,26 | 18,36 | 18,84 |
| | r = 832 | 11,77 | 11,86 | 12,00 | 13,96 | 14,04 | 14,41 |
| | r = 576 | 8,15 | 8,21 | 8,31 | 9,67 | 9,72 | 9,98 |

με διασωλήνωση, η κατανάλωση ενέργειας στα Virtex-5, Virtex-6 και Virtex-7 FPGA ήταν 242 mW, 198 mW και 157 mW, αντίστοιχα.

Σε όλα τα μοντέλα FPGA, η δεύτερη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση μετά το βήμα $\theta$ παρουσιάζει χαμηλότερη κατανάλωση ενέργειας από την πρώτη προτεινόμενη τεχνική μετά το βήμα $\pi$. Μεταξύ των μοντέλων Virtex FPGA, το Virtex-7 επιδεικνύει σταθερά τη χαμηλότερη κατανάλωση ενέργειας και για τις δύο τεχνικές βελτιστοποίησης. Το Virtex-6 γενικά παρουσιάζει χαμηλότερη κατανάλωση ενέργειας από το Virtex-5 και στις δύο περιπτώσεις. Έτσι, η δεύτερη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση, με την πρώτη διασωλήνωση τοποθετημένη μετά το βήμα $\theta$, είναι πιο αποδοτική σε σχέση με τα μοντέλα FPGA που αξιολογήθηκαν.

Πίνακας 3.7: Η κατανάλωση ενέργειας των δύο τεχνικών βελτιστοποίησης με διασωλήνωση για το SHA-3 όταν εφαρμόζεται στα Virtex-5, Virtex-6 και Virtex-7 FPGA.

| **Σχεδίαση** | **FPGA** | **Ενέργεια (mW)** |
|---|---|---|
| **Πρώτη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση όπου ο πρώτος αγωγός τοποθετείται μετά το βήμα $\pi$** | Virtex-5 | 267 |
| | Virtex-6 | 222 |
| | Virtex-7 | 179 |
| **Δεύτερη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση όπου ο πρώτος αγωγός τοποθετείται μετά το βήμα $\theta$** | Virtex-5 | 242 |
| | Virtex-6 | 198 |
| | Virtex-7 | 157 |

## 3.6 Συζήτηση

Ο κύριος στόχος της παρούσας εργασίας είναι η επίτευξη υψηλότερης επιτάχυνσης, ρυθμαπόδοσης (Gbps) και αποδοτικότητας (Mbps/slices) στο σχεδιαζόμενο σύστημα. Η πειραματική διαδικασία κατέδειξε ότι τα αποτελέσματα επηρεάζονται άμεσα από το κρίσιμο μονοπάτι της συνάρτησης $f$. Διαπιστώθηκε ότι η επιτάχυνση και η συνολική βελτίωση στη ρυθμαπόδοση και την αποδοτικότητα είναι σημαντικά μεγαλύτερες όταν ο καταχωρητής εισάγεται μετά το βήμα $\theta$ σε σύγκριση με την εισαγωγή του μετά το βήμα $\pi$.

Οι Πίνακες 3.8 και 3.9 παρουσιάζουν συγκριτικά αποτελέσματα με άλλες σχετικές αρχιτεκτονικές για όλα τα υποστηριζόμενα μήκη εξόδου (224, 256, 384 και 512 bit), ως προς τις μετρικές της ρυθμαπόδοσης (Gbps), της συχνότητας λειτουργίας (MHz) και της αποδοτικότητας (Mbps/slices) για τον αλγόριθμο SHA-3. Αξίζει να σημειωθεί ότι η πλειονότητα των σχετικών μελετών εστιάζει πειραματικά κυρίως σε μήκη εξόδου 256 ή 512 bit. Όλα τα συγκρινόμενα αποτελέσματα αναφέρονται σε επεξεργασία μηνυμάτων ενός μόνο μπλοκ..

Οι ερευνητές στις εργασίες [146–149, 151, 153], αξιοποιώντας την πλακέτα Virtex-5 FPGA, αναφέρουν υψηλές απαιτήσεις σε επιφάνεια (slices) και χαμηλότερες συχνότητες λειτουργίας σε σύγκριση με τη δική μας υλοποίηση. Επιπλέον, στις μελέτες [149, 150] με χρήση Virtex-6 FPGA, παρατηρείται χαμηλότερη συχνότητα σε σχέση με τα αποτελέσματά μας, αν και καταγράφεται σημαντική βελτίωση στην επιφάνεια σε slices. Στην εργασία [150] με πλακέτα Virtex-7 FPGA, καταγράφεται μεγαλύτερη απαίτηση σε επιφάνεια (slices) και συχνότητα λειτουργίας σε σύγκριση με ό,τι επιτύχαμε με τις δικές μας τεχνικές βελτιστοποίησης. Τέλος, στην εργασία [152] με χρήση Stratix IV FPGA, παρουσιάζονται ακόμη μεγαλύτερες απαιτήσεις σε επιφάνεια (slices) και σημαντικά χαμηλότερη συχνότητα από αυτές που καταφέραμε μέσω των δικών μας τεχνικών βελτιστοποίησης.

Με τη μέθοδό μας για μήκος εξόδου 256 bit και υλοποίηση σε Virtex-7 FPGA, η προτεινόμενη αρχιτεκτονική επιτυγχάνει βελτίωση της ρυθμαπόδοσης κατά περισσότερο από 10%, της αποδοτικότητας κατά άνω του 14%, της συχνότητας λειτουργίας κατά πάνω από 11%, καθώς και μείωση της απαιτούμενης επιφάνειας σε slices κατά περισσότερο από 14%, σε σύγκριση με την πλησιέστερη υλοποίηση της βιβλιογραφίας [150]. Αντίστοιχα, για μήκος εξόδου 512 bit με χρήση Virtex-6 FPGA, η αρχιτεκτονική μας παρουσιάζει βελτίωση της ρυθμαπόδοσης κατά πάνω από 10%, της αποδοτικότητας κατά άνω του 11%, της συχνότητας κατά περισσότερο

από 10%, καθώς και μείωση της επιφάνειας σε slices κατά πάνω από 22% σε σύγκριση με την αντίστοιχη καλύτερη υλοποίηση του [149].

Πίνακας 3.8: Αποτελέσματα και συγκρίσεις ρυθμαπόδοσης για καθένα από τα μήκη εξόδου (224, 256, 384 και 512 bit) για τον αλγόριθμο SHA-3.

| **Σχεδίαση** | **FPGA** | **Επιφάνεια σε (Slices)** | **Συχνότητα (MHz)** | **Ρυθμαπόδοση (Gbps) r= 1152** | **Ρυθμαπόδοση (Gbps) r = 1088** | **Ρυθμαπόδοση (Gbps) r = 832** | **Ρυθμαπόδοση (Gbps) r = 576** |
|---|---|---|---|---|---|---|---|
| [146] | Virtex-5 | 2326 | 306 | - | - | - | 5,56 |
| [147] | Virtex-5 | 4793 | 317,11 | - | 12,68 | - | - |
| [148] | Virtex-5 | 1163 | 273 | - | - | - | 7,80 |
| [149] | Virtex-5 | 2652 | 352 | - | - | - | 8,44 |
| | ΐρτεξ-6 | 2296 | 391 | - | - | - | 9,38 |
| [150] | Virtex-5 | 1702 | 389 | - | 18,07 | - | - |
| | Virtex-6 | 1649 | 397 | - | 19,01 | - | - |
| | Virtex-7 | 1618 | 434 | - | 20,80 | - | - |
| [151] | Virtex-5 | 2123 | - | - | 12,523 | - | 7,380 |
| | Virtex-6 | 1456 | - | - | 14,942 | - | 8,114 |
| [152] | Stratix IV | 5363 | 110 | - | - | - | - |
| [153] | Virtex-5 | 1680 | 387 | - | - | - | 8,06 |
| Δεύτερη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση όπου η πρώτη διασωλήνωση τοποθετείται μετά το βήμα $\theta$ | Virtex-5 | 998 | 402 | 19,29 | 18,22 | 13,93 | 9,64 |
| Δεύτερη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση όπου η πρώτη διασωλήνωση τοποθετείται μετά το βήμα $\theta$ | Virtex-6 | 1042 | 422 | 20,25 | 19,13 | 14,62 | 10,12 |
| Δεύτερη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση όπου η πρώτη διασωλήνωση τοποθετείται μετά το βήμα $\theta$ | Virtex-7 | 1150 | 478 | 22,94 | 21,66 | 16,57 | 11,47 |

Πίνακας 3.9: Αποτελέσματα και συγκρίσεις της αποδοτικότητας για κάθε μήκος εξόδου (224, 256, 384 και 512 bit) για τον αλγόριθμο SHA-3.

| **Σχεδίαση** | **FPGA** | **Επιφάνεια σε (Slices)** | **Συχνότητα (MHz)** | **Αποδοτικότητα (Mbps/slices) r = 1152** | **Αποδοτικότητα (Mbps/slices) r = 1088** | **Αποδοτικότητα (Mbps/slices) r = 832** | **Αποδοτικότητα (Mbps/slices) r = 576** |
|---|---|---|---|---|---|---|---|
| [146] | Virtex-5 | 2326 | 306 | - | - | - | 2,40 |
| [147] | Virtex-5 | 4793 | 317,11 | - | 2,71 | - | - |
| [148] | Virtex-5 | 1163 | 273 | - | - | - | 6,06 |
| [149] | Virtex-5 | 2652 | 352 | - | - | - | 6,37 |
| | Virtex-6 | 2296 | 391 | - | - | - | 8,17 |
| [150] | Virtex-5 | 1702 | 389 | - | 10,98 | - | - |
| | Virtex-6 | 1649 | 397 | - | 11,60 | - | - |
| | Virtex-7 | 1618 | 434 | - | 12,90 | - | - |
| [151] | Virtex-5 | 2123 | - | - | 5,90 | - | 4,16 |
| | Virtex-6 | 1456 | - | - | 10,26 | - | 6,42 |
| [152] | Stratix IV | 5363 | 110 | - | - | - | - |
| [153] | Virtex-5 | 1680 | 387 | - | - | - | 4,91 |
| Δεύτερη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση όπου η πρώτη διασωλήνωση τοποθετείται μετά το βήμα $\theta$ | Virtex-5 | 998 | 402 | 19,33 | 18,26 | 13,96 | 9,67 |
| Δεύτερη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση όπου η πρώτη διασωλήνωση τοποθετείται μετά το βήμα $\theta$ | Virtex-6 | 1042 | 422 | 19,44 | 18,36 | 14,04 | 9,72 |
| Δεύτερη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση όπου η πρώτη διασωλήνωση τοποθετείται μετά το βήμα $\theta$ | Virtex-7 | 1150 | 478 | 19,95 | 18,84 | 14,41 | 9,98 |

## 3.7 Συμπεράσματα κεφαλαίου και μελλοντικές εργασίες

Στη σύγχρονη ψηφιακή εποχή, η διακίνηση της πληροφορίας πραγματοποιείται μέσω ποικίλων μορφών, όπως εικόνα, κείμενο, βίντεο και ήχος. Η ασφαλής μετάδοση και αποθήκευση των δεδομένων αυτών, με διασφάλιση της εμπιστευτικότητας, της ακεραιότητας και της διαθεσιμότητάς τους, συνιστά θεμελιώδη απαίτηση για την αποτροπή μη εξουσιοδοτημένης πρόσβασης και την προστασία της ιδιωτικότητας των χρηστών. Για τον σκοπό αυτό, οι κρυπτογραφικοί αλγόριθμοι έχουν καταστεί κομβικό στοιχείο των συστημάτων ψηφιακής ασφάλειας, παρέχοντας ισχυρούς μηχανισμούς προστασίας έναντι απειλών και επιθέσεων που στοχεύουν την παραβίαση ή παραποίηση των δεδομένων.

Οι επιτυχείς κρυπτανάλυσεις και οι τεκμηριωμένες αδυναμίες των αλγορίθμων SHA-1 και SHA-2 οδήγησαν το Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας στην υιοθέτηση ενός νέου, προηγμένου προτύπου κατακερματισμού: του αλγορίθμου SHA-3. Ο SHA-3, αξιοποιώντας καινοτόμες αρχιτεκτονικές προσεγγίσεις, προσφέρει υψηλό επίπεδο ασφάλειας και εμφανίζει αυξημένη ανθεκτικότητα σε σύγχρονες μορφές επιθέσεων κρυπτανάλυσης, όπως ρολλισιον και πρειμαγε αττασκς. Επιπλέον, ο SHA-3 διακρίνεται για την ευελιξία του στη σχεδίαση υλικού, επιτρέποντας την αποτελεσματική επιτάχυνση και βελτιστοποίηση ως προς την απόδοση και την ενεργειακή αποδοτικότητα. Τα χαρακτηριστικά αυτά καθιστούν τον SHA-3 ιδανική επιλογή για προηγμένες και απαιτητικές κρυπτογραφικές εφαρμογές στη σύγχρονη ψηφιακή πραγματικότητα.

Στο παρόν κεφάλαιο, η έρευνα εστιάζει στη βελτιστοποίηση των μετρικών ρυθμαπόδοσης και αποδοτικότητας του αλγορίθμου SHA-3 για όλα τα υποστηριζόμενα μήκη εξόδου (224, 256, 384 και 512 bit), αξιοποιώντας διαφορετικές πλατφόρμες υλοποίησης, και συγκεκριμένα τις Virtex-5, Virtex-6 και Virtex-7 FPGA. Η προτεινόμενη μέθοδος συγκρίνεται με παρόμοιες αρχιτεκτονικές της βιβλιογραφίας και αποδεικνύεται ότι επιτυγχάνει υψηλότερη απόδοση ως προς τα τυπικά κριτήρια αξιολόγησης: ρυθμαπόδοση (Gbps) και αποδοτικότητα (Mbps/slices).

Ειδικότερα, με την υλοποίηση στην πλατφόρμα Virtex-7, επιτεύχθηκαν εξαιρετικά αποτελέσματα, με τη ρυθμαπόδοση να ανέρχεται στα 22,94 Gbps και την αποδοτικότητα στα 19,95 Mbps/slices, τιμές που υπερβαίνουν τα αντίστοιχα

επιτεύγματα άλλων σύγχρονων προσεγγίσεων. Αξίζει να σημειωθεί ότι η προτεινόμενη αρχιτεκτονική διατηρεί υψηλή αποδοτικότητα ακόμη και στην επεξεργασία μηνυμάτων ενός μόνο μπλοκ, γεγονός που υποδηλώνει τη σταθερότητα και την ευελιξία της σχεδίασης σε διαφορετικά σενάρια εφαρμογής.

Σε μελλοντική εργασία, θα αναλύσουμε την αρχιτεκτονική τεχνική με βαθύτερη διασωλήνωση, με στόχο τη μείωση του κρίσιμου μονοπατιού και τη βελτίωση των μετρήσεων ρυθμαπόδοσης και αποδοτικότητας ανά γύρο SHA-3. Επιπλέον, προβλέπεται η εκτέλεση πιο πρακτικών πειραμάτων, με εφαρμογές σε ολοκληρωμένα συστήματα FPGA και ενσωματωμένα συστήματα σε επίπεδο ολοκληρωμένου κυκλώματος (System on Chip).

# Κεφάλαιο 4

# Τεχνική επιτάχυνσης ξετυλίγματος υλικού

Στο προηγούμενο κεφάλαιο παρουσιάστηκε μια τεχνική επιτάχυνσης και βελτιστοποίησης που βασίζεται στην τεχνική της διασωλήνωσης (pipelining), όπου αναλύθηκε η μέθοδος τοποθέτησης ενός πρόσθετου καταχωρητή μετά το βήμα $\theta$ $(theta)$ στη συνάρτηση $f$. Στο παρόν κεφάλαιο[1], προτείνεται μια τεχνική επιτάχυνσης του αλγορίθμου SHA-3 που βασίζεται στη μέθοδο του ξετυλίγματος (unrolling). Η ανάλυση της συγκεκριμένης μεθόδου εστιάζει στη σημαντική επιτάχυνση της διαδικασίας, μέσω της μείωσης του συνολικού αριθμού των κύκλων ρολογιού που απαιτούνται για την παραγωγή του αποτελέσματος της συνάρτησης κατακερματισμού.

## 4.1 Περίληψη

Στον τομέα της ψηφιακής επικοινωνίας και της ασφάλειας δεδομένων, οι κρυπτογραφικοί αλγόριθμοι κατακερματισμού αποτελούν θεμελιώδεις μηχανισμούς για τη διασφάλιση της ηλεκτρονικής ακεραιότητας και της εμπιστευτικότητας της πληροφορίας. Ενσωματώνονται σε κρίσιμους κλάδους, όπως η εθνική άμυνα, η υγειονομική περίθαλψη και οι χρηματοοικονομικές υπηρεσίες, παρέχοντας ισχυρή προστασία κατά τη μεταφορά και αποθήκευση δεδομένων. Μέσω της αξιόπιστης επαλήθευσης της ακεραιότητας των πληροφοριών, οι αλγόριθμοι αυτοί θωρακίζουν

[1]Το κεφάλαιο έχει δημοσιευθεί στο άρθρο [159]

τα συστήματα απέναντι σε ποικίλες απειλές, όπως η παραβίαση δεδομένων, η παραποίηση και η πλαστογράφηση, καθιστώντας τους απαραίτητους για την ορθή λειτουργία των σύγχρονων ψηφιακών υποδομών.

Σε σύγκριση με προηγούμενες γενιές κρυπτογραφικών αλγορίθμων κατακερματισμού, όπως οι SHA-1 και SHA-2, ο αλγόριθμος SHA-3 (Keccak) παρουσιάζει σημαντικά πλεονεκτήματα σε επίπεδο υλοποίησης υλικού, προσφέροντας αυξημένη απόδοση και υψηλότερη ανθεκτικότητα απέναντι στις σύγχρονες μεθόδους κρυπτανάλυσης. Ωστόσο, δεδομένης της διαρκώς αυξανόμενης ζήτησης για ασφαλέστερα και ταχύτερα συστήματα, παραμένει επιτακτική η ανάγκη για περαιτέρω βελτιστοποιήσεις στην απόδοση υλικού, με στόχο την αύξηση της ταχύτητας επεξεργασίας και/ή τη μείωση των απαιτήσεων σε υλικούς πόρους, ώστε να ανταποκρίνονται στις απαιτήσεις των σύγχρονων και μελλοντικών εφαρμογών ασφάλειας.

Η παρούσα έρευνα επικεντρώνεται στην επιτάχυνση της ρυθμαπόδοσης του αλγορίθμου κατακερματισμού SHA-3 μέσω της σχεδίασης και υλοποίησης μιας αρχιτεκτονικής, η οποία μειώνει σημαντικά τον συνολικό αριθμό των κύκλων ρολογιού που απαιτούνται για την ολοκλήρωση της συνάρτησης κατακερματισμού. Η προτεινόμενη αρχιτεκτονική επιτυγχάνει επιτάχυνση σε υλοποιήσεις με συσκευές FPGA, καταγράφοντας ταχύτητες μετάδοσης που υπερβαίνουν τα 36 Gbps. Μέσα από συγκριτική ανάλυση με άλλα σύγχρονα αρχιτεκτονικά σχέδια, η προσέγγιση αυτή αναδεικνύει τη δυναμική της για τη διαμόρφωση νέων προτύπων απόδοσης στον τομέα της κρυπτογραφικής ασφάλειας, υπογραμμίζοντας τη σημασία της συνεχούς καινοτομίας και της επιτάχυνσης στις τεχνολογίες συναρτήσεων κατακερματισμού.

## 4.2 Εισαγωγή

Στη σύγχρονη ψηφιακή εποχή, ένα κρυπτογραφικό πρωτόκολλο ορίζεται ως ένα σύνολο τυποποιημένων κανόνων και διαδικασιών που διασφαλίζουν την ασφάλεια και την αξιοπιστία της επικοινωνίας μεταξύ δύο ή περισσότερων οντοτήτων. Η εφαρμογή αποτελεσματικών κρυπτογραφικών πρωτοκόλλων καθίσταται θεμελιώδης για τη διαφύλαξη της εμπιστευτικότητας, της ακεραιότητας και της διαθεσιμότητας των δεδομένων που μεταφέρονται μέσω δικτύων ή αποθηκεύονται σε βάσεις δεδομένων, προστατεύοντάς τα από μη εξουσιοδοτημένη

πρόσβαση, αλλοίωση ή υποκλοπή. Η έλλειψη επαρκούς ασφάλειας στην ψηφιακή επικοινωνία ενέχει σοβαρούς κινδύνους, καθώς η παραβίαση ευαίσθητων πληροφοριών μπορεί να οδηγήσει σε οικονομικές απώλειες, νομικές κυρώσεις ή ανεπανόρθωτη βλάβη της φήμης οργανισμών και επιχειρήσεων. Μεταξύ των βασικότερων τεχνολογιών που διασφαλίζουν τα υψηλά πρότυπα ασφάλειας στη σύγχρονη ψηφιακή επικοινωνία συγκαταλέγονται η κρυπτογράφηση, οι ψηφιακές υπογραφές και οι κρυπτογραφικές συναρτήσεις κατακερματισμού [160, 161].

Στον τομέα της κρυπτογραφίας, η συνάρτηση κατακερματισμού αποτελεί μια θεμελιώδη μαθηματική διεργασία, η οποία λαμβάνει δεδομένα αυθαίρετου μεγέθους ως είσοδο και παράγει μία έξοδο σταθερού μήκους, γνωστή ως τιμή κατακερματισμού. Οι κρυπτογραφικές συναρτήσεις κατακερματισμού σχεδιάζονται με τέτοιο τρόπο ώστε να καθιστούν πρακτικά αδύνατη την ανάκτηση των αρχικών δεδομένων από τη δεδομένη τιμή κατακερματισμού, διασφαλίζοντας έτσι την μονόδρομη φύση τους. Οι συναρτήσεις αυτές διαδραματίζουν καίριο ρόλο στην επαλήθευση της ακεραιότητας των δεδομένων, την αυθεντικοποίηση, την υλοποίηση ψηφιακών υπογραφών, καθώς και στην ασφαλή αποθήκευση κωδικών πρόσβασης. Η ικανότητά τους να ανιχνεύουν ακόμη και την παραμικρή μεταβολή στα δεδομένα εισόδου - που οδηγεί στη δημιουργία εντελώς διαφορετικής τιμής κατακερματισμού - αποτελεί ένα από τα σημαντικότερα πλεονεκτήματα των κρυπτογραφικών αλγορίθμων κατακερματισμού [162].

Η σύγκριση των τιμών κατακερματισμού των αρχικών και των τροποποιημένων δεδομένων επιτρέπει την έγκαιρη ανίχνευση μη εξουσιοδοτημένων αλλαγών, καθιστώντας τις συναρτήσεις κατακερματισμού ζωτικής σημασίας για τη διασφάλιση της ακεραιότητας των δεδομένων σε πληθώρα εφαρμογών. Τέτοιες εφαρμογές περιλαμβάνουν τις ηλεκτρονικές τραπεζικές συναλλαγές, το ηλεκτρονικό εμπόριο και κρίσιμες κυβερνητικές λειτουργίες, όπου η ακεραιότητα και η ασφάλεια των δεδομένων είναι απολύτως απαραίτητες [163, 164].

Η αρχιτεκτονική Keccak [165], η οποία υιοθετήθηκε ως πρότυπο Secure Hash Algorithm-3 (SHA-3), έχει καθιερωθεί ως μια από τις πλέον διαδεδομένες τεχνικές κρυπτογραφικού κατακερματισμού, αντικαθιστώντας σταδιακά τους προκατόχους της, SHA-1 [25] και SHA-2 [166–168]. Ο SHA-3 προσφέρει σημαντικά πλεονεκτήματα έναντι των παλαιότερων αλγορίθμων, όπως η ευελιξία, η προσαρμοστικότητα και η δυνατότητα επαναχρησιμοποίησης των δομικών του στοιχείων σε ποικίλες εφαρμογές.

Από την επίσημη υιοθέτησή του το 2012, η επιστημονική κοινότητα έχει εστιάσει στη μελέτη και βελτιστοποίηση των παραμέτρων του SHA-3 για συγκεκριμένα σενάρια εφαρμογής, με ιδιαίτερη έμφαση στην υλοποίηση σε συσκευές υλικού. Σε επίπεδο υλικού, ο SHA-3 παρουσιάζει ανώτερη απόδοση συγκριτικά με τις υλοποιήσεις λογισμικού, κυρίως λόγω της αυξημένης ισχύος επεξεργασίας, της ταχύτητας και της ρυθμαπόδοσης που επιτυγχάνονται με εξειδικευμένη σχεδίαση [136, 169, 170].

Τα FPGAs έχουν επικρατήσει ως η πλέον ενδεδειγμένη πλατφόρμα υλοποίησης για τον SHA-3, υπερέχοντας έναντι των ολοκληρωμένων κυκλωμάτων ειδικών εφαρμογών (ASIC) ως προς το χαμηλότερο κόστος και τον συντομότερο χρόνο ανάπτυξης, διατηρώντας παράλληλα υψηλά επίπεδα απόδοσης και ενεργειακής αποδοτικότητας [137, 171].

Ένα από τα κύρια πλεονεκτήματα της υλοποίησης του αλγορίθμου SHA-3 σε συσκευές FPGA είναι η σημαντικά αυξημένη ταχύτητα εκτέλεσης συγκριτικά με παλαιότερους αλγορίθμους της οικογένειας SHA σε υλοποιήσεις υλικού, καθώς και η ικανότητα βέλτιστης απόδοσης σε ένα ευρύ φάσμα πλατφορμών υλικού. Η αξιοποίηση των FPGA για την υλοποίηση του SHA-3 παρέχει υψηλό βαθμό ευελιξίας, καθώς επιτρέπει τη δυναμική διαμόρφωση και επαναδιαμόρφωση των αλγορίθμων, προσαρμόζοντάς τους σε μεταβαλλόμενες απαιτήσεις εφαρμογής [172].

Επιπλέον, τα FPGAs προσφέρουν τη δυνατότητα σχεδιασμού με έμφαση στη χαμηλή ενεργειακή κατανάλωση [173], γεγονός που τα καθιστά εξαιρετικά κατάλληλα για υλοποιήσεις κρυπτογραφικών λειτουργιών, όπως ο SHA-3, σε περιβάλλοντα με περιορισμένους ενεργειακούς πόρους. Επίσης, η παράλληλη αρχιτεκτονική και οι δυνατότητες ταχείας επεξεργασίας των FPGAs μπορούν να οδηγήσουν σε σημαντική αύξηση της απόδοσης των υπολογισμών του SHA-3 [174].

Ως αποτέλεσμα αυτών των πλεονεκτημάτων, έχουν προταθεί στη βιβλιογραφία διάφορες στρατηγικές για τη βέλτιστη υλοποίηση του αλγορίθμου SHA-3 σε υλικό. Οι προσεγγίσεις αυτές εστιάζουν είτε στη μείωση της ενεργειακής κατανάλωσης, είτε στη βελτιστοποίηση της χρήσης των υλικών πόρων (περιοχή κάλυψης), είτε στην επίτευξη μέγιστης ταχύτητας επεξεργασίας, αναδεικνύοντας έτσι την πολυδιάστατη φύση του σχεδιασμού αποδοτικών κρυπτογραφικών συστημάτων.

Συνολικά, οι συνεισφορές που παρουσιάζονται στο παρόν κεφάλαιο συνοψίζονται ως εξής:

- Προτείνεται μια στρατηγική επιτάχυνσης και βελτιστοποίησης αρχιτεκτονικού σχεδιασμού, η οποία βασίζεται στην τεχνική ξετυλίγματος (unrolling) του αλγορίθμου SHA-3. Η προσέγγισή μας επιτυγχάνει σημαντική επιτάχυνση και μεγιστοποίηση των μετρικών ρυθμαπόδοσης και αποδοτικότητας σε συσκευές FPGA, καθιστώντας την ιδανική επιλογή για απαιτητικές κρυπτογραφικές εφαρμογές υψηλής απόδοσης.
- Η ορθότητα και η αξιοπιστία της προτεινόμενης αρχιτεκτονικής επιβεβαιώθηκαν με τη χρήση επίσημων δοκιμαστικών σεναρίων που έχουν καθιερωθεί από το Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας. Παράλληλα, διενεργήθηκε εκτεταμένη συγκριτική ανάλυση, κατά την οποία εξετάστηκαν λεπτομερώς η απαιτούμενη επιφάνεια υλικού, οι επιδόσεις ως προς τη ρυθμαπόδοση, η συχνότητα λειτουργίας και η συνολική αποδοτικότητα του συστήματος. Τα αποτελέσματα αυτά συγκρίθηκαν με εκείνα που αναφέρονται στη διεθνή βιβλιογραφία, αναδεικνύοντας σαφώς τη συνεισφορά της προτεινόμενης λύσης σε ό,τι αφορά την επιτάχυνση και τη βελτιστοποίηση της απόδοσης.

Το υπόλοιπο κεφάλαιο οργανώνεται ως εξής: Στην Ενότητα 4.3, παραθέτουμε τις σχετικές μελέτες που είναι παρόμοιες με την έρευνά μας. Στην ενότητα 4.4 περιγράφεται η επιταχυνόμενη προτεινόμενη υλοποίηση του αλγορίθμου SHA-3 σε πλακέτες FPGA. Στην Ενότητα 4.5, παρουσιάζουμε τα πειραματικά αποτελέσματα της μελέτης μας. Στην Ενότητα 4.6, συζητάμε τα αποτελέσματα της μεθόδου μας και την συγκρίνουμε με άλλες σχετικές έρευνες. Τέλος, στην Ενότητα 4.7 συνοψίζονται τα βασικά συμπεράσματα της παρούσας μελέτης και διατυπώνονται προτάσεις για μελλοντική έρευνα.

## 4.3 Σχετικές εργασίες ξετυλίγματος υλικού

Η κρυπτογραφική ερευνητική κοινότητα έχει επιδείξει έντονο ενδιαφέρον στη βελτιστοποίηση μοντέλων, αρχιτεκτονικών και στρατηγικών για την υλοποίηση του αλγορίθμου SHA-3 σε συσκευές FPGA [141, 175, 176]. Πλήθος προτεινόμενων αρχιτεκτονικών αποσκοπούν στην επίτευξη υψηλότερης ρυθμαπόδοσης, αυξημένης αποδοτικότητας και ανώτερης συχνότητας λειτουργίας, ενώ παράλληλα επιδιώκουν τη μείωση της απαιτούμενης επιφάνειας και της κατανάλωσης ενέργειας στην πλατφόρμα FPGA [148, 177–182]. Παρά τη σημαντική πρόοδο που έχει συντελεστεί,

εξακολουθεί να υφίσταται επιτακτική ανάγκη για περαιτέρω ενίσχυση των μετρικών απόδοσης, με έμφαση στην αύξηση της ρυθμαπόδοσης και της αποδοτικότητας, καθώς και στη μείωση της χρησιμοποιούμενης επιφάνειας. Η παρούσα ενότητα εστιάζει στην παρουσίαση και ανάλυση ερευνητικών προσπαθειών που παρουσιάζουν συνάφεια με το αντικείμενο και τους στόχους της δικής μας προσέγγισης.

Στο [183], παρουσιάστηκε μια μέθοδος για αρχιτεκτονικές SHA-3 για μεγέθη εξόδου 256 και 512 bits. Το RC αποθηκεύεται σε μια κατανεμημένη ROM των $24 \times 64$ bits. Η αρχιτεκτονική Virtex-5 για μέγεθος εξόδου 256 bits χρειάζεται 1217 slices και ρολόι 277 MHz και επιτυγχάνει 12,56 Gbps και ο Virtex-7 χρειάζεται 998 slices και ρολόι 300 MHz και φτάνει σε ρυθμό μετάδοσης 13,60 Gbps. Η αρχιτεκτονική Virtex-5 για μέγεθος εξόδου 512 χρειάζεται 1200 slices και ρολόι 270 MHz και επιτυγχάνει ρυθμό μετάδοσης 6,48 Gbps, ενώ η Virtex-7 χρειάζεται 983 slices, ρολόι 298,68 MHz και επιτυγχάνει ρυθμό μετάδοσης 7,17 Gbps. Ωστόσο, αυτή η αρχιτεκτονική εμφανίζει χαμηλή συχνότητα και ρυθμοαπόδοση.

Οι Paul και Shukla [184] παρουσίασαν δύο αρχιτεκτονικές του SHA-3 για μέγεθος εξόδου 256 bits και μια μέθοδο RC με γεννήτρια για την ανάκτηση της RC με 64 bits από την ενσωματωμένη μνήμη μόνο για ανάγνωση (ROM). Η πρώτη αρχιτεκτονική χρειάζεται 4188 slices, ένα 390,53 MHz και επιτυγχάνει ρυθμό μετάδοσης 16,492 Gbps. Η δεύτερη αρχιτεκτονική χρειάζεται 7139 slices, ρολόι 234,97 MHz και επιτυγχάνει ρυθμό μετάδοσης 19,99 Gbps. Ωστόσο, αυτές οι αρχιτεκτονικές παρήγαγαν χαμηλή συχνότητα και αυξημένη επιφάνεια επικάλυψης.

Οι Wong et al. [144] παρουσίασαν μια μέθοδο για τη μείωση της απαιτούμενης έκτασης για τη ROM με τη μείωση του μήκους των bits από 64 σε 8 και παρουσίασαν πέντε διαφορετικές αρχιτεκτονικές SHA-3 για το μέγεθος εξόδου 512. Η πρώτη αρχιτεκτονική χρειάζεται 871 slices και ρολόι 153 MHz, επιτυγχάνοντας ρυθμό μετάδοσης 3,68 Gbps και 4,22 Mbps/slices. Η δεύτερη αρχιτεκτονική χρειάζεται 1393 slices και ρολόι 335 MHz, επιτυγχάνοντας ρυθμό μετάδοσης 8,04 Gbps και 5,77 Mbps/Slices. Η τρίτη αρχιτεκτονική χρειάζεται 2145 slices και ρολόι 45 MHz, επιτυγχάνοντας ρυθμό μετάδοσης 2,16 Gbps και 1,00 Mbps/Slices. Η τέταρτη αρχιτεκτονική χρειάζεται 1416 slices και ρολόι 85 MHz και επιτυγχάνει ρυθμό μετάδοσης 4,08 Gbps και 2,88 Mbps/Slices. Η πέμπτη αρχιτεκτονική χρειάζεται 1406 slices και ρολόι 344 MHz, επιτυγχάνοντας ρυθμό μετάδοσης 16,51 Gbps και 11,47 Mbps/Slices. Παρόλο που η συνολική καταλαμβανόμενη επιφάνεια δεν ήταν πολύ

μεγάλη, χρειαζόταν περισσότερη από την υψηλότερη συχνότητα που επιτυγχάνεται για να είναι ικανοποιητική.

Η προσέγγιση unrolling, η οποία μειώνει τον συνολικό αριθμό των κύκλων ρολογιού με μια πρόσθετη πράξη στρογγυλοποίησης, εφαρμόζεται στη Virtex-5 στο [185] για μέγεθος εξόδου 256 bits και φτάνει σε ρυθμαπόδοση 5,38 Gbps. Στο [143] μειώθηκε επίσης ο αριθμός των κύκλων ρολογιού για όλα τα μεγέθη εξόδου με τη χρήση της προσέγγισης unrolling με Virtex-5 και Virtex-6. Παρόλα αυτά, η συχνότητα και η ρυθμαπόδοση που παρέχει αυτή η σχεδίαση θα μπορούσαν να είναι καλύτερες.

Στο [186], προτάθηκε μια βασική αρχιτεκτονική του SHA-3 για Virtex-7 FPGA με μέγεθος εξόδου 512 bits. Μια κατανεμημένη ROM με διάσταση $24 \times 64$ bits χρησιμοποιήθηκε για την αποθήκευση των σταθερών γύρου (RC). Η αρχιτεκτονική λειτουργούσε με 1454 slices και χρησιμοποιούσε συχνότητα ρολογιού 374,035 MHz. Αυτή η σχεδίαση απέφερε ρυθμό μετάδοσης 7,979 Gbps και ποσοστό αποδοτικότητας 5,49 Mbps/slices. Ωστόσο, η επιφάνεια σε (slices) και η απόδοση επηρεάστηκαν αρνητικά από αυτή την εφαρμογή.

Οι Assad et al. [187] πρότειναν τρεις υλοποιήσεις του SHA-3 σε Virtex-5 και Virtex-6 FPGA. Η εστίαση ήταν σε όλα τα μεγέθη out-put. Αξίζει να σημειωθεί ότι τα RC που απαιτούνται για την υλοποίηση SHA-3 αποθηκεύτηκαν σε μια ROM των $24 \times 64$ bits. Η βασική υλοποίηση με χρήση Virtex-5 για μέγεθος εξόδου 512 bits απαιτούσε 935 slices και λειτουργούσε σε συχνότητα ρολογιού 338,409 MHz. Αυτή η σχεδίαση πέτυχε ρυθμαπόδοση 8,12 Gbps και ρυθμό αποδοτικότητας 8,68 Mbps/Slices. Η βασική υλοποίηση με χρήση Virtex-6 για μέγεθος εξόδου 512 bits απαιτούσε 1019 slices και λειτουργούσε σε συχνότητα ρολογιού 376,081 MHz. Αυτή η υλοποίηση πέτυχε υψηλότερη ρυθμαπόδοση 9,02 Gbps, με ρυθμό αποδοτικότητας 8,85 Mbps/Slices. Ωστόσο, η επιφάνεια σε (slices) και η αποδοτικότητα επηρεάστηκαν αρνητικά από αυτή τη σχεδίαση.

Στο [153], προτάθηκε μια βασική υλοποίηση του SHA-3 για Virtex-5 FPGA με μέγεθος εξόδου 512 bits. Οι στρογγυλές σταθερές (RC) αποθηκεύτηκαν σε μια κατανεμημένη ROM των $24 \times 64$ bits. Η υλοποίηση χρησιμοποίησε 1680 slices στο Virtex-5 FPGA και λειτούργησε σε συχνότητα ρολογιού 387 MHz. Αυτή η σχεδίαση πέτυχε ρυθμό μετάδοσης 8,06 Gbps και ρυθμό αποδοτικότητας 4,91 Mbps/Slices. Παρόλα αυτά, η επιφάνεια σε (slices) και η αποδοτικότητα που παρείχε αυτή η αρχιτεκτονική θα μπορούσαν να ήταν καλύτερες.

Η ολοκληρωμένη εξέταση και σε βάθος ανάλυση των προαναφερθεισών μεθοδολογιών, καθώς και η αξιολόγηση της επίδρασής τους στην απόδοση της αρχιτεκτονικής SHA-3, ανέδειξαν την ανάγκη για την ανάπτυξη μιας βελτιωμένης και επιταχυνόμενης αρχιτεκτονικής, ικανής να προσφέρει υψηλή ρυθμαπόδοση σε συνδυασμό με ελαχιστοποιημένη επιφάνεια υλικού. Η προτεινόμενη προσέγγιση οδηγεί σε σημαντική μείωση της απαιτούμενης επιφάνειας (slices), ενώ παράλληλα επιτυγχάνει ουσιαστική επιτάχυνση και αύξηση της ρυθμαπόδοσης έναντι των υφιστάμενων λύσεων.

Η αρχιτεκτονική που αναπτύχθηκε για τον αλγόριθμο SHA-3 αξιολογήθηκε και επαληθεύτηκε βάσει των ίδιων κριτηρίων που χρησιμοποιούνται στη διεθνή βιβλιογραφία, διασφαλίζοντας την αντικειμενικότητα της σύγκρισης και επιβεβαιώνοντας τα πλεονεκτήματα της επιτάχυνσης και της βελτιστοποίησης που προσφέρει η νέα μεθοδολογία.

## 4.4 Προτεινόμενο αρχιτεκτονικό σύστημα βελτιστοποίησης

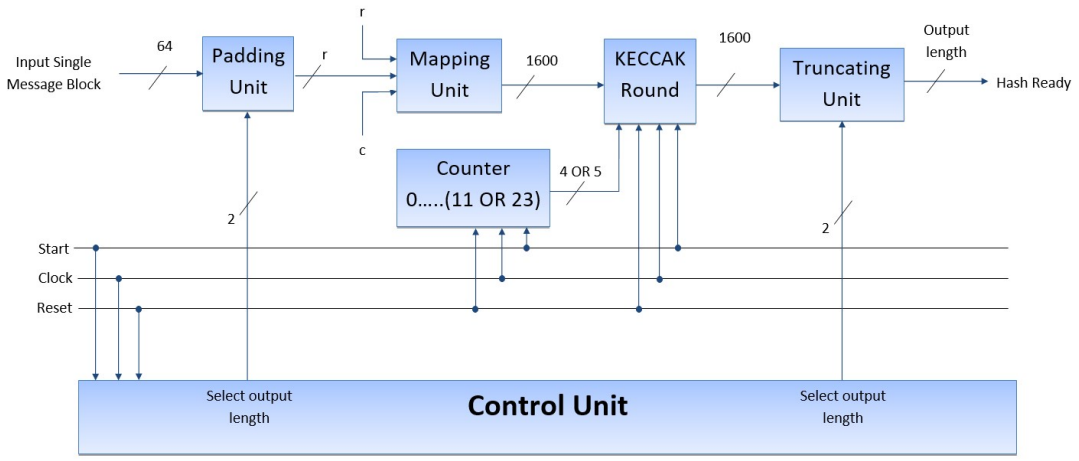
Η παρούσα ενότητα εστιάζει στην ανάλυση της αρχιτεκτονικής σχεδίασης που υλοποιήθηκε για όλα τα υποστηριζόμενα μήκη εξόδου (576, 832, 1088, 1152 bits) του αλγορίθμου SHA-3. Ο βασικός στόχος της ερευνητικής μας προσέγγισης είναι η επίτευξη μέγιστης επιτάχυνσης και υψηλότερης ρυθμαπόδοσης (Gbps), παράλληλα με τη μείωση της απαιτούμενης επιφάνειας υλικού (slices) στο σύστημα. Η στρατηγική αυτή αποσκοπεί στη βέλτιστη εκμετάλλευση των δυνατοτήτων του υλικού, επιτυγχάνοντας επιτάχυνση των υπολογισμών του SHA-3 και ενισχύοντας τη συνολική απόδοση και αποδοτικότητα της αρχιτεκτονικής.

### 4.4.1 Ο αρχιτεκτονικός σχεδιασμός του SHA-3

Η αρχιτεκτονική του συστήματός μας παρουσιάζεται στο Σχήμα 4.1. Η αρχιτεκτονική περιλαμβάνει τα εξής βήματα:

- συμπλήρωση
- χαρτογράφηση

- γύρος Keccak
- αποκοπή
- έλεγχος
- μετρητής




Σχήμα 4.1: Προτεινόμενο αρχιτεκτονικό σύστημα βελτιστοποίησης του SHA-3.

Ο γύρος Keccak αποτελεί τον πυρήνα της προτεινόμενης αρχιτεκτονικής σχεδίασης, υπεύθυνος για την υλοποίηση των βασικών υπολογιστικών διαδικασιών της συνάρτησης κατακερματισμού. Η μονάδα ελέγχου αναλαμβάνει τη διαχείριση, τον ακριβή συγχρονισμό και τη συντονισμένη επικοινωνία της ροής δεδομένων εντός του συστήματος, εξασφαλίζοντας την ομαλή αλληλεπίδραση όλων των επιμέρους δομικών στοιχείων της αρχιτεκτονικής. Τα δεδομένα του μηνύματος εισόδου έχει οριστεί στα 64 bit, ενώ οι διακριτές επιλογές για το μήκος της εξόδου παρατίθενται αναλυτικά στον Πίνακα 4.1.

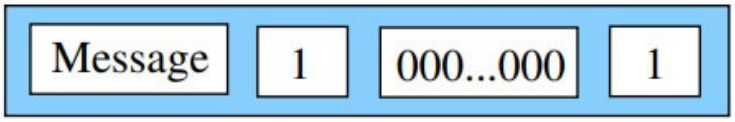
Πίνακας 4.1: Οι τέσσερις διαφορετικές τιμές για το επιλεγμένο μήκος εξόδου του Αλγόριθμου SHA-3.

| **Τιμή εισόδου** | 00 | 01 | 10 | 11 |
|---|---|---|---|---|
| **Κατακερματισμένη έξοδος** | 224 | 256 | 384 | 512 |

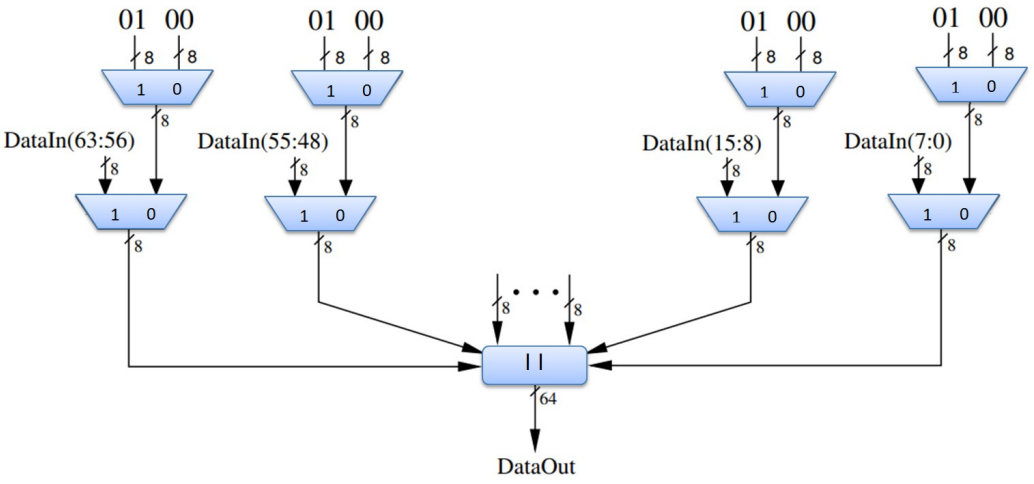
### 4.4.2 Συμπλήρωση, αντιστοίχιση και αποκοπή μονάδας

Η μονάδα συμπλήρωσης του αλγορίθμου SHA-3 για το μήνυμα εισόδου απεικονίζεται στο Σχήμα 4.2. Για ένα μήνυμα εισόδου μήκους 64 bit, η διαδικασία πλήρωσης

περιλαμβάνει την προσθήκη ενός αρχικού "1" bit, ακολουθούμενου από τον απαραίτητο αριθμό "0" bits, έτσι ώστε το συνολικό μήκος του μηνύματος να φτάσει μία θέση πριν το αμέσως επόμενο πολλαπλάσιο του $r$ (576, 832, 1088 ή 1152 bits). Τέλος, προστίθεται ένα ακόμη "1" bit ώστε το τελικό μήκος του μηνύματος να είναι ακριβώς πολλαπλάσιο του $r$. Αυτή η διαδικασία εξασφαλίζει τη σωστή διαμόρφωση του μηνύματος, όπως ορίζουν οι προδιαγραφές του αλγορίθμου, προετοιμάζοντάς το για την επεξεργασία από τον αλγόριθμο κατακερματισμού [154].




Σχήμα 4.2: Η μονάδα συμπλήρωσης του αλγόριθμου κατακερματισμού SHA-3.




Σχήμα 4.3: Διάγραμμα του μπλοκ συμπλήρωσης του SHA-3.

Το βασικό δομικό διάγραμμα του αλγορίθμου SHA-3 παρουσιάζεται στο Σχήμα 4.3 και περιλαμβάνει τη χρήση πολυπλεκτών 2 προς 1 για την επεξεργασία μηνυμάτων εισόδου μεγέθους 64 bits. Η μονάδα συμπλήρωσης υλοποιείται μέσω ενός πολυπλέκτη 4 προς 1, ο οποίος επιτρέπει την επιλογή του κατάλληλου σχήματος συμπλήρωσης, ανάλογα με το επιθυμητό μήκος εξόδου. Για παράδειγμα, όταν επιλέγονται 224 bits ως μήκος εξόδου, εφαρμόζεται το σχήμα συμπλήρωσης που αντιστοιχεί σε $r = 1152$. Τα $r$ bits του συμπληρωμένου μηνύματος (Pad) εισάγονται στη μονάδα χαρτογράφησης και υποβάλλονται σε πράξη XOR με τα αρχικά $r$ bits. Στη συνέχεια, το αποτέλεσμα αυτής της πράξης προσαρτάται με τα αρχικά $c$ bits, σύμφωνα με τις προδιαγραφές του αλγορίθμου [30].

Για τη σωστή διαμόρφωση της εξόδου, απαιτείται κατάλληλος μετασχηματισμός των δεδομένων, όπως περιγράφεται στην Εξίσωση (4.1). Η μονάδα αποκοπής, βασιζόμενη στην εν λόγω εξίσωση, απομονώνει τα απαραίτητα ψηφία της κατάστασης ανάλογα με το επιλεγμένο μήκος εξόδου (576, 832, 1088 ή 1152 bits) και υλοποιείται μέσω πολυπλέκτη 4 προς 1.

$$\text{State}[x, y, z] = ((\text{Pad } r \text{ XOR } r)|c) \times [64 \times (5y + x) + z] \tag{4.1}$$

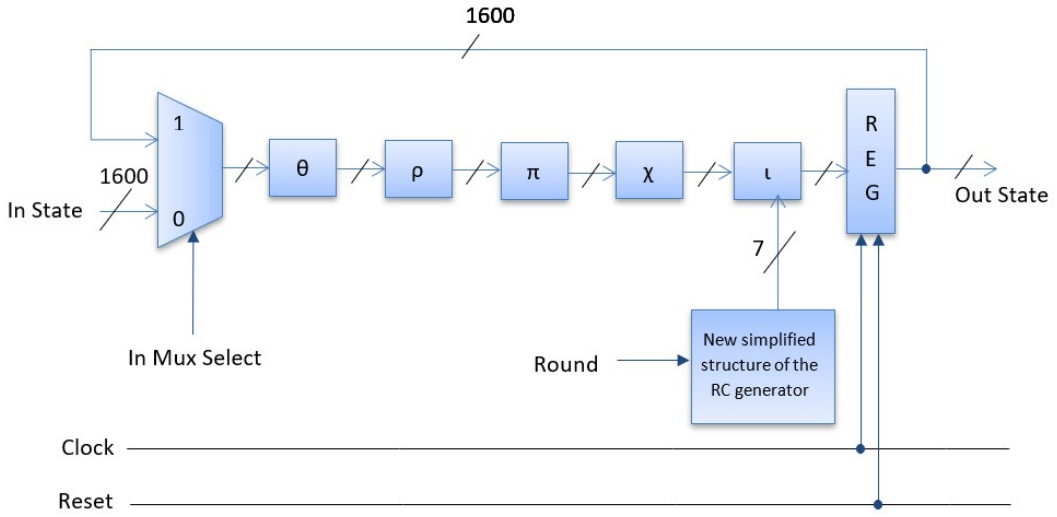
Η παραπάνω εξίσωση ορίζει τον τρόπο με τον οποίο διαμορφώνεται η τελική κατάσταση εξόδου του συστήματος, με βάση το αποτέλεσμα της πράξης XOR μεταξύ του συμπληρωμένου μηνύματος και των $r$ bits, καθώς και την προσάρτηση των $c$ bits. Η μονάδα αποκοπής διασφαλίζει ότι η έξοδος ανταποκρίνεται ακριβώς στο επιθυμητό μήκος, ανάλογα με τις απαιτήσεις της εφαρμογής.

### 4.4.3 Η αρχιτεκτονική του αλγορίθμου SHA-3

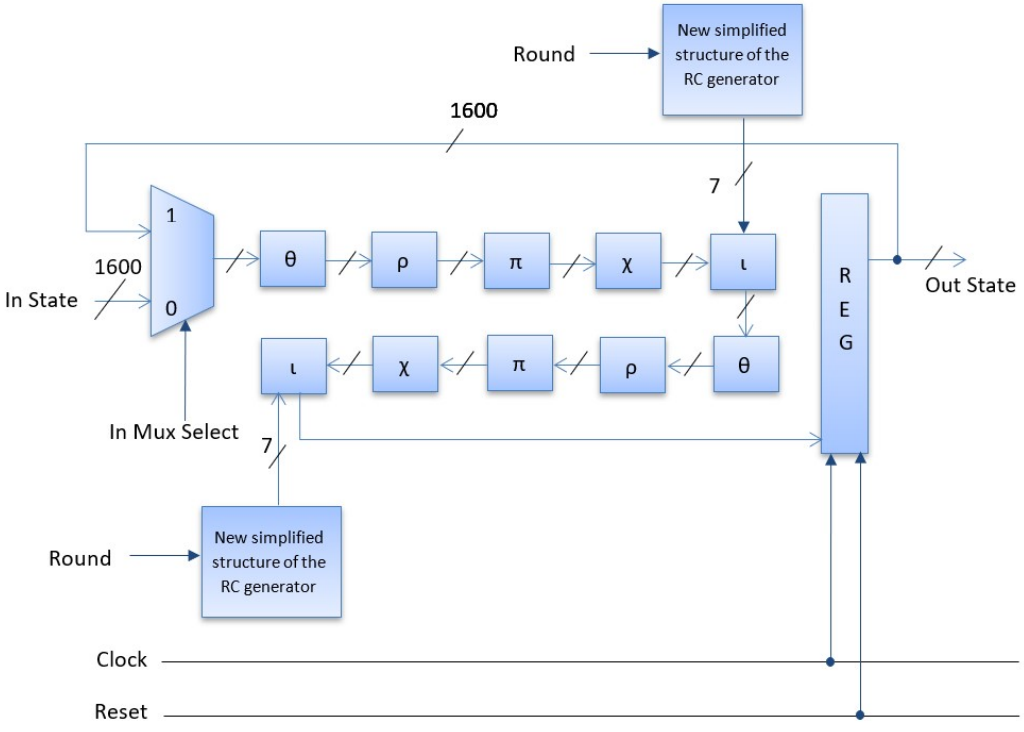
Στην παρούσα μελέτη, ένας από τους βασικούς ερευνητικούς μας στόχους ήταν η επιτάχυνση της εκτέλεσης μέσω της μείωσης του συνολικού αριθμού κύκλων ρολογιού, επιτυγχάνοντας παράλληλα τη διατήρηση της ελάχιστης δυνατής επιφάνειας σε (slices). Η βασική αρχιτεκτονική του μπλοκ γύρων μετατροπής παρουσιάζεται στο Σχήμα 4.4, όπου ο μετρητής κυμαίνεται από 0 έως 23, υποδεικνύοντας ότι στην αρχική μορφή δεν εφαρμόζεται βελτιστοποίηση για μείωση του συνολικού αριθμού των κύκλων ρολογιού.

Η στρατηγική ξετυλίγματος (loop unrolling) αποτελεί μια τεχνική βελτιστοποίησης που στοχεύει στη μείωση της επιβάρυνσης από βρόχους σε υλοποιήσεις αλγορίθμων, αυξάνοντας την αποδοτικότητα και μειώνοντας το χρόνο εκτέλεσης. Στη βασική υλοποίηση του αλγορίθμου SHA-3, όπως απεικονίζεται στο Σχήμα 4.4, ο υπολογισμός πραγματοποιείται μέσω ενός ενιαίου μπλοκ μετασχηματισμών ακολουθώντας τη σειρά των λειτουργιών $\theta, \rho, \pi, \chi,$ και $\iota$. Η εφαρμογή της τεχνικής ξετυλίγματος επιδιώκει την περαιτέρω βελτίωση της απόδοσης του αλγορίθμου, επιτρέποντας την εκτέλεση πολλαπλών μπλοκ μετασχηματισμών εντός ενός κύκλου ρολογιού, μειώνοντας έτσι σημαντικά τη συνολική καθυστέρηση.

Στην παρούσα εργασία εφαρμόστηκε η στρατηγική ξετυλίγματος δύο σταδίων (loop unrolling by 2), όπως απεικονίζεται στο Σχήμα 4.5. Συγκεκριμένα, ένα επιπλέον




Σχήμα 4.4: Ο αλγόριθμος SHA-3 με 24 κύκλους ρολογιού.




Σχήμα 4.5: Ο αλγόριθμος SHA-3 με 12 κύκλους ρολογιού.

μπλοκ μετασχηματισμών ενσωματώθηκε εντός της ενότητας SHA-3, με αποτέλεσμα η εκτέλεση δύο πλήρων γύρων μετασχηματισμών, ακολουθώντας τη σειρά $\boldsymbol{\theta} \rightarrow \boldsymbol{\rho} \rightarrow \boldsymbol{\pi} \rightarrow \boldsymbol{\chi} \rightarrow \boldsymbol{\iota} \rightarrow \boldsymbol{\theta} \rightarrow \boldsymbol{\rho} \rightarrow \boldsymbol{\pi} \rightarrow \boldsymbol{\chi} \rightarrow \boldsymbol{\iota}$, να πραγματοποιείται εντός ενός μόνο κύκλου ρολογιού. Η εφαρμογή αυτής της τεχνικής επιτρέπει τη μείωση του συνολικού αριθμού κύκλων ρολογιού που απαιτούνται για την ολοκλήρωση του αλγορίθμου SHA-3 στο ήμισυ.

Η εφαρμογή της τεχνικής επιτάχυνσης μέσω της στρατηγικής ξετυλίγματος δύο σταδίων επιφέρει μείωση στο συνολικό αριθμό κύκλων ρολογιού κατά το ήμισυ, επιτρέποντας την ολοκλήρωση των υπολογισμών σε 12 κύκλους ρολογιού. Αυτή η μείωση του χρόνου εκτέλεσης συνεπάγεται σημαντική βελτίωση της απόδοσης της υλοποίησης του αλγορίθμου SHA-3, καθιστώντας την ιδιαίτερα κατάλληλη για εφαρμογές σε πλακέτες FPGA όπου απαιτούνται υψηλές επιδόσεις και αποδοτικότητα.

Η έρευνά μας προτείνει μια απλουστευμένη δομή για τη γεννήτρια RC που βελτιώνει σημαντικά την απόδοση του αλγορίθμου μειώνοντας παράλληλα τους πόρους υλικού στο ελάχιστο. Η γεννήτρια RC είναι ένα κρίσιμο στοιχείο του αλγορίθμου SHA-3. Η κύρια λειτουργία της είναι να παράγει μια ακολουθία ψευδοτυχαίων bits που χρησιμοποιούνται για την κρυπτογράφηση των δεδομένων εισόδου. Η υπάρχουσα γεννήτρια RC αποτελείται από 24 σύνολα των 64 bits, γεγονός που οδηγεί σε πολλούς υπολογισμούς στο βήμα $\iota$ του αλγορίθμου SHA-3. Στο βήμα αυτό, χρειάζεται να εκτελεστεί ένας μεγάλος αριθμός πράξεων XOR, γεγονός που μπορεί να μειώσει την ρυθμαπόδοση και την αποδοτικότητα του αλγορίθμου, ιδίως σε συσκευές FPGA με περιορισμένες δυνατότητες.

Για να ξεπεράσουμε αυτό το πρόβλημα, χρησιμοποιούμε μια απλοποιημένη δομή για τη γεννήτρια RC που αποτελείται μόνο από 7 bit [30, 103]. Μειώνοντας τον αριθμό των bits στη γεννήτρια, μειώνουμε αποτελεσματικά τους υπολογισμούς που απαιτούνται στο βήμα $\iota$, με αποτέλεσμα τη βελτίωση της ρυθμαπόδοσης και της αποδοτικότητας. Η μείωση του συνολικού αριθμού των bits μειώνει επίσης τους πόρους υλικού που απαιτούνται για τη γεννήτρια RC, με αποτέλεσμα μια πιο συμπαγή σχεδίαση, ιδανική για συσκευές FPGA με περιορισμένους πόρους.

Το βήμα $\iota$ συνίσταται στην τροποποίηση ορισμένων από τα bits της συστοιχίας καταστάσεων A, όπως φαίνεται στην εξίσωση (4.2).

$$A'[x, y, z] = A[x, y, z] \,\mathrm{XOR}\, RC\,[i_r] \tag{4.2}$$

Σύμφωνα με τις προδιαγραφές του SHA-3, τα RC δίνονται από την εξίσωση (4.3),

$$\mathrm{RC}\left[i_{\mathrm{r}}\right][0][0]\left[2^{j}-1\right]=\mathrm{rc}\left[j+7i_{\mathrm{r}}\right],\ \text{for all } 0\leq j\leq \ell \tag{4.3}$$

και όλες οι άλλες τιμές του $RC[i_{\mathrm{r}}][x][y][z]$ είναι μηδενικές. Από την εξίσωση (4.3) προκύπτει ότι μόνο 7 από τα 64 bits μπορούν να έχουν την τιμή 1. Στον Πίνακα 4.2 παρουσιάζονται οι συγκεκριμένες θέσεις για τα 7 bit όπου $\ell = 6$, σύμφωνα με τις προδιαγραφές του SHA-3.

Πίνακας 4.2: Ειδικές θέσεις για τα 7 bits με τιμή 1

| **j** | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| **[z]** | 0 | 1 | 3 | 7 | 15 | 31 | 63 |

Έτσι, μόνο αυτά τα 7 από τα 64 bits είναι οι θεμελιώδεις σταθερές και εμφανίζονται σε συγκεκριμένες θέσεις με μη μηδενικά bits, ενώ οι υπόλοιπες θέσεις είναι μηδενικές. Οι συγκεκριμένες θέσεις bit που φέρουν την τιμή $1$ είναι οι 0,1,3,7,15,31 και 63, ενώ οι υπόλοιπες είναι $0$. Ένα παράδειγμα της απλουστευμένης δομής για το $RC[3]$ του Πίνακα 4.3 παρουσιάζεται στον Πίνακα 4.4. Έτσι, επτά συγκεκριμένα bit μπορούν να ρυθμιστούν για την πύλη XOR στη συστοιχία καταστάσεων **A**.

Πίνακας 4.3: Η απλουστευμένη δομή των σταθερών γύρου $RC_i$ στο βήμα $\iota$ του αλγορίθμου SHA-3.

| | | | |
|---|---|---|---|
| $RC_0$ | 1000000 | $RC_{12}$ | 1111110 |
| $RC_1$ | 0101100 | $RC_{13}$ | 1111001 |
| $RC_2$ | 0111101 | $RC_{14}$ | 1011101 |
| $RC_3$ | 0000111 | $RC_{15}$ | 1100101 |
| $RC_4$ | 1111100 | $RC_{16}$ | 0100101 |
| $RC_5$ | 1000010 | $RC_{17}$ | 0001001 |
| $RC_6$ | 1001111 | $RC_{18}$ | 0110100 |
| $RC_7$ | 1010101 | $RC_{19}$ | 0110011 |
| $RC_8$ | 0111000 | $RC_{20}$ | 1001111 |
| $RC_9$ | 0011000 | $RC_{21}$ | 0001101 |
| $RC_{10}$ | 1010110 | $RC_{22}$ | 1000010 |
| $RC_{11}$ | 0110010 | $RC_{23}$ | 0010101 |

Πίνακας 4.4: Παράδειγμα της απλουστευμένης δομής του $RC[3]$ στο βήμα $\iota$.

| **Δεκαεξαδικό** | **Δυαδικό** | | | | **Θέσεις με τιμή 1** |
|---|---|---|---|---|---|
| 8000 | 1000 | 0000 | 0000 | 0000 | 0th = 0<br>1st = 0<br>3rd = 0<br>7th = 0<br>15th = 1 |
| 8000 | 1000 | 0000 | 0000 | 0000 | 31st = 1 |
| 0000 | 0000 | 0000 | 0000 | 0000 | - |
| 8000 | 1000 | 1000 | 1000 | 1000 | 63th = 1 |

## 4.5 Πειραματικά αποτελέσματα

Στα πειράματά μας, χρησιμοποιήσαμε τις πλακέτες FPGA Virtex-5, Virtex-6, Virtex-7 και Artix-7, προκειμένου να κάνουμε μια δίκαιη αξιολόγηση μεταξύ του προτεινόμενου σχεδιασμού και των άλλων υφιστάμενων έργων, ενώ παράλληλα παρέχουμε μια διεξοδική, ολοκληρωμένη σύγκριση σε διαφορετικές πλακέτες FPGA για μια ευρύτερη αξιολόγηση της αποδοτικότητας και της ρυθμαπόδοσης του σχεδιασμού.

Το λογισμικό Xilinx ISE χρησιμοποιήθηκε για την υλοποίηση της σχεδίασης στις πλακέτες Virtex- 5/Virtex-6. Για τις πλακέτες Virtex-7/Artix-7 χρησιμοποιήθηκε για την υλοποίηση της αρχιτεκτονικής το λογισμικό Xilinx Vivado. Η υλοποίηση έγινε με τη γλώσσα περιγραφής υλικού ολοκληρωμένων κυκλωμάτων πολύ υψηλής ταχύτητας (VHDL). Ο προτεινόμενος σχεδιασμός προσομοιώθηκε και επιβεβαιώθηκε για την λειτουργικότητα του στο σύνολο των πλακετών με έγκυρα δείγματα που παρέχονται από το NIST [156].

### 4.5.1 Μετρήσεις επιδόσεων

Προκειμένου να διασφαλιστεί μια δίκαιη σύγκριση μεταξύ του προτεινόμενου σχεδιασμού και άλλων υφιστάμενων εργασιών, χρησιμοποιήσαμε τις καθιερωμένες μετρικές της αποδοτικότητας και της ρυθμαπόδοσης που χρησιμοποιούνται στη βιβλιογραφία [141, 157, 188, 189]. Η τυποποίηση αυτή των μετρικών επιτρέπει την συνεπή σύγκριση των αποτελεσμάτων μας με εκείνα των προηγούμενων μελετών, διευκολύνοντας την ακριβή αξιολόγηση και ανάδειξη των βελτιώσεων στην απόδοση που επιτυγχάνει ο προτεινόμενος σχεδιασμός.

Η ρυθμαπόδοση συμβολίζει τον συνολικό αριθμό των bit που επεξεργάζονται ανά μονάδα περιόδου (χρόνου) και ορίζεται σε Gbps ή Mbps. Η ρυθμαπόδοση υπολογίζεται χρησιμοποιώντας την Εξίσωση (4.4).

$$Throughput = \frac{Bmb}{Ccmb} \times Max_f \tag{4.4}$$

Στην Εξίσωση (4.4), τα Bmb (bits in a message block) είναι το μέγεθος του ρυθμού bit $r$ (576, 832, 1088, 1152), το $\text{Max}_f$ είναι η μέγιστη συχνότητα περιοδικότητας ρολογιού και το Ccmb (clock cycles per message block) αντιπροσωπεύουν τον αριθμό της επανάληψης που απαιτείται για τις πέντε ειδικές λειτουργίες: $\theta, \rho, \pi, \chi,$ και $\iota$ για τη δημιουργία της τιμής κατακερματισμού. Η αποδοτικότητα υπολογίζεται χρησιμοποιώντας την Εξίσωση (4.5).

$$Efficiency = \frac{Throughput}{Area} \tag{4.5}$$

### 4.5.2 Αποτελέσματα

Ο αρχιτεκτονικός σχεδιασμός που παρουσιάσαμε επιτυγχάνει υψηλή ρυθμαπόδοση και διασφαλίζει τη ελαχιστοποίηση των πόρων υλικού στην επιφάνεια σε (slices) για διάφορα μήκη εξόδου που απαιτούνται για την παραγωγή μιας τιμής κατακερματισμού. Τα αποτελέσματα της υλοποίησης αυτού του αρχιτεκτονικού σχεδίου συνοψίζονται στον Πίνακα 4.5, ο οποίος δείχνει τη μέγιστη συχνότητα και ρυθμαπόδοση όλων των μηκών εξόδου.

Όπως φαίνεται στην Εξίσωση (4.5), η μείωση του συνολικού αριθμού κύκλων ρολογιού και η ελαχιστοποίηση της επιφάνειας επικάλυψης σε (slices) αυξάνει την αποδοτικότητα, που ήταν ο πρωταρχικός μας στόχος. Η στρατηγική που υιοθετήθηκε εστιάζει στη μείωση του συνολικού αριθμού επαναλήψεων που απαιτούνται για τη δημιουργία μιας τιμής κατακερματισμού, με στόχο την αύξηση της αποδοτικότητας του συστήματος. Επιπλέον, η απλοποιημένη δομή της γεννήτριας RC προσφέρει σημαντικά πλεονεκτήματα, όπως η ελαχιστοποίηση της χρήσης υλικών πόρων που απαιτούνται για την υλοποίηση, η επιτάχυνση του σχεδιαστικού κύκλου μέσω της μείωσης της υπολογιστικής και υλοποιητικής

πολυπλοκότητας, καθώς και η επίτευξη υψηλότερων συχνοτήτων ρολογιού, οι οποίες συντελούν στη βελτιωμένη συνολική αποδοτικότητα της αρχιτεκτονικής.

Σύμφωνα με τα πειραματικά αποτελέσματα, ο προτεινόμενος αρχιτεκτονικός σχεδιασμός επιτυγχάνει μέγιστη ρυθμαπόδοση 36,358 Gbps για μήκος εξόδου 224 και 18,179 Gbps για μήκος εξόδου 512, με απαιτούμενη επιφάνεια 1375 slices. Ο Πίνακας 4.5 παρουσιάζει μια συγκριτική αποτίμηση των επιδόσεων της προτεινόμενης αρχιτεκτονικής σε σχέση με πρόσφατες μελέτες της διεθνούς βιβλιογραφίας, προσφέροντας μια δίκαιη και αντικειμενική σύγκριση.

Πέραν των αποτελεσμάτων του Πίνακα 4.5, η ρυθμαπόδοση του προτεινόμενου σχεδιασμού αξιολογήθηκε και ως λόγος της επιτευχθείσας ρυθμαπόδοσης (σε Mbps) προς την κατανάλωση υλικού, η οποία μετρήθηκε ως ο αριθμός slices που χρησιμοποιήθηκαν. Τα αποτελέσματα της εν λόγω αξιολόγησης συνοψίζονται στον Πίνακα 4.6 για όλα τα εξεταζόμενα μήκη εξόδου, αναδεικνύοντας περαιτέρω την αποδοτικότητα της αρχιτεκτονικής σε όρους σχέσης απόδοσης–πόρων.

Πίνακας 4.5: Τα αποτελέσματα της εφαρμογής όσον αφορά την ρυθμαπόδοση και τη σύγκριση.

| Σχεδιασμός | Συσκευή FPGA | Κύκλοι ρολογιού | Μέγιστη Συχνότητα (Mhz) | Ρυθμαπόδοση (r = 1152) | Ρυθμαπόδοση (r = 1088) | Ρυθμαπόδοση (r = 832) | Ρυθμαπόδοση (r = 576) |
|---|---|---|---|---|---|---|---|
| [183] | Virtex-5 | 24 | 277 | - | 12,56 | - | 6,48 |
| | Virtex-7 | 24 | 300 | - | 13,60 | - | 7,17 |
| [184] | Artix-7 | 24 | 390,53 | - | - | - | 16,492 |
| | | 12 | 234,97 | - | - | - | 19,99 |
| [144] | Virtex-6 | 24 | 153 | - | - | - | 3,68 |
| | | 12 | 344 | - | - | - | 16,51 |
| [185] | Virtex-5 | 24 | 111,732 | - | 4,67 | - | - |
| | | 16 | 84,21 | - | 5,38 | - | - |
| [143] | Virtex-5 | 24 | 326,38 | 15,66 | 14,79 | 11,31 | 7,83 |
| | | 12 | 192,25 | 18,45 | 17,43 | 13,32 | 9,228 |
| | Virtex-6 | 24 | 413,77 | 19,86 | 18,75 | 14,34 | 9,93 |
| | | 12 | 232,45 | 22,31 | 21,07 | 16,11 | 11,15 |
| [186] | Virtex-7 | 24 | 374,035 | - | - | - | 7,979 |
| [187] | Virtex-5 | 24 | 338,409 | 15,86 | 15,34 | 11,73 | 8,12 |
| | Virtex-6 | 24 | 376,081 | 17,63 | 17,05 | 13,04 | 9,02 |
| [153] | Virtex-5 | 24 | 387 | - | - | - | 8,06 |
| **Παρούσα εργασία** | Virtex-5 | 24 | 347,49 | 16,680 | 15,753 | 12,046 | 8,340 |
| | | 12 | 203,28 | 19,515 | 18,431 | 14,094 | 9,757 |
| | Virtex-6 | 24 | 438,49 | 21,048 | 19,878 | 15,201 | 10,524 |
| | | 12 | 347,84 | 33,393 | 31,537 | 24,117 | 16,696 |
| | Virtex-7 | 24 | 498,27 | 23,917 | 22,588 | 17,273 | 11,958 |
| | | 12 | 378,73 | 36,358 | 34,338 | 26,259 | 18,179 |
| | Artix-7 | 24 | 397,41 | 19,075 | 18,015 | 13,776 | 9,537 |
| | | 12 | 254,46 | 24,428 | 23,071 | 17,642 | 12,214 |

Πίνακας 4.6: Τα αποτελέσματα της εφαρμογής από άποψη αποδοτικότητας και σύγκρισης.

| **Σχεδιασμός** | **Συσκευή FPGA** | **Κύκλοι ρολογιού** | **Επιφάνεια (συνολικός αριθμός από slices)** | **Αποδοτικότητα (r = 1152)** | **Αποδοτικότητα (r = 1088)** | **Αποδοτικότητα (r = 832)** | **Αποδοτικότητα (r = 576)** |
|---|---|---|---|---|---|---|---|
| [183] | Virtex-5 | 24 | 1217 | - | 10,31 | - | 5,4 |
| | Virtex-7 | 24 | 998 | - | 13,63 | - | 7,27 |
| [184] | Artix-7 | 24 | 4188 | - | - | - | 3,93 |
| | | 12 | 7139 | - | - | - | 2,80 |
| [144] | Virtex-6 | 24 | 871 | - | - | - | 4,22 |
| | | 12 | 1406 | - | - | - | 11,47 |
| [185] | Virtex-5 | 24 | 1434 | - | 3,32 | - | - |
| | | 16 | 1562 | - | 3,44 | - | - |
| [143] | Virtex-5 | 24 | 1365 | 11,47 | 10,83 | 8,28 | 5,73 |
| | | 12 | 2144 | 8,60 | 8,13 | 6,21 | 4,30 |
| | Virtex-6 | 24 | 1432 | 13,87 | 13,10 | 10,02 | 6,93 |
| | | 12 | 3557 | 6,27 | 5,93 | 4,53 | 3,14 |
| [186] | Virtex-7 | 24 | 1454 | - | - | - | 5,49 |
| [187] | Virtex-5 | 24 | 935 | 16,96 | 16,40 | 12,54 | 8,68 |
| | Virtex-6 | 24 | 1019 | 17,30 | 16,73 | 12,80 | 8,85 |
| [153] | Virtex-5 | 24 | 1680 | - | - | - | 4,91 |
| **Παρούσα εργασία** | Virtex-5 | 24 | 868 | 19,22 | 18,15 | 13,88 | 9,61 |
| | | 12 | 1112 | 17,55 | 16,57 | 12,67 | 8,77 |
| | Virtex-6 | 24 | 946 | 22,25 | 21,01 | 16,07 | 11,12 |
| | | 12 | 1287 | 25,95 | 24,50 | 18,74 | 12,97 |
| | Virtex-7 | 24 | 1094 | 21,86 | 20,65 | 15,79 | 10,93 |
| | | 12 | 1375 | 26,44 | 24,97 | 19,10 | 13,22 |
| | Artix-7 | 24 | 902 | 21,14 | 19,97 | 15,27 | 10,57 |
| | | 12 | 1184 | 20,63 | 19,48 | 14,90 | 10,31 |

## 4.6 Συζήτηση αποτελεσμάτων

Η απόδοση και η επιφάνεια σε slices αποτελούν θεμελιώδεις μετρικές αξιολόγησης της αποτελεσματικότητας αρχιτεκτονικών υλοποιήσεων, ιδίως σε εφαρμογές που άπτονται της ασφάλειας πληροφοριών και απαιτούν υψηλή αποδοτικότητα σε επίπεδο υλικού. Η απόδοση αποτυπώνει την ικανότητα του συστήματος να επεξεργάζεται δεδομένα με υψηλή ταχύτητα, ενώ η επιφάνεια σε slices αντανακλά το εύρος των υλικών πόρων που απαιτούνται για την υλοποίηση του εκάστοτε αλγορίθμου, αποτελώντας κρίσιμο δείκτη της συνολικής αποδοτικότητας του σχεδιασμού.

Στην πλακέτα Virtex-5 με 24 συνολικούς κύκλους ρολογιού, το σχέδιο [153] παρουσίασε την υψηλότερη κατανάλωση υλικών πόρων, καταλαμβάνοντας 1680 slices. Αντιθέτως, η προτεινόμενη αρχιτεκτονική πέτυχε σημαντική μείωση της απαιτούμενης επιφάνειας, περιορίζοντας τη χρήση σε μόλις 868 slices, γεγονός που αναδεικνύει την αποδοτικότητά της σε επίπεδο υλοποίησης.

Στην πλακέτα Virtex-6, το σχέδιο [143] καταλάμβανε την υψηλότερη επιφάνεια σε (slices) των 1432, ενώ το σχέδιο [144] κατανάλωσε τη χαμηλότερη επιφάνεια σε (slices) περίπου 871. Αν και το [144] καταλάμβανε ελαφρώς μικρότερη επιφάνεια σε (slices) από την προτεινόμενη σχεδίαση, είχε ως αποτέλεσμα χαμηλή ρυθμαπόδοση, συχνότητα και αποδοτικότητα, υποδεικνύοντας ότι η χρήση της επιφάνειας σε (slices) του σχεδίου είναι μόνο ένας από τους παράγοντες που πρέπει να ληφθούν υπόψη για την αξιολόγηση της απόδοσής του.

Στην πλακέτα Virtex-7, το προτεινόμενο σχέδιο καταλάμβανε οριακά την υψηλότερη επιφάνεια σε (slices) 1094, ενώ το σχέδιο [183] κατανάλωσε τη χαμηλότερη επιφάνεια σε (slices) περίπου 998. Ωστόσο, παρατηρείται ότι το [183] παρήγαγε χαμηλή συχνότητα, ρυθμαπόδοση και αποδοτικότητα. Έτσι, υποδηλώνει ότι η προτεινόμενη σχεδίαση μπορεί να έχει ελαφρώς μεγαλύτερη χρήση επιφάνειας σε (slices), αλλά εξακολουθεί να είναι πιο αποτελεσματική στην ρυθμαπόδοση σε σύγκριση με τον πόρο σχεδίασης [183].

Στην πλακέτα Artix-7, το σχέδιο [184] καταλαμβάνει σημαντικά μεγαλύτερη επιφάνεια σε (slices), συγκεκριμένα 4188 slices, ενώ ο προτεινόμενος σχεδιασμός καταλαμβάνει μια πολύ χαμηλότερη επιφάνεια σε (slices) περίπου 902. Παρόλο που η σχεδίαση [184] πέτυχε υψηλότερη ρυθμαπόδοση για 512 bit μήκος εξόδου δεν είχε καλή απόδοση, με μέτρο αποδοτικότητας 3,93 Mbps/slices.

Αντίθετα, ο προτεινόμενος σχεδιασμός παρουσίασε υψηλότερη αποδοτικότητα 10,57 Mbps/slices. Αυτό υποδηλώνει ότι ο προτεινόμενος σχεδιασμός είναι πιο αποτελεσματικός σε μετρήσεις επιφάνειας σε (slices) και ρυθμαπόδοσης και δεν είναι τόσο αποτελεσματικός στις μετρήσεις αποδοτικότητας.

Ο προτεινόμενος σχεδιασμός με 24 κύκλους ρολογιού προσφέρει την υψηλότερη ρυθμαπόδοση όταν εφαρμόζεται σε πλακέτες Virtex-5, Virtex-6 και Virtex-7. Εκτός από τα επιτεύγματα ρυθμαπόδοσης, η προτεινόμενη σχεδίαση υπερέχει επίσης στη χαμηλή χρήση της επιφάνειας σε (slices) και στην αποδοτικότητα όταν αναπτύσσεται στην πλακέτα Artix-7.

Στην πλακέτα Virtex-5 με 12 συνολικά κύκλους ρολογιού, η σχεδίαση που αναφέρεται στο [143] καταλάμβανε την υψηλότερη επιφάνεια σε (slices) των 2144, ενώ η προτεινόμενη σχεδίαση κατανάλωσε τη χαμηλότερη επιφάνεια σε (slices) περίπου 1112. Υποδεικνύει ότι ο προτεινόμενος σχεδιασμός είναι πιο αποδοτικός ως προς την επιφάνεια σε (slices) από τον σχεδιασμό που αναφέρεται στο [143].

Στην πλακέτα Virtex-6 με 12 συνολικά κύκλους ρολογιού, η σχεδίαση που αναφέρεται στο [143] καταλάμβανε την υψηλότερη επιφάνεια σε (slices) των 3557, ενώ η προτεινόμενη σχεδίαση κατανάλωσε τη χαμηλότερη επιφάνεια σε (slices) περίπου 1287. Σε αυτήν την περίπτωση, ο προτεινόμενος σχεδιασμός είναι επίσης πιο αποδοτικός ως προς την επιφάνεια σε (slices) από τον σχεδιασμό που αναφέρεται στο [143].

Στην πλακέτα Artix-7, το σχέδιο [184] καταλάμβανε πολύ μεγαλύτερη επιφάνεια σε (slices) 7139, αλλά το προτεινόμενο σχέδιο κατανάλωσε μια σημαντικά χαμηλότερη επιφάνεια σε (slices), περίπου 1184. Η σχεδίαση [184] είχε κακή απόδοση ως προς την αποδοτικότητα, με τιμή αποδοτικότητα 2,80 Mbps/slices, ενώ η σχεδίαση που παρουσιάστηκε παρουσίασε υψηλότερο ρυθμό αποδοτικότητας 10,31 Mbps/slices.

Ο προτεινόμενος σχεδιασμός με 12 κύκλους ρολογιού επιτυγχάνει την υψηλότερη αποδοτικότητα όταν χρησιμοποιείται σε πλακέτες Virtex-5, Virtex-6 και Virtex-7 FPGA. Επιπλέον, παρουσιάζει εξαιρετική ρυθμαπόδοση και χαμηλή χρήση επιφάνειας όταν εφαρμόζεται στην πλακέτα Artix-7. Κατά συνέπεια, η αξιολόγηση του προτεινόμενου αρχιτεκτονικού σχεδίου σε σύγκριση με πρόσφατες μελέτες της σχετικής βιβλιογραφίας αποδεικνύει ότι το συγκεκριμένο σχέδιο υπερέχει τόσο στην ρυθμαπόδοση όσο και στη μείωση της απαιτούμενης επιφάνειας υλικού, αναδεικνύοντας το ως μια ουσιαστική συνεισφορά στον τομέα του σχεδιασμού λειτουργιών κατακερματισμού. Η επιτάχυνση που επιτυγχάνεται

καθιστά την προτεινόμενη αρχιτεκτονική κατάλληλη για εφαρμογές που απαιτούν ταχεία και αποδοτική εκτέλεση κρυπτογραφικών συναρτήσεων κατακερματισμού, συμβάλλοντας στην ανάπτυξη πιο γρήγορων και αποτελεσματικών συστημάτων ασφάλειας δεδομένων.

## 4.7 Συμπεράσματα κεφαλαίου και μελλοντικές εργασίες

Η κρυπτογραφία κατέχει κεντρικό ρόλο στη διασφάλιση της ασφάλειας και της εμπιστευτικότητας των ψηφιακών μέσων, ιδίως στον σημερινό παγκοσμίως διασυνδεδεμένο κόσμο. Η μετάδοση ευαίσθητων πληροφοριών σε ποικίλες μορφές (όπως κείμενο, εικόνα, βίντεο και ήχος) απαιτεί την ύπαρξη ισχυρών κρυπτογραφικών αλγορίθμων που παρέχουν υψηλό επίπεδο ασφάλειας και ανθεκτικότητα απέναντι σε εξελιγμένες επιθέσεις. Παράλληλα, η ανάγκη για επιτάχυνση της κρυπτογραφικής επεξεργασίας γίνεται ολοένα πιο επιτακτική, ώστε να διασφαλίζεται η ταχεία και αποδοτική προστασία των δεδομένων σε πραγματικό χρόνο, χωρίς να θυσιάζεται η ασφάλεια. Η βελτιστοποίηση και η επιτάχυνση των κρυπτογραφικών αλγορίθμων αποτελούν βασικούς παράγοντες για την ανταπόκριση στις αυξανόμενες απαιτήσεις των σύγχρονων εφαρμογών.

Ο αλγόριθμος SHA-3 έχει κερδίσει ευρεία αποδοχή και αναγνώριση λόγω της ισχυρής του ανθεκτικότητας σε επιθέσεις κρυπτανάλυσης, καθώς και του ισορροπημένου συνδυασμού υψηλής ταχύτητας, αποδοτικότητας και ασφάλειας. Η επίσημη υιοθέτησή του από το NIST ως αντικαταστάτη των παλαιότερων αλγορίθμων SHA-1 και SHA-2 αναδεικνύει τη σημασία του ως θεμελιώδους εργαλείου για τη διασφάλιση της ασφάλειας και της ακεραιότητας των ψηφιακών δεδομένων. Παράλληλα, η επιτάχυνση της υλοποίησης του SHA-3 αποτελεί κρίσιμο παράγοντα για την ικανοποίηση των απαιτήσεων των σύγχρονων εφαρμογών υψηλής απόδοσης, καθιστώντας τον αλγόριθμο ιδανικό για εφαρμογές που απαιτούν γρήγορη και ασφαλή επεξεργασία δεδομένων.

Η παρούσα μελέτη εστιάζει στη βελτιστοποίηση της απόδοσης του αλγορίθμου SHA-3 για όλα τα υποστηριζόμενα μήκη εξόδου (224, 256, 384 και 512 bit), με πειραματικές υλοποιήσεις σε πλακέτες Artix-7, Virtex-5, Virtex-6 και Virtex-7 FPGA. Η προτεινόμενη μέθοδος συγκρίνεται με παρόμοια αρχιτεκτονικά σχέδια που έχουν δημοσιευθεί στη διεθνή βιβλιογραφία. Τα αποτελέσματα καταδεικνύουν

ότι η προτεινόμενη αρχιτεκτονική επιτυγχάνει υψηλότερη απόδοση στα κλασικά κριτήρια αξιολόγησης, συμπεριλαμβανομένης της ελάχιστης επιφάνειας σε (slices), της ρυθμαπόδοσης (Gbps) και της βελτιωμένης αποδοτικότητας (Mbps/slices).

Σε μελλοντική εργασία, σκοπεύουμε να εμβαθύνουμε περαιτέρω στη βελτίωση των μετρικών ρυθμαπόδοσης και αποδοτικότητας μέσω της εφαρμογής της στρατηγικής ξετυλίγματος τεσσάρων σταδίων (loop unrolling by 4). Η υιοθέτηση αυτής της προσέγγισης αναμένεται να οδηγήσει σε σημαντική επιτάχυνση της εκτέλεσης, μειώνοντας περαιτέρω τον αριθμό των απαιτούμενων κύκλων ρολογιού και βελτιώνοντας την αποδοτικότητα χρήσης πόρων. Επιπλέον, σχεδιάζουμε να διεξάγουμε πιο εκτενείς και ρεαλιστικές πειραματικές δοκιμές τόσο σε πλακέτες FPGA όσο και σε ολοκληρωμένα συστήματα στο ίδιο τσιπ (SoC), με σκοπό την επικύρωση της αποδοτικότητας και της επεκτασιμότητας της μεθόδου σε πραγματικά περιβάλλοντα εφαρμογής.

# Κεφάλαιο 5

# Τεχνική επιτάχυνσης διοχετεύσεων και ξετυλίγματος υλικού

Στα προηγούμενα κεφάλαια, τρία και τέσσερα, παρουσιάσαμε δύο μεθόδους βελτιστοποίησης του αλγορίθμου SHA-3, η μία βασισμένη στην τεχνική της διασωλήνωσης και η άλλη στην προσέγγιση του ξετυλίγματος. Στο παρόν κεφάλαιο[1], εισάγουμε την τεχνική βελτιστοποίησης που συνδυάζει τις στρατηγικές διασωλήνωσης και ξετυλίγματος, με κύριο στόχο την επιτάχυνση και τη βελτίωση της αποδοτικότητας σε επίπεδο υλικού. Η προτεινόμενη μέθοδος επιδιώκει τη μείωση του αριθμού των κύκλων ρολογιού ανά απαιτούμενη λειτουργία, εξασφαλίζοντας ταυτόχρονα αυξημένη ταχύτητα επεξεργασίας και βελτιωμένη αποδοτικότητα πόρων. Με αυτόν τον τρόπο, η νέα προσέγγιση καθιστά δυνατή την επίτευξη ταχύτερης και πιο αποτελεσματικής επεξεργασίας δεδομένων, ενισχύοντας την απόδοση του SHA-3 σε συσκευές FPGA.

## 5.1 Περίληψη

Οι συναρτήσεις κατακερματισμού συνιστούν θεμελιώδη μηχανισμό στο πεδίο της ασφάλειας πληροφοριών, καθώς διαδραματίζουν καθοριστικό ρόλο στη διασφάλιση της ακεραιότητας, της εμπιστευτικότητας και της αυθεντικότητας των δεδομένων. Η ευρεία χρήση τους καλύπτει κρίσιμες εφαρμογές, όπως η αποθήκευση και η επαλήθευση κωδικών πρόσβασης, η παραγωγή ψευδοτυχαίων ακολουθιών, καθώς

[1]Το κεφάλαιο έχει δημοσιευθεί στο άρθρο [190]

και η εξαγωγή και διαχείριση κρυπτογραφικών κλειδιών. Οι εφαρμογές τους εκτείνονται από το στρατιωτικό και τον κυβερνητικό τομέα έως το ηλεκτρονικό εμπόριο, τις τραπεζικές υπηρεσίες, τη διαχείριση συστημάτων υγειονομικής περίθαλψης και τις συσκευές του Διαδικτύου των Πραγμάτων (Internet of Things - IoT).

Μεταξύ των κρυπτογραφικών αλγορίθμων κατακερματισμού, η συνάρτηση Keccak (πλέον υιοθετημένη ως SHA-3), διακρίνεται για την απόδοση σε επίπεδο υλικού και την ισχυρή ανθεκτικότητά της απέναντι στις σύγχρονες τεχνικές κρυπτανάλυσης, σε σύγκριση με προηγούμενους αλγόριθμους όπως ο SHA-1 και ο SHA-2. Παρά τις ήδη σημαντικές επιδόσεις, η συνεχιζόμενη εξέλιξη των υπολογιστικών απαιτήσεων καθιστά επιτακτική την ανάγκη για περαιτέρω βελτιστοποιήσεις, με στόχο τόσο την αύξηση του ρυθμού διεκπεραίωσης όσο και τη μείωση της απαιτούμενης επιφάνειας επικάλυψης. Τέτοιες βελτιώσεις συμβάλλουν καθοριστικά στην αποτελεσματικότερη υλοποίηση του αλγορίθμου σε περιορισμένα και απαιτητικά υπολογιστικά περιβάλλοντα, όπως οι συσκευές FPGA.

Η παρούσα μελέτη επικεντρώνεται στην επιτάχυνση του ρυθμού διεκπεραίωσης του αλγορίθμου κατακερματισμού SHA-3 μέσω της εισαγωγής μιας νέας αρχιτεκτονικής, η οποία επιτυγχάνει αποδοτικά αποτελέσματα σε υλοποιήσεις υλικού. Η προτεινόμενη αρχιτεκτονική αξιολογήθηκε σε πλακέτες FPGA των σειρών Virtex-5, Virtex-6 και Virtex-7, όπου κατέγραψε ρυθμούς διεκπεραίωσης 26.151 Gbps, 33.084 Gbps και 38.043 Gbps αντίστοιχα. Τα αποτελέσματα αυτά υπογραμμίζουν οτι η προτεινόμενη προσέγγιση είναι ιδιαίτερα κατάλληλη για εφαρμογές υψηλής απόδοσης, όπου η ταχύτητα επεξεργασίας και η αποδοτική αξιοποίηση των πόρων αποτελούν κρίσιμες παραμέτρους.

Επιπλέον, η παρούσα εργασία περιλαμβάνει συγκριτική ανάλυση της προτεινόμενης αρχιτεκτονικής έναντι πρόσφατων μεθόδων, αποδεικνύοντας βελτίωση στον ρυθμό διεκπεραίωσης πάνω από 11,37% στην Virtex-5, 10,49% στην Virtex-6 και 11,47% στην Virtex-7. Αυτή η σύγκριση παρέχει ένα ολοκληρωμένο πλαίσιο αξιολόγησης, αναδεικνύοντας τη βελτιωμένη ρυθμαπόδοση και αποδοτικότητα της νέας αρχιτεκτονικής σε σύγκριση με τις πιο σύγχρονες και αναγνωρισμένες υλοποιήσεις στον τομέα.

## 5.2 Εισαγωγή

Οι συναρτήσεις κατακερματισμού διαδραματίζουν ζωτικό ρόλο στον τομέα της ασφάλειας πληροφοριών, χρησιμεύοντας ως θεμελιώδη εργαλεία σε διάφορες εφαρμογές. Ένας από τους πρωταρχικούς σκοπούς τους είναι να επιτύχουν την ασφάλεια των πληροφοριών, παρέχοντας υπηρεσίες όπως η αυθεντικοποίηση ταυτότητας και η ακεραιότητα. Στο πλαίσιο της αποθήκευσης και της επαλήθευσης κωδικού πρόσβασης, οι λειτουργίες κατακερματισμού χρησιμοποιούνται εκτενώς για τη μετατροπή των κωδικών πρόσβασης σε μη αναστρέψιμους κατακερματισμούς, προστατεύοντας τους αρχικούς κωδικούς πρόσβασης από μη εξουσιοδοτημένες ενέργειες [191, 192]. Επιπλέον, αυτές οι συναρτήσεις βρίσκουν εφαρμογή στη δημιουργία ψευδοτυχαίων ακολουθιών, οι οποίες είναι απαραίτητες στην κρυπτογραφία και την κρίσιμη παραγωγή για διάφορους σκοπούς, όπως ο στρατιωτικός, το διαδικτυακό εμπόριο, η διαχείριση υγειονομικής περίθαλψης, οι τραπεζικές συναλλαγές και το Διαδίκτυο των Πραγμάτων (IoT) [193–195].

Ωστόσο, αξίζει να σημειωθεί ότι αρκετοί ευρέως χρησιμοποιούμενοι αλγόριθμοι κατακερματισμού, συμπεριλαμβανομένων των SHA-1, Snefru, MD4, MD5, RIPEMD και HAVAL, έχει επιβεβαιωθεί ότι είναι ευάλωτοι σε επιθέσεις σύγκρουσης. Οι επιθέσεις σύγκρουσης συμβαίνουν όταν δύο διαφορετικές είσοδοι παράγουν την ίδια έξοδο κατακερματισμού, θέτοντας σε κίνδυνο την ασφάλεια της συνάρτησης κατακερματισμού [164, 196, 197]. Το Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας (NIST) έλαβε προληπτικά μέτρα για την αντιμετώπιση αυτού του ζητήματος και την ενίσχυση της ασφάλειας των πληροφοριών. Οργάνωσαν έναν διαγωνισμό τριών γύρων για να αντικαταστήσουν το πρότυπο κατακερματισμού SHA-2, το οποίο θεωρούνταν ασφαλές μέχρι εκείνη την εποχή. Σκοπός του διαγωνισμού ήταν να εντοπίσει ένα νέο πρότυπο κατακερματισμού που θα μπορούσε να αντέξει σε επιθέσεις σύγκρουσης και να παρέχει ισχυρότερη ασφάλεια [69, 198, 199].

Ο διαγωνισμός του οργανισμό NIST ολοκληρώθηκε τελικά με την επιλογή και την υιοθέτηση της συνάρτησης Keccak ως το νέο πρότυπο κατακερματισμού. Ο αλγόριθμος Keccak, και πλέον το νέο πρότυπο SHA-3, έχει σχεδιαστεί για να αντιστέκεται σε επιθέσεις σύγκρουσης και άλλες κρυπτογραφικές ευπάθειες. Η επιλογή του ως το νέο πρότυπο υποδηλώνει τη σημασία της συνεχούς προώθησης των λειτουργιών κατακερματισμού για την κάλυψη των εξελισσόμενων απαιτήσεων ασφαλείας και την προστασία ευαίσθητων πληροφοριών στο σημερινό διασυνδεδεμένο ψηφιακό τοπίο [200]. Αν υιοθετήσουμε πιο ασφαλείς αλγόριθμους

κατακερματισμού όπως ο SHA-3, οι οργανισμοί και τα άτομα αυτών μπορούν να βελτιώσουν τη συνολική ασφάλεια των συστημάτων και των δεδομένων τους, μειώνοντας τον κίνδυνο μη εξουσιοδοτημένης πρόσβασης και κακόβουλων δραστηριοτήτων [201]. Ο αλγόριθμος SHA-3 είναι μια ευρέως χρησιμοποιούμενη κρυπτογραφική συνάρτηση κατακερματισμού που διαδραματίζει κρίσιμο ρόλο στη διασφάλιση του απορρήτου των δεδομένων και στη διατήρηση της ακεραιότητας της ανταλλαγής δεδομένων σε διάφορα συστήματα. Ωστόσο, υπάρχει μια συνεχής πρόκληση για τη βελτίωση της απόδοσης της υλοποίησης του αλγόριθμου SHA-3, ιδιαίτερα στο πλαίσιο των ενσωματωμένων συστημάτων [202, 203].

Ένα πλεονέκτημα της εφαρμογής του SHA-3 σε FPGAs είναι η ανώτερη ταχύτητά του σε σύγκριση με προηγούμενους αλγόριθμους SHA όταν εφαρμόζονται σε υλικό. Ο SHA-3 έχει σχεδιαστεί για να προσφέρει εξαιρετική απόδοση σε διάφορες πλατφόρμες υλικού. Η χρήση FPGA για την υλοποίηση του SHA-3 προσφέρει τα πλεονεκτήματα της προσαρμογής αλγορίθμων και της ευελιξίας στην επαναδιαμόρφωση. Τα FPGA μπορούν να προσαρμοστούν ώστε να καταναλώνουν λιγότερη ενέργεια από τους παραδοσιακούς επεξεργαστές, καθιστώντας τα ιδανικά για την υλοποίηση κρυπτογραφικών λειτουργιών όπως ο SHA-3. Επιπλέον, τα FPGA μπορούν να βελτιώσουν σημαντικά την απόδοση των υπολογισμών του SHA-3 [139, 177].

Ως αποτέλεσμα αυτών των πλεονεκτημάτων, έχουν προταθεί διάφορες στρατηγικές για την αποτελεσματική εφαρμογή του αλγόριθμου SHA-3. Αυτές οι προσεγγίσεις επικεντρώνονται στη μείωση της κατανάλωσης ενέργειας, στη μεγιστοποίηση της χρήσης των πόρων της επιφάνειας επικάλυψης σε (slices) ή στη βελτίωση της ταχύτητας επεξεργασίας [204]. Ερευνητές και προγραμματιστές διερευνούν καινοτόμες τεχνικές για τη βελτιστοποίηση της εφαρμογής του SHA-3 σε ενσωματωμένα συστήματα, αξιοποιώντας τις δυνατότητες των πλακετών FPGA για να επιτύχουν βελτιωμένη ρυθμαπόδοση, ασφάλεια και αποδοτικότητα [141].

Αντιμετωπίζοντας τις προκλήσεις που σχετίζονται με την υλοποίηση του SHA-3, ιδιαίτερα στον τομέα των ενσωματωμένων συστημάτων, καθίσταται εφικτή η πλήρης αξιοποίηση των δυνατοτήτων αυτού του αλγορίθμου για τη διασφάλιση του απορρήτου των δεδομένων και τη διατήρηση της ακεραιότητας της ανταλλαγής δεδομένων σε διάφορες εφαρμογές. Οι συνεχείς εξελίξεις στην τεχνολογία των FPGAs και οι συνεχείς ερευνητικές προσπάθειες στη βελτιστοποίηση των αλγορίθμων συμβάλλουν στην εξέλιξη και την ευρύτερη υιοθέτηση του SHA-3 στην ασφάλεια ευαίσθητων πληροφοριών σε καθημερινά συστήματα [205, 206].

Ακολουθεί μια σύντομη περίληψη των διαφόρων συνεισφορών που έγιναν σε αυτό το κεφάλαιο:

- Προτείνουμε μια νέα στρατηγική βελτιστοποίησης της αποτελεσματικότητας του αλγορίθμου SHA-3. Η προσέγγισή μας βασίζεται στις τεχνικές του ξετυλίγματος και της διασωλήνωσης, οι οποίες είναι καθιερωμένες μέθοδοι στις συσκευές FPGA. Ο πρωταρχικός στόχος αυτής της εργασίας είναι να βελτιώσουμε την επιτάχυνση του SHA-3 όσον αφορά τη ρυθμαπόδοση, τη συχνότητα και τη χρήση της επιφάνειας σε (slices) (Ενότητα 5.4.3).
- Για να διασφαλίσουμε την ακρίβεια και την αξιοπιστία της προτεινόμενης μεθόδου μας, πραγματοποιήσαμε μια διεξοδική διαδικασία επικύρωσης χρησιμοποιώντας καθιερωμένα δείγματα που παρέχονται από τον οργανισμό NIST. Αυτό το βήμα επικύρωσης είναι ζωτικής σημασίας για να διασφαλίσουμε ότι η προτεινόμενη στρατηγική μας διατηρεί τις απαραίτητες κρυπτογραφικές ιδιότητες και συμμορφώνεται με τις προδιαγραφές που περιγράφονται από τον οργανισμό NIST για τον αλγόριθμο SHA-3 (Ενότητα 5.5.1).
- Τέλος, πραγματοποιήσαμε μια εκτενή αξιολόγηση και ανάλυση της προτεινόμενης αρχιτεκτονικής μας, συγκρίνοντάς την με άλλες παρόμοιες μεθόδους που περιγράφονται στην πρόσφατη δημοσιευμένη βιβλιογραφία. Η αξιολόγηση επικεντρώθηκε σε βασικές μετρήσεις απόδοσης, συμπεριλαμβανομένης της χρήσης επιφάνειας επικάλυψης σε (slices), της ρυθμαπόδοσης σε Gbps, της συχνότητας σε MHz και της αποδοτικότητας σε Mbps/slice. Αξιοποιώντας αυτές τις αξιολογήσεις και συγκρίσεις, μπορούμε με σιγουριά να επιβεβαιώσουμε την ανωτερότητα του σχεδιασμού μας όσον αφορά την ρυθμαπόδοση, την αποδοτικότητα και τη χρήση της επιφάνειας επικάλυψης σε (slices) (Ενότητα 5.5.3).

Η υπόλοιπη μελέτη οργανώνεται ως εξής: Στην επόμενη Ενότητα 5.3, παρουσιάζουμε τις σχετικές πιο σύγχρονες μελέτες στη βιβλιογραφία. Στην ενότητα 5.4 περιγράφουμε τη νέα προτεινόμενη στρατηγική βελτιστοποίησης σε υλικό του αλγόριθμου SHA-3 σε πλακέτές FPGA. Στην ενότητα 5.5, παρουσιάζουμε τα πειραματικά αποτελέσματα της εργασίας μας και τις συγκρίσεις με άλλες εργασίες. Στην ενότητα 5.6, συζητάμε τη νέα προτεινόμενη στρατηγική βελτιστοποίησης. Τέλος, στην Ενότητα 5.7 συνοψίζουμε τα ευρήματα της μελέτης μας και αναφέρουμε πιθανές μελλοντικές εργασίες.

## 5.3 Σχετικές εργασίες διοχετεύσεων και ξετυλίγματος υλικού

Οι ερευνητικές κοινότητες στον τομέα της κρυπτογραφίας διεξάγουν συνεχώς εκτεταμένες μελέτες με στόχο τη βελτιστοποίηση αρχιτεκτονικών και μεθοδολογιών για την αποδοτική υλοποίηση του αλγορίθμου SHA-3 σε συσκευές FPGA. Ο πρωταρχικός σκοπός αυτών των προσεγγίσεων είναι η ενίσχυση της απόδοσης του αλγορίθμου σε επίπεδο υλικού, με παράλληλη μείωση της κατανάλωσης των απαιτούμενων πόρων, επιτυγχάνοντας έτσι μία ισορροπία μεταξύ της υψηλής ρυθμαπόδοσης και της αποδοτικής χρήσης επιφάνειας [136, 141, 188, 207, 208]. Σε αυτή την ενότητα θα εξετάσουμε άλλες παρόμοιες εργασίες και θα συζητήσουμε λεπτομερώς τα ευρήματά τους.

Η μελέτη των [153] προτείνει μια νέα τεχνική για την υλοποίηση του αλγόριθμου SHA-3 με μέγεθος εξόδου 512 bit. Οι συγγραφείς πραγματοποίησαν μια αξιολόγηση του προτεινόμενου σχεδιασμού σε πλακέτα Virtex-5. Σε αυτήν την υλοποίηση, για μέγεθος εξόδου 512 bit, στην Virtex-5 FPGA απαιτούνται 1680 (slices) και λειτουργεί με συχνότητα ρολογιού 387 MHz. Αυτός ο συγκεκριμένος σχεδιασμός πέτυχε ρυθμαπόδοση 8,06 Gbps. Επιπλέον, η ρυθμαπόδοση αυτού του σχεδιασμού μετρήθηκε και βρέθηκε ότι είναι 4,91 Mbps/Slice.

Στο [186], οι συγγραφείς πρότειναν μια νέα προσέγγιση για την εφαρμογή της συνάρτησης SHA-3 με μέγεθος εξόδου 512 bit. Οι συγγραφείς αξιολόγησαν την απόδοση του προτεινόμενου σχεδιασμού στη συσκευή Virtex-7 FPGA. Σε αυτήν την υλοποίηση, για μέγεθος εξόδου 512 bit, το Virtex-7 FPGA χρειάζεται 1454 (slices) και λειτουργεί με συχνότητα ρολογιού 374,035 MHz. Αυτή η διαμόρφωση πέτυχε ρυθμαπόδοση 7,979 Gbps και αποδοτικότητα 5,49 Mbps/Slice.

Οι Rao et al. [209] πρότειναν μια νέα μέθοδο για την υλοποίηση του SHA-3 στις πλακέτες Virtex-5 και Virtex-6. Επικεντρώθηκαν σε πειραματικές εφαρμογές με μεγέθη εξόδου των 256 και 512 bit. Για την αρχιτεκτονική του SHA-3, κατά την υλοποίηση του στην πλακέτα Virtex-5 με μέγεθος εξόδου 256 bit, ο σχεδιασμός χρησιμοποίησε επιφάνεια επικάλυψης 1291 (slices) και λειτουργεί με συχνότητα ρολογιού 377,86 MHz. Αυτή η διαμόρφωση πέτυχε ρυθμαπόδοση 17,132 Gbps. Από την άλλη πλευρά, με μέγεθος εξόδου 512 bit, στην πλακέτα Virtex-5 χρησιμοποίθηκε επιφάνεια επικάλυψης 1409 slices και πέτυχε ρυθμαπόδοση 10,19 Gbps. Στην περίπτωση της αρχιτεκτονικής στην πλακέτα Virtex-6, η προτεινόμενη υλοποίηση

του SHA-3 με μέγεθος εξόδου 256 bit απαιτούσε επιφάνεια επικάλυψης 1028 slices και λειτουργεί με συχνότητα ρολογιού 424,44 MHz. Αυτό είχε ως αποτέλεσμα υψηλότερη ρυθμαπόδοση 19,241 Gbps από την υλοποίηση του στην πλακέτα Virtex-5. Η υλοποίηση σε πλακέτα Virtex-6 με μέγεθος εξόδου 512 bit χρησιμοποίησε επιφάνεια επικάλυψης 1227 slices και πέτυχε ρυθμαπόδοση 8,22 Gbps.

Στο [210], οι συγγραφείς προτείνουν ένα νέο σχέδιο για την υλοποίηση της αρχιτεκτονικής SHA-3 με μέγεθος εξόδου 512 bit. Αυτός ο σχεδιασμός προσφέρει μια αντιστάθμιση μεταξύ της μέγιστης συχνότητας και της επιφάνειας επικάλυψης, επιτρέποντας ευελιξία στη βελτιστοποίηση της απόδοσης και στη χρήση των πόρων. Οι συγγραφείς αξιολόγησαν την απόδοση του προτεινόμενου σχεδιασμού σε διαφορετικές συσκευές FPGA, εστιάζοντας συγκεκριμένα στις Virtex-5, Virtex-6 και Virtex-7. Για μέγεθος εξόδου 512 bit, η αρχιτεκτονική στην πλακέτα Virtex-5 απαιτεί 1388 slices και λειτουργεί με συχνότητα ρολογιού 287,39 MHz. Αυτή η διαμόρφωση πέτυχε ρυθμαπόδοση 11,50 Gbps. Προχωρώντας στην αρχιτεκτονική στην Virtex-6, ο προτεινόμενος σχεδιασμός απαιτεί 1167 slices και λειτουργεί με υψηλότερη συχνότητα ρολογιού 394,01 MHz. Αυτή η αυξημένη συχνότητα ρολογιού βελτίωσε την ρυθμαπόδοση κατά 15,76 Gbps. Τέλος, κατά την εξέταση της αρχιτεκτονικής στην Virtex-7, ο σχεδιασμός χρησιμοποιεί 1418 slices και λειτουργεί με συχνότητα ρολογιού 414,54 MHz. Αυτή η διαμόρφωση πέτυχε την υψηλότερη ρυθμαπόδοση μεταξύ της αρχιτεκτονικής που αξιολογήθηκε, φτάνοντας τα 16,58 Gbps.

Οι συγγραφείς του [149] εισάγησαν μια διαφορετική προσέγγιση σχεδίασης για την υλοποίηση του SHA-3 με μέγεθος εξόδου 512 bit. Οι συγγραφείς πραγματοποίησαν αξιολόγηση του προτεινόμενου σχεδιασμού στα Virtex-5 και Virtex-6 FPGA. Σε αυτό το σχέδιο, στο Virtex-6 FPGA απαιτεί 2296 slices και λειτουργεί με συχνότητα ρολογιού 391 MHz. Αυτός ο συγκεκριμένος σχεδιασμός πέτυχε ρυθμαπόδοση 9,38 Gbps. Επιπλέον, η αποδοτικότητα αυτού του σχεδιασμού μετρήθηκε και βρέθηκε ότι είναι 8,17 Mbps/Slice.

Στο [146], οι συγγραφείς πρότειναν μια νέα προσέγγιση για την εφαρμογή του αλγόριθμου SHA-3 με μέγεθος εξόδου 512 bit. Οι συγγραφείς αξιολόγησαν την απόδοση του προτεινόμενου σχεδίου στο Virtex-5 FPGA. Σε αυτήν την υλοποίηση, για μέγεθος εξόδου 512 bit, το Virtex-5 FPGA χρειάζεται 2326 slices και λειτουργεί με συχνότητα ρολογιού 306 MHz. Αυτός ο σχεδιασμός πέτυχε ρυθμαπόδοση 5,56 Gbps και αποδοτικότητα 2,40 Mbps/Slice.

Η μελέτη των [148] παρουσιάζει μια νέα σχεδίαση για την υλοποίηση του αλγορίθμου SHA-3 με μέγεθος εξόδου 512 bit. Σε αυτήν την υλοποίηση, στο Virtex-5 FPGA απαιτούνται 1163 slices και λειτουργεί με συχνότητα ρολογιού 273 MHz. Αυτός ο σχεδιασμός πέτυχε μετρήσεις ρυθμαπόδοσης 7,80 Gbps. Επίσης, η αποδοτικότητα αυτού του σχεδιασμού έφτασε τα 6,06 Mbps/Slice.

Ο Πίνακας 5.1 περιλαμβάνει συγκεντωτικά τις υλοποιήσεις που δημοσιεύτηκαν πρόσφατα του αλγόριθμου SHA-3. Οι περισσότερες προηγούμενες εργασίες με τον αλγόριθμο SHA-3 επικεντρώνονται κυρίως στη χρήση της κλασικής γεννήτριας RC 64-bit. Ωστόσο, η συγκεκριμένη εργασία στοχεύει να βελτιώσει αυτές τις υπάρχουσες προσεγγίσεις με την εισαγωγή της βελτιστοποιημένης γεννήτριας RC που μειώνει σημαντικά το μέγεθός της.

Ο πρωταρχικός στόχος αυτής της μελέτης είναι να βελτιώσει και να συγκρίνει τις μετρικές απόδοσης, συγκεκριμένα την αποδοτικότητα και την ρυθμαπόδοση, με τη βελτιωμένη γεννήτρια RC που ενσωματώνεται στον προτεινόμενο αλγόριθμο SHA-3. Με τη μείωση του μεγέθους της γεννήτριας RC, η προτεινόμενη τεχνική βελτιστοποίησης στοχεύει στην επίτευξη ανώτερων αποτελεσμάτων απόδοσης σε σύγκριση με προηγούμενες έρευνες.

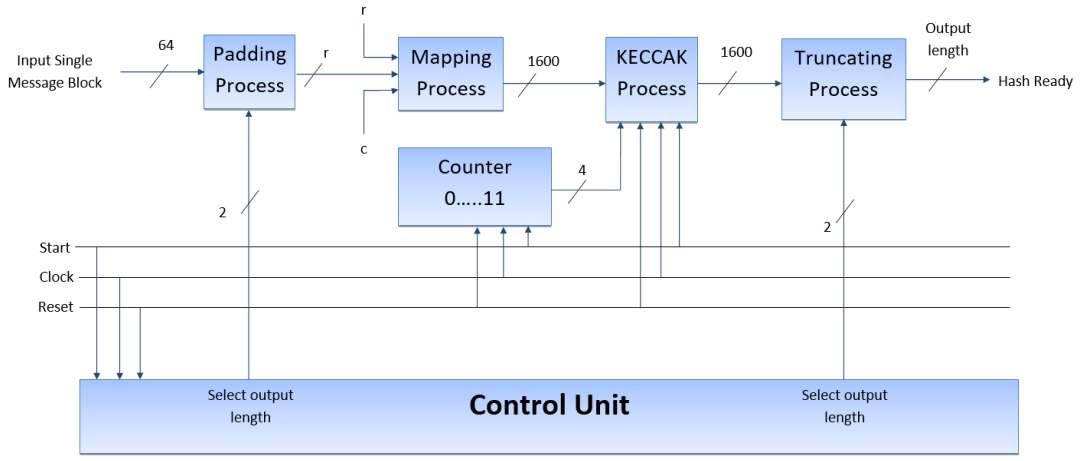
Τα ευρήματα αυτής της μελέτης καταδεικνύουν ότι η προτεινόμενη τεχνική βελτιστοποίησης ξεπερνά τα μέτρα απόδοσης που έχουν επιτευχθεί από προηγούμενες προσεγγίσεις. Τα αποτελέσματα αυτής της μελέτης υποδηλώνουν ότι η βελτιστοποιημένη γεννήτρια RC μπορεί να επιταχύνει αποτελεσματικά τον αλγόριθμο SHA-3.

Πίνακας 5.1: Περίγραμμα με πρόσφατες δημοσιεύσεις για τον αλγόριθμο SHA-3.

| **Μελέτη** | **Μέγεθος Εξόδου** | **Γεννήτρια RC** | **FPGA** |
|---|---|---|---|
| [153] | 512 | 64 | Virtex-5 |
| [186] | 512 | 64 | Virtex-7 |
| [209] | 512 | 64 | Virtex-5 και Virtex-6 |
| [210] | 512 | 64 | Virtex-5, Virtex-6, και Virtex-7 |
| [149] | 512 | 64 | Virtex-5 καιVirtex-6 |
| [146] | 512 | 64 | Virtex-5 |
| [148] | 512 | 64 | Virtex-5 |

## 5.4 Νέα στρατηγική βελτιστοποίησης υλικού

Η προτεινόμενη αρχιτεκτονική συνδυάζει τα πλεονεκτήματα των τεχνικών ξετυλίγματος και διασωλήνωσης, επιτυγχάνοντας ουσιαστικές βελτιώσεις ως προς την αποδοτικότητα και τη ρυθμαπόδοση. Συγκεκριμένα, η αξιοποίηση της δυνατότητας παραλληλοποίησης που προσφέρει το ξετύλιγμα, σε συνδυασμό με τη βελτιωμένη συχνότητα λειτουργίας που επιτυγχάνεται μέσω της διασωλήνωσης, καθιστά εφικτή τη βελτιστοποίηση του αλγορίθμου SHA-3. Το αποτέλεσμα είναι η επίτευξη αυξημένης απόδοσης, υψηλότερης συχνότητας λειτουργίας και αποτελεσματικότερης χρήσης της επιφάνειας υλοποίησης σε slices. Το Σχήμα 5.1 απεικονίζει την αρχιτεκτονική συστήματος της προτεινόμενης στρατηγικής βελτιστοποίησης.




Σχήμα 5.1: Επισκόπηση της προτεινόμενης προσέγγισης της αρχιτεκτονικής.

### 5.4.1 Διαδικασία πλήρωσης

Η κρυπτογραφική συνάρτηση κατακερματισμού έχει σχεδιαστεί ώστε να επεξεργάζεται μηνύματα αυθαίρετου μήκους. Ωστόσο, η εσωτερική συνάρτηση μετάθεσης που ενσωματώνεται στον αλγόριθμο απαιτεί σταθερό μέγεθος εισόδου, το οποίο συμβολίζεται με $r$, και καθορίζει τον όγκο των δεδομένων που υφίστανται επεξεργασία σε κάθε βήμα. Η αναντιστοιχία μεταξύ του μεταβλητού μήκους του μηνύματος εισόδου και της σταθερού μεγέθους μετάθεσης δημιουργεί την ανάγκη εισαγωγής κατάλληλης διαδικασίας πλήρωσης, προκειμένου να διασφαλιστεί η ορθή και συνεπής λειτουργία του αλγορίθμου.

Η τεχνική padding εφαρμόζεται στο αρχικό μήνυμα με σκοπό τη δημιουργία ενός συμπληρωμένου μηνύματος, το οποίο ευθυγραμμίζεται με το απαιτούμενο μέγεθος μπλοκ για επεξεργασία. Συγκεκριμένα, παράγεται ένα μήνυμα μεγέθους $w \times r$, όπου το $w$ είναι ακέραιος, μέσω της προσθήκης κατάλληλης ακολουθίας bits στο τέλος του αρχικού μηνύματος. Η φύση και η διάταξη των επιπρόσθετων bits εξαρτώνται από το επιλεγμένο σχήμα πλήρωσης, το οποίο διασφαλίζει την ορθή στοίχιση των δεδομένων με τις εσωτερικές παραμέτρους του αλγορίθμου.

Η επιλογή του μεγέθους μπλοκ $r$ καθορίζεται από το επιθυμητό μέγεθος της παραγόμενης σύνοψης, με αποτέλεσμα διαφορετικές παραλλαγές της συνάρτησης κατακερματισμού SHA-3, όπως οι SHA-3-224, SHA-3-256, SHA-3-384 και SHA-3-512, να υιοθετούν διαφορετικούς συνδυασμούς τιμών για το $r$ και το μήκος εξόδου. Ο Πίνακας 5.2 συνοψίζει τις αντίστοιχες τιμές των παραμέτρων $r$ (ρυθμός) και $c$ (χωρητικότητα) για κάθε παραλλαγή, παρέχοντας μια ολοκληρωμένη επισκόπηση της εσωτερικής τους διαμόρφωσης.

Πίνακας 5.2: Τα μήκη εξόδου του SHA-3 και οι παράμετροι (r,c).

| **Επιθυμητή έξοδος** | **Μέγεθος μπλοκ** $r$ | **Χωρητικότητα** $c$ |
|---|---|---|
| $SHA-3-224$ | 1152 | 448 |
| $SHA-3-256$ | 1088 | 512 |
| $SHA-3-384$ | 832 | 768 |
| $SHA-3-512$ | 576 | 1024 |

Πέραν της εξασφάλισης της συμβατότητας μεταξύ του μηνύματος εισόδου και της εσωτερικής μετάθεσης, η διαδικασία πλήρωσης συμβάλλει καθοριστικά και στην ενίσχυση της ασφάλειας του αλγορίθμου SHA-3 έναντι επιθέσεων επέκτασης μήκους. Οι εν λόγω επιθέσεις εκμεταλλεύονται ευπάθειες που παρατηρούνται σε ορισμένες συναρτήσεις κατακερματισμού, επιτρέποντας σε έναν αντίπαλο να επεκτείνει το μήνυμα που έχει ήδη κατακερματιστεί και να παράγει νέα έγκυρη τιμή κατακερματισμού, χωρίς να διαθέτει γνώση του αρχικού μηνύματος εισόδου. Η ενσωμάτωση κατάλληλων μηχανισμών πλήρωσης αποτρέπει αυτή τη μορφή εκμετάλλευσης, διασφαλίζοντας την ανθεκτικότητα του SHA-3 απέναντι σε τέτοιες επιθέσεις.

### 5.4.2 Διαδικασία χαρτογράφησης

Η διαδικασία χαρτογράφησης στη φάση προεπεξεργασίας του αλγόριθμου SHA-3 στοχεύει στη δημιουργία δεδομένων εισόδου σε τρεις διαστάσεις, προκειμένου να καταστεί δυνατή η εσωτερική αναπαράσταση της κατάστασης (state array). Αυτό επιτυγχάνεται χρησιμοποιώντας την ακόλουθη Εξίσωση (5.1):

$$\text{State}[x, y, z] = [((\text{Data supplement } r \text{ XOR } r)||c)]^* [(z + 64^*(5^*y + x))] \quad (5.1)$$

Η εξίσωση αντιστοιχίζει τις τρισδιάστατες συντεταγμένες $(x, y, z)$ σε έναν γραμμικό δείκτη εντός της δομής κατάστασης, με σκοπό τη διαμόρφωση των δεδομένων εισόδου σε τρεις διαστάσεις. Η κατάσταση του αλγορίθμου έχει μέγεθος $5 \times 5 \times 64$, όπου οι δείκτες $x$ και $y$ λαμβάνουν τιμές στο διάστημα $[0, 4]$, ενώ ο δείκτης $z$ κυμαίνεται στο διάστημα $[0, 63]$.

Με την αντικατάσταση των επιμέρους τιμών των $x$, $y$ και $z$ στην εξίσωση, καθίσταται δυνατός ο υπολογισμός της αντίστοιχης τιμής κατάστασης $(x, y, z)$. Η διαδικασία αυτή εξασφαλίζει την ορθή απεικόνιση των δεδομένων στο τρισδιάστατο πλέγμα της εσωτερικής κατάστασης, επιτρέποντας την κατάλληλη προετοιμασία τους για τις επόμενες φάσεις επεξεργασίας του αλγορίθμου SHA-3, όπως οι λειτουργίες διάχυσης και μη-γραμμικού μετασχηματισμού.

### 5.4.3 Διαδικασία SHA-3 - στρατηγική βελτιστοποίησης

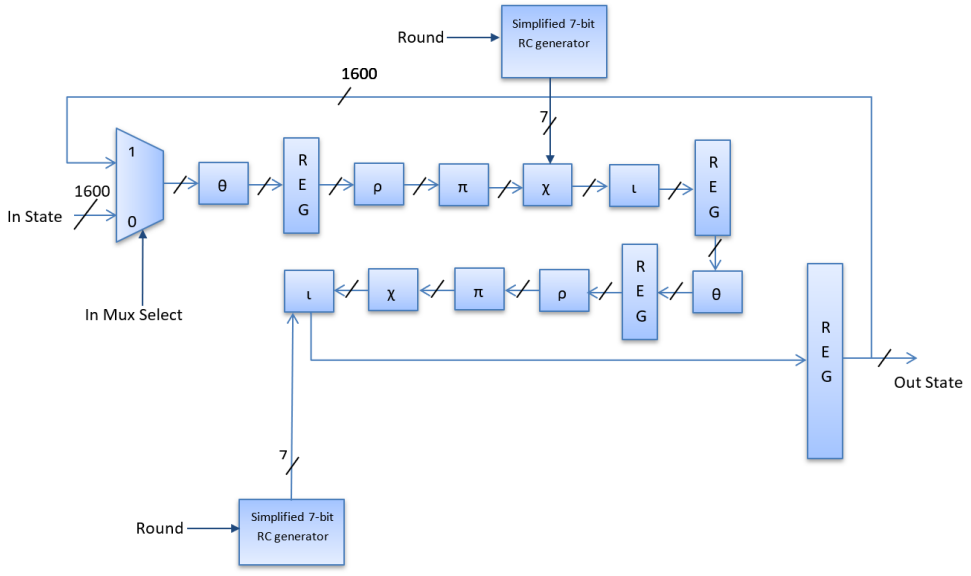
Στην προτεινόμενη βελτιστοποίηση, όπως απεικονίζεται στο Σχήμα 5.2, υιοθετήθηκε μια προσέγγιση διασωλήνωσης δύο σταδίων εντός του μπλοκ μετάθεσης $f$ της συνάρτησης κατακερματισμού SHA-3. Παράλληλα, η συνολική συνάρτηση κατακερματισμού ξετυλίχθηκε κατά δύο επαναλήψεις, ενώ εισήχθησαν δύο αγωγοί μεταξύ των διαδοχικών γύρων, με σκοπό την αύξηση του βαθμού παραλληλίας.

Η διασωλήνωση δύο σταδίων κατανέμει τον υπολογισμό μεταξύ του βήματος $\theta$ και των υπόλοιπων τεσσάρων βημάτων ($\rho$, $\pi$, $\chi$ και $\iota$) που συγκροτούν το μπλοκ μετάθεσης $f$. Η συγκεκριμένη διάσπαση οδηγεί σε πιο αποδοτική εκτέλεση των επιμέρους λειτουργιών, μειώνοντας την κρίσιμη διαδρομή του κυκλώματος και καθιστώντας δυνατή την επίτευξη υψηλότερης συχνότητας ρολογιού. Ως

αποτέλεσμα, βελτιώνεται σημαντικά η συνολική ρυθμαπόδοση της αρχιτεκτονικής υλοποίησης.

Στο πρώτο τμήμα του υπολογισμού, που περιλαμβάνει το βήμα $\theta$, η μεγαλύτερη καθυστέρηση περιλαμβάνει πέντε λειτουργίες XOR. Από την άλλη πλευρά, το δεύτερο τμήμα, που καλύπτει τα βήματα $\pi$ έως $\iota$, επιφέρει την πιο εκτεταμένη καθυστέρηση δύο λειτουργιών XOR, μίας λειτουργίας AND και μίας πρόσθετης λειτουργίας XOR.

Εφαρμόζοντας αυτήν την προσέγγιση διασωλήνωσης και βελτιστοποιώντας την κρίσιμη διαδρομή, μπορούμε να μειώσουμε σημαντικά τη συνολική καθυστέρηση και να βελτιώσουμε τη συχνότητα ρολογιού στην οποία μπορεί να λειτουργήσει η συνάρτηση κατακερματισμού. Αυτή η βελτίωση οδηγεί σε μια πιο αποτελεσματική και υψηλής απόδοσης υλοποίηση της συνάρτησης κατακερματισμού SHA-3 στην προτεινόμενη αρχιτεκτονική μας.




Σχήμα 5.2: Η προτεινόμενη βελτιστοποίηση με τεχνικές ξετυλίγματος και διασωλήνωσης.

Σε αυτήν την εργασία, έχουμε βελτιώσει τη γεννήτρια RC στον αλγόριθμο SHA-3 μειώνοντας σημαντικά το μέγεθός της. Προηγουμένως, η γεννήτρια RC αποθήκευε 24 προ-υπολογισμένες σταθερές, η καθεμία με ένα ορισμένο μήκος, όπως εμφανίζεται στον Πίνακα 5.3.

Η μείωση μεγέθους επιτυγχάνεται με την αποθήκευση μόνο των μη μηδενικών bits σε κάθε τιμή RC, όπως φαίνεται στην Εξίσωση (5.2).

Πίνακας 5.3: Οι τυπικές τιμές RC 64-bit.

| | | | | | |
|---|---|---|---|---|---|
| $RC_0$ | 0000000000000001 | $RC_1$ | 0000000000008082 | $RC_2$ | 800000000000808A |
| $RC_3$ | 8000000080008000 | $RC_4$ | 000000000000808B | $RC_5$ | 0000000080000001 |
| $RC_6$ | 8000000080008081 | $RC_7$ | 8000000000008009 | $RC_8$ | 000000000000008A |
| $RC_9$ | 0000000000000088 | $RC_{10}$ | 0000000080008009 | $RC_{11}$ | 000000008000000A |
| $RC_{12}$ | 000000008000808B | $RC_{13}$ | 800000000000008B | $RC_{14}$ | 8000000000008089 |
| $RC_{15}$ | 8000000000008003 | $RC_{16}$ | 8000000000008002 | $RC_{17}$ | 8000000000000080 |
| $RC_{18}$ | 000000000000800A | $RC_{19}$ | 800000008000000A | $RC_{20}$ | 8000000080008081 |
| $RC_{21}$ | 8000000000008080 | $RC_{22}$ | 0000000080000001 | $RC_{23}$ | 8000000080008008 |

$$A'[x, y, z] = A[x, y, z] \text{ XOR } RC\,[i_g] \tag{5.2}$$

Σύμφωνα με τις προδιαγραφές του αλγόριθμου SHA-3, όπως εμφανίζεται στην Εξίσωση (5.3), κάθε τιμή RC έχει το πολύ επτά μη μηδενικά bits.

$$\mathrm{RC}\,[i_{\mathrm{g}}]\,[0][0]\left[2^h - 1\right] = \mathrm{gc}\,[h + 7i_{\mathrm{g}}] \text{ for all } 0 \leq g \leq n \tag{5.3}$$

Αυτές οι Εξισώσεις (5.2) και (5.3) υπογραμμίζουν τον τρόπο με τον οποίο το μέγεθος της γεννήτριας RC μειώνεται μέσω της διατήρησης μη μηδενικών bit εντός των τιμών RC. Έτσι, εκμεταλλευτήκαμε αυτήν την παρατήρηση και απλοποιήσαμε τις τιμές στις στρογγυλές σταθερές αναλόγως. Οι απλουστευμένες τιμές της γεννήτριας RC των επτά bit παρουσιάζονται στον Πίνακα 5.4.

Πίνακας 5.4: Οι απλοποιημένες τιμές της γεννήτριας $RC_7$.

| | | | | | |
|---|---|---|---|---|---|
| $RC_0$ | 1000000 | $RC_1$ | 0101100 | $RC_2$ | 0111101 |
| $RC_3$ | 0000111 | $RC_4$ | 1111100 | $RC_5$ | 1000010 |
| $RC_6$ | 1001111 | $RC_7$ | 1010101 | $RC_8$ | 0111000 |
| $RC_9$ | 0011000 | $RC_{10}$ | 1010110 | $RC_{11}$ | 0110010 |
| $RC_{12}$ | 1111110 | $RC_{13}$ | 1111001 | $RC_{14}$ | 1011101 |
| $RC_{15}$ | 1100101 | $RC_{16}$ | 0100101 | $RC_{17}$ | 0001001 |
| $RC_{18}$ | 0110100 | $RC_{19}$ | 0110011 | $RC_{20}$ | 1001111 |
| $RC_{21}$ | 0001101 | $RC_{22}$ | 1000010 | $RC_{23}$ | 0010101 |

Αυτή η μείωση στο μέγεθος της γεννήτριας RC απλοποιεί τον υπολογισμό στο βήμα $\iota$ του αλγορίθμου SHA-3. Προηγουμένως, το βήμα $\iota$ απαιτούσε 64 λογικές λειτουργίες XOR. Ωστόσο, με την απλοποιημένη γεννήτρια RC, ο αριθμός των λειτουργιών XOR που απαιτούνται στο βήμα $\iota$ μειώνονται μόνο σε επτά. Συγκεκριμένα, η λειτουργία XOR bitwise εκτελείται τώρα στις θέσεις bit 0, 1, 3, 7, 15, 31 και 63 του πίνακα

καταστάσεων $A[0,0]$ όπως φαίνεται στον Πίνακα 5.5, σύμφωνα με Εξίσωση (5.3). Αυτές οι θέσεις αντιστοιχούν στα μη μηδενικά bit στις απλοποιημένες τιμές της γεννήτριας RC.

Πίνακας 5.5: Οι θέσεις με μη μηδενικά bit.

| **g** | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| **[z]** | 0 | 1 | 3 | 7 | 15 | 31 | 63 |

Ο Πίνακας 5.6 απεικονίζει ένα παράδειγμα των απλουστευμένων τιμών που χρησιμοποιούνται για την γεννήτρια RC. Βελτιστοποιώντας τη γεννήτρια RC και απλοποιώντας το βήμα $\iota$, επιτυγχάνουμε μια πιο αποτελεσματική διαδικασία υπολογισμού στον αλγόριθμο SHA-3. Αυτή η βελτίωση συμβάλλει στη μείωση των υπολογιστικών επιβαρύνσεων και στη βελτίωση της συνολικής απόδοσης της συνάρτησης κατακερματισμού.

Πίνακας 5.6: Παράδειγμα των απλουστευμένων τιμών που χρησιμοποιούνται στην γεννήτρια $RC_7$.

| **Δεκαεξαδικό** | **Δυαδικό** | **Θέσεις με τιμή 1** |
|---|---|---|
| 8009 | 1000000000001001 | 15th=1, 7th=0, 3rd=1, 1st=0, 0th=1 |
| 0000 | 0000000000000000 | 31st=0 |
| 0000 | 0000000000000000 | - |
| 8000 | 1000000010001000 | 63th=1 |

### 5.4.4 Διαδικασία περικοπής

Η διαδικασία περικοπής στον αλγόριθμο SHA-3 χρησιμεύει ως αντίστροφη λειτουργία στη φάση της χαρτογράφησης. Στοχεύει στη δημιουργία μιας δυαδικής λέξης (string) 1600 bits από μια κατάσταση που αναπαρίσταται ως τρισδιάστατος πίνακας με διαστάσεις $5 \times 5 \times 64$ bit. Μόλις δημιουργηθεί η δυαδική λέξη 1600 bits, υποβάλλεται σε μια διαδικασία τμηματοποίησης για να παραχθεί μια έξοδος σύνοψης του επιθυμητού μεγέθους. Το μέγεθος εξόδου της σύνοψης μπορεί να ποικίλλει ανάλογα με τις συγκεκριμένες απαιτήσεις ή το επίπεδο ασφάλειας για την εφαρμογή του αλγόριθμου SHA-3. Η διαδικασία περικοπής περιλαμβάνει τη μετατροπή του τρισδιάστατου πίνακα καταστάσεων σε μια γραμμική ακολουθία bit. Τα $5 \times 5 \times 64$ bit συνδέονται μεταξύ τους, με αποτέλεσμα μια δυαδική λέξη 1600 bit.

Στη συνέχεια, η δυαδική λέξη 1600 bit που δημιουργείται, τμηματοποιείται ή χωρίζεται σε μικρότερα τμήματα, για να παραχθεί η επιθυμητή έξοδος σύνοψης.

Η διαδικασία τμηματοποίησης τυπικά περιλαμβάνει την εξαγωγή μιας συνεχούς ακολουθίας bits από τη δυαδική λέξη, η οποία ταιριάζει με το επιθυμητό μέγεθος εξόδου. Περικόπτοντας τη δυαδική λέξη των 1600 bit και εξάγοντας το κατάλληλο τμήμα, ο αλγόριθμος SHA-3 παράγει την τελική έξοδο σύνοψης, η οποία προκύπτει από την εφαρμογή της συνάρτησης κατακερματισμού κρυπτογράφησης στα δεδομένα εισόδου. Η έξοδος σύνοψης μπορεί να είναι διαφόρων μεγεθών, όπως 224, 256, 384 ή 512 bits, ανάλογα με τη συγκεκριμένη παραλλαγή του SHA-3 που χρησιμοποιείται και το επιθυμητό επίπεδο ασφάλειας.

## 5.5 Πειραματικά Αποτελέσματα

Στη μελέτη μας, χρησιμοποιήσαμε τις πλακέτες Virtex-5, Virtex-6 και Virtex-7 FPGA για να συγκρίνουμε διεξοδικά την προτεινόμενη στρατηγική με άλλες υπάρχουσες μελέτες, διασφαλίζοντας μια δίκαιη αξιολόγηση. Για την υλοποίηση των μεθόδων, χρησιμοποιήσαμε το λογισμικό Xilinx ISE για τα σχέδια στις πλακέτες Virtex-5 και Virtex-6, ενώ η σχεδίαση στην πλακέτα Virtex-7 υλοποιήθηκε με τη χρήση του λογισμικού Xilinx Vivado. Οι πληροφορίες που παρέχονται στους Πίνακες 5.7 και 5.8 αντιστοιχούν στα αποτελέσματα που προέκυψαν από το στάδιο μετά την υλοποίηση στη διαδικασία σχεδιασμού στις πλακέτες FPGA. Θέλουμε να τονίσουμε ότι το στάδιο μετά την υλοποίηση είναι κρίσιμο, καθώς λαμβάνει υπόψη τον πλήρη σχεδιασμό και παρέχει την πιο ακριβή αναπαράσταση των πόρων που χρησιμοποιούνται από τη σχεδίαση στην πλακέτα FPGA.

### 5.5.1 Δοκιμές επαλήθευσης

Πραγματοποιήσαμε προσομοιώσεις και δοκιμές επαλήθευσης για να επικυρώσουμε τη λειτουργικότητα των τεχνικών μας. Συγκεκριμένα, χρησιμοποιήσαμε έγκυρα δείγματα που παρέχονται από τον οργανισμό NIST [156] για να επαληθεύσουμε την πλήρη λειτουργία της υλοποίησής μας. Αυτή η διαδικασία επικύρωσης διασφαλίζει ότι οι εφαρμοζόμενες τεχνικές λειτουργούν, όπως προβλέπεται από τον οργανισμό με τα σωστά και τα επιθυμητά αποτελέσματα, και επαληθεύει εάν τα αποτελέσματα ταιριάζουν με τα αναμενόμενα, διασφαλίζοντας ότι οι μέθοδοι είναι αξιόπιστες.

### 5.5.2 Μετρήσεις απόδοσης και αποτελέσματα της αρχιτεκτονικής μας

Τα αποτελέσματα υλοποίησης στις πλακέτες FPGAs εξετάστηκαν εκτενώς για να αξιολογηθούν διάφορες τυπικές μετρικές απόδοσης για να εξασφαλιστεί μια δίκαιη και ουσιαστική σύγκριση που χρησιμοποιείται στην υπάρχουσα βιβλιογραφία [141], συμπεριλαμβανομένης της επιτεύξιμης συχνότητας (μέγιστης), της χρήσης επιφάνειας επικάλυψης, της αποδοτικότητας και της ρυθμαπόδοσης. Η ρυθμαπόδοση [211] είναι ένα κρίσιμο μέτρο στον κατακερματισμό μηνυμάτων, καθώς καθορίζει τον ρυθμό με τον οποίο μπορούν να υποστούν επεξεργασία τα μηνύματα. Η υψηλότερη ρυθμαπόδοση υποδηλώνει τη δυνατότητα χειρισμού μεγαλύτερου αριθμού μηνυμάτων εντός ενός δεδομένου χρονικού πλαισίου, κάτι που είναι επιθυμητό για εφαρμογές που απαιτούν γρήγορο και αποτελεσματικό κατακερματισμό.

$$Throughput_{Fpga} = \frac{\textit{Bitrate size "r"}}{\textit{Total clock cycles}} \times \textit{ Frequency maximum clock} \tag{5.4}$$

Η επιτεύξιμη συχνότητα [212] αντιπροσωπεύει τη μέγιστη συχνότητα ρολογιού που η σχεδίαση στην πλακέτα FPGA επιτρέπει να λειτουργήσει αξιόπιστα. Υποδεικνύει την ταχύτητα με την οποία το σύστημα μπορεί να επεξεργαστεί τα εισερχόμενα δεδομένα και να εκτελέσει τις λειτουργίες κατακερματισμού. Μια υψηλότερη επιτεύξιμη συχνότητα σημαίνει βελτιωμένες δυνατότητες επεξεργασίας και ταχύτερη συνολική ρυθμαπόδοση.

Στο πλαίσιο της υλοποίησης στην πλακέτα FPGA, με την αποδοτικότητα [168] αξιολογούμε την αναλογία της χρήσιμης εργασίας που εκτελείται προς την ποσότητα των πόρων που χρησιμοποιούνται. Παρέχονται πληροφορίες για τη συνολική αποτελεσματικότητα του σχεδιασμού και την ικανότητά του να επιτυγχάνει τους επιθυμητούς στόχους με ελάχιστη σπατάλη ή πλεονασμό υλικού. Οι υψηλότερες τιμές αποδοτικότητας σημαίνουν βελτιστοποιημένη χρήση των πόρων της πλακέτα FPGA και βελτιωμένη αποδοτικότητα.

$$Efficiency_{Fpga} = \frac{Throughput_{Fpga}}{Area_{Fpga}} \tag{5.5}$$

Η χρήση της επιφάνειας επικάλυψης [213, 214] αναφέρεται στην ποσότητα των πόρων της πλακέτας FPGA που καταναλώνονται από τη σχεδίαση. Η χρήση χαμηλότερης επιφάνειας επικάλυψης συνεπάγεται αποτελεσματικότερη χρήση των πόρων FPGA και δυνητικά χαμηλότερο κόστος παραγωγής. Αυτές οι μετρήσεις παρουσιάζονται στον Πίνακα 5.7, επιτρέποντας μια σαφή σύγκριση και ανάλυση των αποτελεσμάτων.

Πίνακας 5.7: Τα αποτελέσματα υλοποίησης στις πλακέτες FPGA.

| **Μετρικές** | **Μέγεθος μπλοκ $r$** | **Virtex-5** | **Virtex-6** | **Virtex-7** |
|---|---|---|---|---|
| Συχνότητα (MHz) | | 272,41 | 344,62 | 396,28 |
| Επιφάνεια (slices) | | 1186 | 1348 | 1452 |
| Ρυθμαπόδοση (Gbps) | 1152 | 26,151 | 33,084 | 38,043 |
| | 1088 | 24,699 | 31,246 | 35,929 |
| | 832 | 18,887 | 23,894 | 27,475 |
| | 576 | 13,076 | 16,542 | 19,021 |
| Αποδοτικότητα (Mbps/slices) | 1152 | 22,05 | 24,54 | 26,20 |
| | 1088 | 20,83 | 23,18 | 24,74 |
| | 832 | 15,93 | 17,73 | 18,92 |
| | 576 | 11,03 | 12,27 | 13,10 |

Όπως φαίνεται στον Πίνακα 5.7, η πλακέτα Virtex-7 FPGA παρουσιάζει την υψηλότερη χρήση επιφάνειας επικάλυψης μεταξύ των τριών πλακετών, με 1452 slices. Η πλακέτα Virtex-6 την ακολουθεί με 1348 slices και η πλακέτα Virtex-5 με 1186 slices. Δεύτερον, όσον αφορά τη συχνότητα, η πλακέτα Virtex-7 FPGA επιτυγχάνει την υψηλότερη τιμή των 396,28 MHz, υποδεικνύοντας την ικανότητά της να λειτουργεί με μεγαλύτερη ταχύτητα ρολογιού. Η πλακέτα Virtex-6 ακολουθεί αρκετά κοντά με 344,62 MHz, ενώ η πλακέτα Virtex-5 έχει τη χαμηλότερη συχνότητα στα 272,41 MHz. Τέλος, η πλακέτα Virtex-7 παρουσιάζει σταθερά την υψηλότερη αποδοτικότητα και ρυθμαπόδοση σε όλες τις τιμές $r$, ακολουθούμενη από τις πλακέτες Virtex-6 και Virtex-5.

### 5.5.3 Συγκριτική ανάλυση με άλλα ισοδύναμα μοντέλα

Ο Πίνακας 5.8 εμφανίζει τη σύγκριση με άλλα ισοδύναμα μοντέλα για μήκος εξόδου 512 bit, εστιάζοντας στη συχνότητα (MHz), την επιφάνεια σε (slices), την ρυθμαπόδοση (Gbps) και την απόδοτικότητα (Mbps/slice) για τον αλγόριθμο SHA-3. Όλα τα αναφερόμενα αποτελέσματα βασίζονται σε μηνύματα ενός μπλοκ. Ο προτεινόμενος σχεδιασμός που χρησιμοποιεί την πλακέτα Virtex-5

FPGA επιτυγχάνει πλήθος 1186 slices, που είναι χαμηλότερος από τον αριθμό τμημάτων των σχεδίων Virtex-5 που παρουσιάζονται στις μελέτες [146, 149, 153, 209, 210]. Αν και η συχνότητα λειτουργίας του προτεινόμενου σχεδίου στην πλακέτα Virtex-5 είναι 272,41 MHz, η οποία είναι χαμηλότερη από την υψηλότερη συχνότητα που αναφέρεται στο [153] (387 MHz), καταφέρνει να επιτύχει υψηλότερη ρυθμαπόδοση 13,076 Gbps σε σύγκριση με το άλλο σχέδιο σε Virtex-5. Επιπλέον, ο προτεινόμενος σχεδιασμός Virtex-5 επιδεικνύει υψηλότερη αποδοτικότητα με ρυθμό 11,03 Mbps/slice, ξεπερνώντας την απόδοση των άλλων σχεδίων σε πλακέτες Virtex-5.

Πίνακας 5.8: Αποτελέσματα και συγκρίσεις για τον αλγόριθμο SHA-3 μήκους εξόδου 512 bits.

| **Σχεδίαση** | **FPGA** | **Επιφάνεια (Slices)** | **Συχνότητα (MHz)** | **Ρυθμαπόδοση (Gbps) r = 576** | **Αποδοτικότητα (Mbps/Slices) r = 576** |
|---|---|---|---|---|---|
| [153] | Virtex-5 | 1680 | 387 | 8,06 | 4,91 |
| [186] | Virtex-7 | 1454 | 374,035 | 7,979 | 5,49 |
| [209] | Virtex-5 | 1409 | 377,86 | 8,22 | 5,83 |
| | Virtex-6 | 1227 | 424,44 | 10,19 | 8,30 |
| [210] | Virtex-5 | 1388 | 287,39 | 11,50 | 8,48 |
| | Virtex-6 | 1167 | 394,01 | 15,76 | 13,83 |
| | Virtex-7 | 1418 | 414,54 | 16,58 | 11,97 |
| [149] | Virtex-5 | 2652 | 352 | 8,44 | 6,37 |
| | Virtex-6 | 2296 | 391 | 9,38 | 8,17 |
| [146] | Virtex-5 | 2326 | 306 | 5,56 | 2,40 |
| [148] | Virtex-5 | 1163 | 273 | 7,80 | 6,06 |
| **Προτεινόμενος** | Virtex-5 | 1186 | 272,41 | 13,076 | 11,03 |
| | Virtex-6 | 1348 | 344,62 | 16,542 | 12,27 |
| | Virtex-7 | 1452 | 396,28 | 19,021 | 13,10 |

Ο προτεινόμενος σχεδιασμός στην πλακέτα Virtex-6 FPGA εμφανίζει υψηλότερο αριθμό slices 1348, σε σύγκριση με τα σχέδια άλλων ερευνητών στην ίδια πλακέτα [149, 209, 210]. Αν και η προτεινόμενη σχεδίαση στην Virtex-6 λειτουργεί με συχνότητα 344,62 MHz, η οποία είναι χαμηλότερη από την υψηλότερη συχνότητα που αναφέρεται στο [209] με 424,44 MHz, επιτυγχάνει υψηλότερη ρυθμαπόδοση 16,542 Gbps σε σύγκριση με την άλλη σχεδίαση στην Virtex-6. Ομοίως, ο προτεινόμενος σχεδιασμός στην Virtex-6 παρουσιάζει βελτιωμένη αποδοτικότητα με τιμή 12,27 Mbps/slice, ξεπερνώντας την αποδοτικότητα των άλλων σχεδίων στην ίδια πλακέτα Virtex-6.

Επιπλέον, ο προτεινόμενος σχεδιασμός που χρησιμοποιούμε στην πλακέτα Virtex-7 FPGA παρουσιάζει έναν αριθμό slices 1452, συγκρίσιμο με αυτόν των διαφόρων σχεδίων σε πλακέτες Virtex-7 που αναφέρονται στις μελέτες [186, 210]. Η συχνότητα

λειτουργίας του προτεινόμενου σχεδίου στην Virtex-7 είναι 396,28 MHz, ξεπερνώντας τις συχνότητες που αναφέρονται στις άλλες μελέτες, οι οποίες κυμαίνονται από 374,035 MHz έως 414,54 MHz. Επιπλέον, ο προτεινόμενος σχεδιασμός στην Virtex-7 επιτυγχάνει υψηλότερη ρυθμαπόδοση 19,021 Gbps από τα άλλα σχέδια στην Virtex-7. Επιπλέον, η αποδοτικότητα του προτεινόμενου σχεδίου στην Virtex-7 είναι 13,10 Mbps/slice, που είναι υψηλότερη από την απόδοση των άλλων σχεδίων σε πλακέτες Virtex-7. Η παραπάνω ανάλυση καθιστά απολύτως σαφές ότι ο σχεδιασμός που έχει προταθεί επιδεικνύει ανώτερη απόδοση τόσο όσον αφορά την ρυθμαπόδοση όσο και την αποδοτικότητα σε σύγκριση με τα άλλα σχέδια που έχουν συζητηθεί στις σχετικές δημοσιεύσεις.

## 5.6 Συζήτηση της στρατηγικής για την βελτιστοποίηση του SHA-3

Οι κρυπτογραφικοί αλγόριθμοι και οι υλοποιήσεις τους σε υλικό, έχουν σημειώσει σημαντική πρόοδο τα τελευταία χρόνια. Ωστόσο, παρά την πρόοδο αυτή, εξακολουθούν να υπάρχουν ορισμένες προκλήσεις και κενά. Ένα σημαντικό ζήτημα είναι η συνεχώς αυξανόμενη ζήτηση για βελτίωση της απόδοσης στα κρυπτογραφικά συστήματα. Οι παραδοσιακοί κρυπτογραφικοί αλγόριθμοι χρειάζονται εξέλιξη για να συμβαδίζουν με την αυξανόμενη υπολογιστική ισχύ των σύγχρονων αντιπάλων. Επιπλέον, η ανάγκη για ταχύτερες και πιο αποδοτικές από πλευράς πόρων υλοποιήσεις σε περιορισμένα περιβάλλοντα, όπως συσκευές IoT και ενσωματωμένα συστήματα, παρουσιάζει ένα μοναδικό σύνολο προκλήσεων. Οι υπάρχουσες συναρτήσεις κατακερματισμού, αν και αποτελεσματικές, συχνά δυσκολεύονται να επιτύχουν τη σωστή ισορροπία μεταξύ απόδοσης και χρήσης πόρων. Αυτό δημιουργεί ένα ερευνητικό κενό όπου υπάρχει χώρος για καινοτόμες λύσεις που αντιμετωπίζουν αυτές τις προκλήσεις ολοκληρωμένα.

Το κίνητρο πίσω από την έρευνά μας πηγάζει από το προαναφερθέν κενό και την ανάγκη για νέες προσεγγίσεις που μπορούν να γεφυρώσουν το χάσμα μεταξύ της χρήσης πόρων και της αποτελεσματικότητας. Η συνάρτηση κατακερματισμού SHA-3 υπόσχεται λόγω των ισχυρών ιδιοτήτων ασφαλείας της για αποτελεσματική εφαρμογή υλικού. Η κατασκευή σφουγγαριού του SHA-3 προσφέρει την ευελιξία προσαρμογής της λειτουργίας κατακερματισμού σε ποικίλες απαιτήσεις ασφαλείας χωρίς συμβιβασμούς στην απόδοση. Η τεχνολογία FPGA παρουσιάζει μια ευκαιρία

για αξιοποίηση της επιτάχυνσης υλικού για την επίτευξη αποτελεσματικών και προσαρμόσιμων κρυπτογραφικών υλοποιήσεων. Ωστόσο, οι περιορισμοί πόρων μπορούν να κάνουν τη βελτιστοποίηση κρυπτογραφικών αλγορίθμων για αυτές τις συσκευές δύσκολη.

Η προτεινόμενη προσέγγισή μας επικεντρώνεται στην αξιοποίηση των δυνατοτήτων του αλγορίθμου SHA-3 και της τεχνολογίας FPGA για την αντιμετώπιση των προκλήσεων που θέτει το ερευνητικό κενό. Διερευνώντας τις δυνατότητες της επιτάχυνσης FPGA για κρυπτογραφικές λειτουργίες που βασίζονται στο SHA-3, στοχεύουμε να παρέχουμε μια λύση που ενισχύει αποτελεσματικές κρυπτογραφικές εφαρμογές. Η εργασία μας συμβάλλει στο σύνολο της γνώσης, επιδεικνύοντας τη σκοπιμότητα και τα πλεονεκτήματα των εφαρμογών στον αλγόριθμο SHA-3 που βασίζονται σε FPGA. Τονίζουμε τις δυνατότητες της τεχνολογίας FPGA για την επίτευξη αρμονικής συνέργειας μεταξύ της κρυπτογραφικής ισχύος και της υπολογιστικής αποτελεσματικότητας. Τα ευρήματά μας ανοίγουν δρόμους για περαιτέρω έρευνα για τη βελτιστοποίηση και τη βελτίωση των κρυπτογραφικών συστημάτων που βασίζονται σε FPGA με επεκτάσεις σε άλλους κρυπτογραφικούς αλγόριθμους και εφαρμογές.

## 5.7 Συμπεράσματα κεφαλαίου και μελλοντικές εργασίες

Οι λειτουργίες κατακερματισμού διαδραματίζουν ουσιαστικό ρόλο στον τομέα της ασφάλειας πληροφοριών, εξυπηρετώντας διάφορους σκοπούς στον σημερινό ψηφιακό κόσμο. Η σημασία των συναρτήσεων κατακερματισμού εκτείνεται σε διάφορους τομείς, συμπεριλαμβανομένου του στρατιωτικού, του διαδικτυακού εμπορίου, των τραπεζών, της διαχείρισης υγειονομικής περίθαλψης και του Διαδικτύου των Πραγμάτων. Μεταξύ των διαφόρων διαθέσιμων αλγορίθμων κατακερματισμού, ο αλγόριθμος SHA-3 ξεχωρίζει για το σημαντικά υψηλότερο επίπεδο ασφάλειας. Ο αλγόριθμος SHA-3 παρέχει έναν κατάλληλο συνδυασμό απόδοσης, επιτάχυνσης και ασφάλειας, καθιστώντας τον μια προτιμώμενη επιλογή για πολλές κρυπτογραφικές εφαρμογές στο σημερινό τοπίο ασφάλειας πληροφοριών.

Η έμφαση αυτού του άρθρου είναι στη μελέτη της βέλτιστης ρυθμαπόδοσης της αποδοτικότητας και των κριτηρίων απόδοσης για τον αλγόριθμο SHA-3 σε διάφορα

μήκη εξόδου (224, 256, 384 και 512 bits) στα Virtex-5, Virtex-6 και Virtex-7 πλακέτες FPGA. Διεξάγοντας μια ολοκληρωμένη ανάλυση, συγκρίνουμε την προσέγγισή μας με παρόμοια σχέδια και αποδεικνύουμε ότι η προτεινόμενη στρατηγική μας επιτυγχάνει την υψηλότερη απόδοση όσον αφορά τα τυπικά μέτρα αξιολόγησης της ρυθμαπόδοσης και της αποδοτικότητας. Οι υψηλότεροι ρυθμοί απόδοσης για μήκος εξόδου 512 bits ήταν πάνω από 11,37% Gbps στο Virtex-5, 10,49% Gbps στο Virtex-6 και 11,47% Gbps στο Virtex-7 σε σύγκριση με άλλα πρόσφατα ισοδύναμα μοντέλα.

Σε μελλοντικές εργασίες, σκοπεύουμε να βελτιώσουμε τη συνολική απόδοση και τα μέτρα απόδοσης (ρυθμαπόδοση και αποδοτικότητα) ανά γύρο.

# Κεφάλαιο 6

# Συμπεράσματα και προτάσεις για μελλοντική έρευνα

Σε αυτό το κεφάλαιο αναλύονται τα συμπεράσματα και τα ευρήματα που προέκυψαν από την παρούσα διατριβή σχετικά με τις τεχνικές βελτιστοποίησης διασωλήνωσης, ξετυλίγματος και του υβριδικού τους συνδυασμού, με επίκεντρο τον αλγόριθμο SHA-3 και την υλοποίησή του σε συσκευές FPGA. Η προτεινόμενη προσέγγιση οδήγησε στη διατύπωση σημαντικών συμπερασμάτων, υπογραμμίζοντας τη συμβολή των βελτιστοποιήσεων αυτών όχι μόνο στη βελτίωση της ρυθμαπόδοσης και της αποδοτικότητας, αλλά και στην ουσιαστική επιτάχυνση της εκτέλεσης του SHA-3 σε επίπεδο υλικού.

Η ανάλυση καταδεικνύει ότι η βελτιστοποίηση μέσω διασωλήνωσης και ξετυλίγματος μειώνει την κρίσιμη διαδρομή και ενισχύει την παραλληλοποίηση, οδηγώντας σε ταχύτερους χρόνους εκτέλεσης, υψηλότερη συχνότητα λειτουργίας και πιο αποδοτική χρήση των διαθέσιμων πόρων. Αυτή η επιτάχυνση καθιστά τον SHA-3 ιδιαίτερα κατάλληλο για εφαρμογές που απαιτούν υψηλές επιδόσεις και αυξημένες απαιτήσεις ασφάλειας σε πραγματικό χρόνο.

Τέλος, αναδεικνύονται οι υφιστάμενες προκλήσεις και η ανάγκη για συνεχή ανάπτυξη καινοτόμων τεχνικών βελτιστοποίησης, που θα επιτρέψουν ακόμη ταχύτερη και πιο αποδοτική εκτέλεση του αλγορίθμου, ανταποκρινόμενες στις διαρκώς αυξανόμενες απαιτήσεις ρυθμαπόδοσης της σύγχρονης ψηφιακής εποχής. Μέσα από την παρούσα προσέγγιση καθίσταται δυνατή η αναγνώριση νέων κατευθύνσεων για την περαιτέρω εξέλιξη του SHA-3, προωθώντας την έρευνα πέρα από τα υφιστάμενα όρια και ενθαρρύνοντας την ανάπτυξη λύσεων που

θα ενισχύουν την επιτάχυνση και την αποδοτικότητα σε εφαρμογές αιχμής της ψηφιακής ασφάλειας.

## 6.1 Συμπεράσματα των δύο μεθόδων βελτιστοποίησης διοχετεύσεων υλικού

Για τη βελτίωση της απόδοσης του αλγορίθμου SHA-3, καθίσταται κρίσιμη η εστίαση στον εντοπισμό και τη βελτιστοποίηση των υπολογιστικά πιο απαιτητικών σταδίων της διαδικασίας. Ένα από τα πλέον επιβαρυντικά στάδια αφορά τον υπολογισμό των bit ισοτιμίας στις στήλες του πίνακα κατάστασης, ο οποίος προϋποθέτει την πρόσβαση σε ολόκληρο τον πίνακα. Η διαδικασία αυτή συνεπάγεται εκτεταμένη κίνηση δεδομένων και υψηλό υπολογιστικό κόστος, με αποτέλεσμα αυξημένη κατανάλωση πόρων και αρνητική επίδραση στη συνολική ρυθμαπόδοση και αποδοτικότητα του αλγορίθμου.

Η βελτιστοποίηση του εν λόγω σταδίου είναι καθοριστικής σημασίας, καθώς επιτρέπει τη μείωση της κρίσιμης διαδρομής και τη βελτίωση της διαχείρισης δεδομένων, οδηγώντας σε σημαντική επιτάχυνση της εκτέλεσης του SHA-3. Η επιτάχυνση αυτή ενισχύει τη ρυθμαπόδοση και την αποδοτικότητα της υλοποίησης, καθιστώντας τον αλγόριθμο πιο αποτελεσματικό για απαιτητικά υπολογιστικά περιβάλλοντα, όπως οι συσκευές FPGA, όπου η απόδοση σε πραγματικό χρόνο είναι καίριας σημασίας.

Η εφαρμογή των τεχνικών βελτιστοποίησης που περιορίζουν τον όγκο της κίνησης δεδομένων και μειώνουν το υπολογιστικό κόστος μπορεί να οδηγήσει σε ουσιαστικές βελτιώσεις της συνολικής απόδοσης. Ιδιαίτερα, η αξιοποίηση των τεχνικών διασωλήνωσης (pipelining) συμβάλλει καθοριστικά στη μείωση της χρονικής καθυστέρησης, ενισχύοντας τη ρυθμαπόδοση του αλγορίθμου. Μέσω της διασωλήνωσης, οι επιμέρους λειτουργίες εκτελούνται με αυξημένο βαθμό παραλληλίας, γεγονός που επιτρέπει τη σημαντική επιτάχυνση της επεξεργασίας.

Η επιτάχυνση αυτή έχει ως άμεσο αποτέλεσμα τη μείωση του κόστους και του χρόνου της επεξεργασίας, καθώς και τη βελτιστοποίηση της χρήσης των διαθέσιμων πόρων. Κατά συνέπεια, η υλοποίηση του αλγορίθμου SHA-3 καθίσταται αποδοτικότερη και πιο κατάλληλη για απαιτητικά περιβάλλοντα υψηλών επιδόσεων, όπως οι συσκευές FPGA, όπου η επίτευξη υψηλής ταχύτητας εκτέλεσης αποτελεί κρίσιμο παράγοντα.

Ο Πίνακας 6.1 εμφανιζει την σύγκριση των δύο μεθόδων βελτιστοποίησης διοχετεύσεων υλικού για την υλοποίηση του αλγορίθμου SHA-3 στις συσκευές FPGA (Virtex-5, Virtex-6, και Virtex-7) αποκαλύπτει σημαντικές διαφορές στην ρυθμαπόδοση και την αποδοτικότητα των δύο προσεγγίσεων.

Πίνακας 6.1: Σύγκριση των δύο μεθόδων βελτιστοποίησης διοχετεύσεων υλικού στις συσκευές FPGA (Virtex-5, Virtex-6, και Virtex-7).

| **Μετρικές** | **Μήκος** | **Πρώτη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση όπου ο πρώτος αγωγός τοποθετείται μετά το βήμα $\pi$** | | | **Δευτερη προτεινόμενη τεχνική βελτιστοποίησης με διασωλήνωση όπου ο πρώτος αγωγός τοποθετείται μετά το βήμα $\theta$** | | |
|---|---|---|---|---|---|---|---|
| **FPGA** | | Virtex-5 | Virtex-6 | Virtex-7 | Virtex-5 | Virtex-6 | Virtex-7 |
| **Επιφάνεια (slices)** | | 1102 | 1146 | 1288 | 998 | 1042 | 1150 |
| **Συχνότητα (MHz)** | | 374 | 392 | 446 | 402 | 422 | 478 |
| **Ρυθμαπόδοση (Gbps)** | $r$ = 1152 | 17,952 | 18,816 | 21,408 | 19,296 | 20,256 | 22,944 |
| | $r$ = 1088 | 16,955 | 17,771 | 20,219 | 18,224 | 19,131 | 21,669 |
| | $r$ = 832 | 12,965 | 13,589 | 15,461 | 13,936 | 14,629 | 16,571 |
| | $r$ = 576 | 8,976 | 9,408 | 10,704 | 9,648 | 10,128 | 11,472 |
| **Αποδοτικότητα (Mbps/slices)** | $r$ = 1152 | 16,29 | 16,42 | 16,62 | 19,33 | 19,44 | 19,95 |
| | $r$ = 1088 | 15,39 | 15,51 | 15,70 | 18,26 | 18,36 | 18,84 |
| | $r$ = 832 | 11,77 | 11,86 | 12,00 | 13,96 | 14,04 | 14,41 |
| | $r$ = 576 | 8,15 | 8,21 | 8,31 | 9,67 | 9,72 | 9,98 |

Η ενσωμάτωση ενός καταχωρητή αμέσως μετά το βήμα $\theta$ αναδεικνύεται ως μία κρίσιμης σημασίας τεχνική για τη βελτίωση της απόδοσης της συνάρτησης μετάθεσης του αλγορίθμου SHA-3. Αυτό το βήμα, αναγνωριζόμενο ως το πλέον υπολογιστικά απαιτητικό μέρος της διαδικασίας, καταλαμβάνει περισσότερο από το μισό του συνολικού υπολογιστικού χρόνου. Ο καταχωρητής, ως προσωρινή μονάδα αποθήκευσης, αποθηκεύει τα υπολογισμένα bit ισοτιμίας, διαγράφοντας την ανάγκη για επαναλαμβανόμενη πρόσβαση σε ολόκληρο τον πίνακα καταστάσεων και μειώνοντας σημαντικά την κίνηση δεδομένων και τον επανυπολογισμό. Αυτή η τακτική μειώνει δραστικά το υπολογιστικό φορτίο και τις απαιτήσεις πόρων στα επόμενα στάδια, ενισχύοντας την αποδοτικότητα και τη ρυθμαπόδοση του αλγορίθμου. Επιπροσθέτως, η απλοποίηση της ροής δεδομένων εντός του αλγορίθμου επιτρέπει ταχύτερη και πιο αποτελεσματική επεξεργασία. Η εφαρμογή αυτής της βελτιστοποίησης, επομένως, συμβάλλει στην ελαχιστοποίηση του συνολικού κόστους υλοποίησης του SHA-3, βελτιστοποιώντας παράλληλα τη χρήση πόρων.

Αντιθετικά, η τοποθέτηση ενός καταχωρητή μετά το βήμα $\pi$ προσφέρει βελτίωση στην απόδοση του αλγορίθμου, αλλά σε μικρότερο βαθμό σε σχέση με την τοποθέτηση μετά το βήμα $\theta$. Το βήμα $\pi$ εμπλέκεται κυρίως στην αναδιάταξη της σειράς των bit στον πίνακα καταστάσεων και ο υπολογισμός του απαιτεί λιγότερη ενταση σε σύγκριση με το βήμα $\theta$. Συνεπώς, ενώ η εισαγωγή του καταχωρητή μετά

το βήμα $\pi$ μπορεί να προσφέρει κάποιες βελτιώσεις στην απόδοση, αυτές δεν είναι τόσο σημαντικές όσο εκείνες που επιτυγχάνονται με την τοποθέτηση μετά το βήμα $\theta$. Αυτό καταδεικνύει τη σημασία της επιλογής του σωστού σημείου για την εισαγωγή τεχνικών βελτιστοποίησης στη δομή ενός αλγορίθμου. Η αποτελεσματικότητα μιας τέτοιας παρέμβασης εξαρτάται σημαντικά από την κατανόηση των υπολογιστικών απαιτήσεων κάθε βήματος του αλγορίθμου και την επίδρασή τους στη συνολική απόδοση.

Η αποδοτική διαχείριση των υπολογιστικών πόρων και η μείωση της ανάγκης για επανυπολογισμό ή εκτεταμένη μετακίνηση δεδομένων μέσω της εισαγωγής ενός καταχωρητή αποτελούν κρίσιμα βήματα προς την αύξηση της απόδοσης του αλγορίθμου SHA-3. Τέλος, αυτή η προσέγγιση όχι μόνο βελτιώνει την αποδοτικότητα και την ρυθμαπόδοση, αλλά συμβάλλει και στην ελαχιστοποίηση του συνολικού κόστους και της κατανάλωσης πόρων, προσφέροντας μια πιο οικονομικά αποδοτική λύση για την εφαρμογή του αλγορίθμου SHA-3. Αυτή η βελτιστοποίηση είναι ιδιαίτερα σημαντική σε εφαρμογές όπου οι πόροι είναι περιορισμένοι και η ανάγκη για υψηλή απόδοση είναι κρίσιμη, όπως σε ενσωματωμένα συστήματα και συσκευές IoT.

Η ενσωμάτωση τεχνολογιών που επιτρέπουν την ταχύτερη επεξεργασία και μείωση του υπολογιστικού φορτίου, όπως ο καταχωρητής μετά το βήμα $\theta$, προάγει την ανάπτυξη πιο αποδοτικών και ασφαλών κρυπτογραφικών λύσεων. Η προσέγγιση αυτή επιτρέπει την επίτευξη καλύτερων επιδόσεων χωρίς να θυσιάζεται η ασφάλεια ή η ακεραιότητα των δεδομένων, προσφέροντας ένα αξιόπιστο εργαλείο για την προστασία της πληροφορίας στον ψηφιακό κόσμο.

Συμπερασματα απο την σύγκριση των δύο μεθόδων βελτιστοποίησης διοχετεύσεων υλικού :

1. Επιφάνεια σε (slices): Η δεύτερη προτεινόμενη τεχνική παρουσιάζει μικρότερο αριθμό slices σε σχέση με την πρώτη τεχνική σε όλες τις συσκευές FPGA. Ειδικότερα, στο Virtex-7, η δεύτερη μέθοδος χρησιμοποιεί 138 λιγότερα slices από την πρώτη μέθοδο, υποδεικνύοντας μια πιο αποδοτική χρήση των πόρων.

2. Συχνότητα Λειτουργίας (MHz): Και οι δύο μέθοδοι παρουσιάζουν αυξημένες τιμές συχνότητας στα νεότερα μοντέλα FPGA (Virtex-5, Virtex-6 και Virtex-7). Η δεύτερη τεχνική όμως εμφανίζει υψηλότερες συχνότητες λειτουργίας σε όλες τις πλατφόρμες, υποδηλώνοντας ότι μπορεί να διαχειρίζεται αποδοτικότερα

τη διαδικασία κατακερματισμού. Ειδικότερα, η δεύτερη μέθοδος εμφανιζει σημαντικότερη διαφορά στο Virtex-7, όπου η διαφορά φτάνει τα 32 MHz.

3. Ρυθμαπόδοση (Gbps): Σε σύγκριση μεταξύ των δύο τεχνικών, παρατηρείται σημαντική βελτίωση της ρυθμαπόδοσης με την εφαρμογή της δεύτερης τεχνικής, ειδικά στη νεότερη συσκευή FPGA Virtex-7. Αυτό υποδηλώνει ότι η βελτιστοποίηση της ροής δεδομένων μέσω της εν λόγω τεχνικής αποδίδει καλύτερα σε πιο πρόσφατες και πιο ισχυρές πλατφόρμες, αξιοποιώντας την αυξημένη επεξεργαστική τους ικανότητα. Στο Virtex-7, με μήκος $r$=1152, η δεύτερη τεχνική επιτυγχάνει ρυθμαπόδοση 22,944 Gbps, η οποία είναι αισθητά υψηλότερη από την ρυθμαπόδοση 21,408 Gbps της πρώτης τεχνικής.

4. Αποδοτικότητα (Mbps/slices): Η δεύτερη προτεινόμενη τεχνική δείχνει επίσης υψηλότερη αποδοτικότητα σε σχέση με την πρώτη, σε όλες τις μετρήσεις του μήκους $r$. Η δεύτερη τεχνική, εκτός από την προσφορά υψηλότερης ρυθμαπόδοσης, κάνει πιο αποδοτική χρήση των πόρων του FPGA. Στο Virtex-7 με μήκος $r$=1152, η αποδοτικότητα της δεύτερης τεχνικής φτάνει τα 19,95 Mbps/slice, ενώ για την πρώτη τεχνική είναι 16,62 Mbps/slice. Αυτό υποδηλώνει ότι η δεύτερη τεχνική καταφέρνει να επιτύχει μεγαλύτερη ρυθμαπόδοση ανά χρησιμοποιούμενο slice, εξασφαλίζοντας έτσι μεγαλύτερη αποδοτικότητα στην επεξεργασία δεδομένων.

Η δεύτερη τεχνική βελτιστοποίησης διοχετεύσεων υλικού, με τον πρώτο αγωγό να τοποθετείται μετά το βήμα $\theta$, παρουσιάζει σημαντικά καλύτερες επιδόσεις σε όλους τους τομείς σε σχέση με την πρώτη προτεινόμενη τεχνική, που ενσωματώνει τον πρώτο αγωγό μετά το βήμα $\pi$. Αυτό καθιστά τη δεύτερη τεχνική προτιμότερη για την υλοποίηση του αλγορίθμου SHA-3 σε FPGA, καθώς προσφέρει μεγαλύτερη αποδοτικότητα, υψηλότερη ρυθμαπόδοση και καλύτερη χρήση πόρων.

## 6.2 Συμπεράσματα βελτιστοποίησης ξετυλίγματος υλικού

Η στρατηγική του ξετυλίγματος συνιστά μία κρίσιμη τεχνική βελτιστοποίησης για την υλοποίηση του αλγορίθμου SHA-3 σε συστήματα FPGA, προσφέροντας ουσιαστικά πλεονεκτήματα ως προς τη μείωση του χρόνου εκτέλεσης και την ενίσχυση της ρυθμαπόδοσης. Μέσω της διαδικασίας ξετυλίγματος καθίσταται

δυνατή η εκτέλεση πολλαπλών γύρων μετασχηματισμών σε κάθε κύκλο ρολογιού, γεγονός που οδηγεί σε σημαντική επιτάχυνση της επεξεργασίας δεδομένων.

Η επιτάχυνση αυτή μειώνει τις απαιτήσεις χρόνου για την ολοκλήρωση του αλγορίθμου και ενισχύει την αποδοτικότητα της υλοποίησης, καθιστώντας τον SHA-3 ιδιαίτερα κατάλληλο για εφαρμογές που απαιτούν υψηλές επιδόσεις σε πραγματικό χρόνο. Ως αποτέλεσμα, η τεχνική του ξετυλίγματος συμβάλλει καθοριστικά στην ανάπτυξη αρχιτεκτονικών με αυξημένη ταχύτητα και αποδοτικότητα, ικανοποιώντας τις διαρκώς αυξανόμενες απαιτήσεις επιτάχυνσης σε περιβάλλοντα FPGA.

Η εφαρμογή του συντελεστή ξετυλίγματος 2 καταδεικνύει την ικανότητα της μεθόδου να διπλασιάζει την ρυθμαπόδοση της υλοποίησης, μετατρέποντας την παραδοσιακή δομή του αλγορίθμου σε μια πιο αποτελεσματική μορφή. Ταυτόχρονα, αυτή η προσέγγιση επιτρέπει την ευκολότερη διαχείριση των δεδομένων και την μείωση της επιβάρυνσης στους πόρους της FPGA συσκευής, κάτι που είναι ιδιαίτερα σημαντικό σε εφαρμογές όπου οι πόροι είναι περιορισμένοι. Επιπλέον, η μείωση του αριθμού των απαιτούμενων κύκλων ρολογιού για την ολοκλήρωση του αλγορίθμου SHA-3 καθιστά την υλοποίηση πιο αποδοτική, ενισχύοντας την απόδοση και την εφαρμοσιμότητα του αλγορίθμου σε διάφορες εφαρμογές ασφαλείας.

Ο Πίνακας 6.2 εμφανιζει τα αποτελέσματα ρυθμαπόδοσης και αποδοτικότητας της μεθόδου βελτιστοποίησης ξετυλίγματος υλικού για την υλοποίηση του αλγορίθμου SHA-3 στις συσκευές FPGA (Virtex-5, Virtex-6, και Virtex-7).

Πίνακας 6.2: Σύγκριση της τεχνικής βελτιστοποίησης ξετυλίγματος σε διαφορετικές συσκευές FPGA για 12 και 24 κύκλους ρολογιού.

| Μετρικές | **Μήκος** | **12 κύκλοι ρολογιού** | | | **24 κύκλοι ρολογιού** | | |
|---|---|---|---|---|---|---|---|
| **FPGA** | | Virtex-5 | Virtex-6 | Virtex-7 | Virtex-5 | Virtex-6 | Virtex-7 |
| **Επιφάνεια (slices)** | | 1112 | 1287 | 1375 | 868 | 946 | 1094 |
| **Συχνότητα (MHz)** | | 203,28 | 347,84 | 378,73 | 347,49 | 438,49 | 498,27 |
| **Ρυθμαπόδοση (Gbps)** | $r$ = 1152 | 17,55 | 25,95 | 26,44 | 19,22 | 22,25 | 21,86 |
| | $r$ = 1088 | 6,57 | 24,50 | 24,97 | 18,15 | 21,01 | 20,65 |
| | $r$ = 832 | 12,67 | 18,74 | 19,10 | 13,88 | 16,07 | 15,79 |
| | $r$ = 576 | 8,77 | 12,97 | 13,22 | 9,61 | 11,12 | 10,93 |
| **Αποδοτικότητα (Gbps/slices)** | $r$ = 1152 | 19,515 | 33,393 | 36,358 | 16,680 | 21,048 | 23,917 |
| | $r$ = 1088 | 18,431 | 31,537 | 34,338 | 15,753 | 19,878 | 22,588 |
| | $r$ = 832 | 14,094 | 24,117 | 26,259 | 12,046 | 15,201 | 17,273 |
| | $r$ = 576 | 9,757 | 16,696 | 18,179 | 8,340 | 10,524 | 11,958 |

Συμπερασματα απο την τεχνική ξετυλίγματος:

1. Απόδοση συσκευών FPGA: Η εφαρμογή της τεχνικής ξετυλίγματος επηρεάζει θετικά την απόδοση των συσκευών FPGA, με τις συσκευές Virtex-7 να παρουσιάζουν την υψηλότερη αύξηση σε ρυθμαπόδοση και αποδοτικότητα όταν λειτουργούν με 12 κύκλους ρολογιού συγκριτικά με 24 κύκλους ρολογιού. Αυτό υποδηλώνει ότι η στρατηγική ξετυλίγματος μπορεί να βελτιστοποιήσει τη χρήση των πόρων του FPGA για να επιτύχει ταχύτερη επεξεργασία.

2. Μείωση κύκλων ρολογιού: Η μείωση των κύκλων ρολογιού από 24 σε 12 οδηγεί σε σημαντική αύξηση της αποδοτικότητας σε όλες τις μελετημένες συσκευές FPGA. Αυτό επιβεβαιώνει την αποτελεσματικότητα της τεχνικής ξετυλίγματος στη μείωση του απαιτούμενου χρόνου για την εκτέλεση των υπολογιστικών εργασιών.

3. Αποδοτικότητα και πόροι: Συγκριτικά, η αποδοτικότητα (μετρούμενη ως Mbps/slices) αυξάνεται με τη μείωση των κύκλων ρολογιού, δείχνοντας ότι μια μικρότερη ποσότητα πόρων του FPGA απαιτείται για να επιτευχθεί η ίδια ή και καλύτερη απόδοση. Αυτό σημαίνει ότι η βελτιστοποίηση ξετυλίγματος βοηθάει στην επίτευξη υψηλότερης αποδοτικότητας ανά χρησιμοποιούμενο πόρο, κάτι που είναι ζωτικής σημασίας σε εφαρμογές όπου οι πόροι είναι περιορισμένοι.

4. Συχνότητα λειτουργίας: Η αύξηση της μέγιστης επιτρεπόμενης συχνότητας λειτουργίας με την τεχνική ξετυλίγματος επιτρέπει την ταχύτερη επεξεργασία των δεδομένων στις συσκευές FPGA. Αυτό δείχνει ότι μπορούμε να επιτύχουμε μεγαλύτερη απόδοση σε μικρότερο χρονικό διάστημα, βελτιστοποιώντας την κατανάλωση ενέργειας και την αποδοτικότητα της συσκευής.

5. Συνολική επίδραση στην απόδοση: Η εφαρμογή της τεχνικής ξετυλίγματος παρέχει έναν σημαντικό τρόπο για την βελτίωση της απόδοσης των συσκευών FPGA, μειώνοντας τον απαιτούμενο χρόνο εκτέλεσης και αυξάνοντας την αποδοτικότητα της επεξεργασίας. Η επιλογή του κατάλληλου αριθμού κύκλων ρολογιού και η λεπτομερής ρύθμιση των παραμέτρων της τεχνικής είναι κρίσιμη για την επίτευξη των μέγιστων οφελών.

6. Αξιολόγηση κόστους-οφέλους: Σημαντικός παράγοντας για την εφαρμογή της τεχνικής ξετυλίγματος είναι η αξιολόγηση του κόστους-οφέλους, καθώς η βελτιστοποίηση μπορεί να απαιτεί επιπλέον πόρους ή να έχει επιπτώσεις στην κατανάλωση ενέργειας. Η συνολική απόδοση του συστήματος πρέπει να αξιολογηθεί με βάση τις συγκεκριμένες απαιτήσεις της εφαρμογής για να

διασφαλιστεί ότι οι οφέλη από την βελτίωση της απόδοσης υπερβαίνουν τυχόν επιπλέον κόστη.

Με τη συνεχή εξέλιξη της τεχνολογίας FPGA και την αυξανόμενη πολυπλοκότητα των εφαρμογών, η έρευνα και ανάπτυξη στρατηγικών βελτιστοποίησης, όπως η τεχνική του ξετυλίγματος, παραμένει καίριας σημασίας για τη βελτίωση της απόδοσης και της αποδοτικότητας των συστημάτων. Η διαρκής αναζήτηση προηγμένων λύσεων οδηγεί σε ουσιαστική επιτάχυνση της εκτέλεσης των αλγορίθμων, προσφέροντας παράλληλα μεγαλύτερη ευελιξία και επεκτασιμότητα στις αρχιτεκτονικές υλοποιήσεις.

Η ενσωμάτωση τεχνικών ξετυλίγματος σε συνδυασμό με άλλες στρατηγικές βελτιστοποίησης σε ολοκληρωμένα περιβάλλοντα ανάπτυξης και υπολογιστικά συστήματα παρέχει στους μηχανικούς και τους σχεδιαστές τα απαραίτητα εργαλεία για την αποτελεσματική αντιμετώπιση των σύγχρονων τεχνολογικών προκλήσεων. Η εφαρμογή αυτών των τεχνικών δεν βελτιώνει μόνο τη ρυθμαπόδοση και την αποδοτικότητα, αλλά επιτυγχάνει και σημαντική επιτάχυνση της επεξεργασίας, μειώνοντας τον χρόνο εκτέλεσης και επιτρέποντας την ανάπτυξη συστημάτων υψηλών επιδόσεων που ανταποκρίνονται στις αυξανόμενες απαιτήσεις της ψηφιακής εποχής.

## 6.3 Συμπεράσματα βελτιστοποίησης διοχετεύσεων και ξετυλίγματος υλικού

Στην προτεινόμενη βελτιστοποίησή μας, έχουμε εφαρμόσει μια προσέγγιση διασωλήνωσης δύο σταδίων εντός του μπλοκ μετάθεσης $f$ της συνάρτησης κατακερματισμού Keccak. Επιπλέον, ξετυλίξαμε τη συνολική συνάρτηση κατακερματισμού κατά 2 και εισάγαμε δύο αγωγούς μεταξύ των γύρων. Η διασωλήνωση δύο σταδίων χωρίζει συγκεκριμένα τον υπολογισμό μεταξύ του βήματος $\theta$ και των υπόλοιπων τεσσάρων βημάτων ($\rho$, $\pi$, $\chi$ και $\iota$) του μπλοκ μετάθεσης $f$. Αυτή η διαίρεση επιτρέπει την πιο αποτελεσματική επεξεργασία και μειώνει την κρίσιμη διαδρομή, στοχεύοντας τελικά στην επίτευξη υψηλότερης συχνότητας ρολογιού.

Στο πρώτο μισό του υπολογισμού, που περιλαμβάνει το βήμα $\theta$, η μεγαλύτερη καθυστέρηση περιλαμβάνει πέντε λειτουργίες XOR. Από την άλλη πλευρά, το

δεύτερο μισό, που καλύπτει τα βήματα $\pi$ έως $\iota$, επιφέρει την πιο εκτεταμένη καθυστέρηση δύο λειτουργιών XOR, μίας λειτουργίας AND και μίας πρόσθετης λειτουργίας XOR. Εφαρμόζοντας αυτήν την προσέγγιση υποδιασωλήνωσης και βελτιστοποιώντας την κρίσιμη διαδρομή, μπορούμε να μειώσουμε σημαντικά τη συνολική καθυστέρηση και να βελτιώσουμε τη συχνότητα ρολογιού στην ο ποία μπορεί να λειτουργήσει η συνάρτηση κατακερματισμού. Αυτή η βελτίωση οδηγεί σε μια πιο αποτελεσματική και υψηλής απόδοσης υλοποίηση της συνάρτησης κατακερματισμού Keccak στην προτεινόμενη αρχιτεκτονική μας.

Η υλοποίηση της διασωλήνωσης και του ξετυλίγματος της διαδικασίας κατακερματισμού αυξάνει την παραλληλία στην επεξεργασία, επιτρέποντας την ταυτόχρονη εκτέλεση πολλαπλών βημάτων. Αυτό μειώνει τον συνολικό χρόνο απαιτούμενο για την ολοκλήρωση ενός κύκλου κατακερματισμού, προσφέροντας σημαντικά οφέλη σε εφαρμογές που απαιτούν υψηλής ταχύτητας επεξεργασία δεδομένων.

Μέσω αυτής της μεθόδου, δίνεται έμφαση στην εξοικονόμηση πόρων μέσω της αποφυγής της υπερβολικής χρήσης λογικών πυλών για την υλοποίηση των συναρτήσεων XOR και AND, αξιοποιώντας έτσι με τον καλύτερο δυνατό τρόπο τη διαθέσιμη χωρητικότητα των FPGA συσκευών. Η αποδοτικότητα αυτής της προσέγγισης καθίσταται εξαιρετικά σημαντική σε περιβάλλοντα όπου οι χρονικοί και υλικοτεχνικοί πόροι είναι περιορισμένοι, επιτρέποντας την ανάπτυξη πιο σύνθετων και απαιτητικών εφαρμογών χωρίς συμβιβασμούς στην απόδοση.

Πίνακας 6.3: Τα αποτελέσματα της βελτιστοποίησης διοχετεύσεων και ξετυλίγματος υλικού στις συσκευές FPGA (Virtex-5, Virtex-6, και Virtex-7).

| **Μετρικές** | **Μέγεθος μπλοκ $r$** | **Virtex-5** | **Virtex-6** | **Virtex-7** |
|---|---|---|---|---|
| **Συχνότητα (MHz)** | | 272,41 | 344,62 | 396,28 |
| **Επιφάνεια (slices)** | | 1186 | 1348 | 1452 |
| **Ρυθμαπόδοση (Gbps)** | $r$=1152 | 26,151 | 33,084 | 38,043 |
| | $r$=1088 | 24,699 | 31,246 | 35,929 |
| | $r$=832 | 18,887 | 23,894 | 27,475 |
| | $r$=576 | 13,076 | 16,542 | 19,021 |
| **Αποδοτικότητα (Gbps/slices)** | $r$=1152 | 22,05 | 24,54 | 26,20 |
| | $r$=1088 | 20,83 | 23,18 | 24,74 |
| | $r$=832 | 15,93 | 17,73 | 18,92 |
| | $r$=576 | 11,03 | 12,27 | 13,10 |

Συμπερασματα απο την τεχνική διοχετεύσεων και ξετυλίγματος υλικού:

1. Αύξηση της συχνότητας λειτουργίας: Υπάρχει αύξηση της συχνότητας λειτουργίας μεταξύ των διάφορων μοντέλων FPGA, με το Virtex-7 να εμφανίζει την υψηλότερη συχνότητα. Αυτό υποδηλώνει την αποτελεσματικότητα της βελτιστοποίησης στην επίτευξη υψηλότερων επιδόσεων σε πιο πρόσφατα και προηγμένα μοντέλα FPGA.

2. Χρήση πόρων: Ο αριθμός των slices που απαιτούνται για την υλοποίηση της βελτιστοποίησης αυξάνεται επίσης με την εξέλιξη των μοντέλων FPGA, δείχνοντας ότι οι πιο πρόσφατες συσκευές επιτρέπουν πιο περίπλοκες υλοποιήσεις, λόγω της μεγαλύτερης διαθέσιμης χωρητικότητας και της βελτιωμένης αρχιτεκτονικής τους.

3. Αύξηση ρυθμαπόδοσης: Η ρυθμαπόδοση αυξάνεται σημαντικά σε κάθε μοντέλο FPGA, καθώς αυξάνεται το μέγεθος του μπλοκ $r$. Αυτό υποδηλώνει ότι η βελτιστοποίηση διοχετεύσεων και ξετυλίγματος βελτιώνει την ικανότητα των συσκευών να επεξεργάζονται μεγαλύτερους όγκους δεδομένων σε δεδομένο χρονικό διάστημα, προσφέροντας μεγάλα οφέλη σε εφαρμογές που απαιτούν υψηλή ταχύτητα επεξεργασίας.

4. Βελτίωση της αποδοτικότητας: Η αποδοτικότητα, δείχνει θετική τάση βελτίωσης καθώς μεταβαίνουμε από το Virtex-5 στο Virtex-7, υποδηλώνοντας ότι η πιο πρόσφατη τεχνολογία FPGA επιτρέπει πιο αποδοτική χρήση των πόρων για την επίτευξη υψηλότερων ταχυτήτων επεξεργασίας. Αυτό είναι ιδιαίτερα σημαντικό σε σενάρια όπου οι περιορισμένοι πόροι απαιτούν την μέγιστη δυνατή αποδοτικότητα.

5. Επιδράσεις σε διάφορα μεγέθη μπλοκ $r$: Η απόδοση αυξάνεται με την αύξηση του μεγέθους του μπλοκ $r$, επιβεβαιώνοντας ότι η βελτιστοποίηση παρέχει συνεχή οφέλη ακόμα και καθώς οι απαιτήσεις δεδομένων αυξάνονται. Αυτό υπογραμμίζει την κλιμακωσιμότητα της προτεινόμενης αρχιτεκτονικής και την καταλληλότητά της για εφαρμογές με μεγάλες απαιτήσεις δεδομένων.

Η βελτιστοποίηση μέσω τεχνικών διασωλήνωσης και ξετυλίγματος προσφέρει ουσιαστικά οφέλη στη ρυθμαπόδοση και την αποδοτικότητα των συσκευών FPGA, ενισχύοντας την ικανότητά τους να εκτελούν περίπλοκες διεργασίες δεδομένων με πολλαπλάσια επιτάχυνση. Η συνεχής αύξηση της συχνότητας λειτουργίας, σε συνδυασμό με τη βελτίωση της ρυθμαπόδοσης και της ενεργειακής αποδοτικότητας σε διαδοχικές γενιές συσκευών FPGA, αναδεικνύει την καθοριστική συμβολή των

εν λόγω τεχνικών στη σχεδίαση και ανάπτυξη προηγμένων ψηφιακών συστημάτων υψηλών επιδόσεων. Η προτεινόμενη αρχιτεκτονική καθίσταται ιδιαίτερα σημαντική σε περιβάλλοντα όπου η επεξεργασία μεγάλων όγκων δεδομένων και η επίτευξη υψηλής απόδοσης αποτελούν κρίσιμες απαιτήσεις, όπως στην επεξεργασία σήματος, την κρυπτογραφία και τις εφαρμογές πραγματικού χρόνου.

Περαιτέρω, οι τεχνικές αυτές δεν περιορίζονται μόνο στη βελτίωση της ρυθμαπόδοσης, αλλά επιτυγχάνουν και σημαντική επιτάχυνση της εκτέλεσης του αλγορίθμου, μειώνοντας τον χρόνο επεξεργασίας και βελτιστοποιώντας τη χρήση των διαθέσιμων πόρων. Η αύξηση της αποδοτικότητας καθιστά δυνατή την επίτευξη ίσων ή και ανώτερων επιπέδων απόδοσης με τη χρήση μικρότερου αριθμού slices, γεγονός που επιτρέπει την επιλογή μικρότερων ή οικονομικότερων μοντέλων FPGA. Με αυτόν τον τρόπο, μειώνεται το συνολικό κόστος υλοποίησης χωρίς να θυσιάζεται η απόδοση, ενώ ταυτόχρονα εξασφαλίζεται η απαραίτητη επιτάχυνση για απαιτητικές εφαρμογές σε πραγματικά υπολογιστικά περιβάλλοντα.

## 6.4 Συμπεράσματα από τις τρεις τεχνικές βελτιστοποίησης σε συσκευές FPGA

Από τις τρεις τεχνικές βελτιστοποίησης σε συσκευές FPGA (Virtex-5, Virtex-6, και Virtex-7) προκύπτουν τα εξης συμπεράσματα:

1. Βελτίωση στη συχνότητα λειτουργίας: Οι τεχνικές βελτιστοποίησης καταφέρνουν να αυξήσουν τη συχνότητα λειτουργίας σε όλες τις συσκευές FPGA, με την δεύτερη προτεινόμενη τεχνική (διασωλήνωση μετά το βήμα $\theta$) να προσφέρει την υψηλότερη συχνότητα σε σχέση με την πρώτη προτεινόμενη τεχνική (διασωλήνωση μετά το βήμα $\pi$) και τις τεχνικες ξετυλίγματος.

2. Διαφορές στην κατανάλωση πόρων: Η δεύτερη τεχνική βελτιστοποίησης διασωλήνωσης μετά το βήμα $\theta$ δείχνει απαιτεί λιγότερους πόρους σε σχέση με την πρώτη προτεινόμενη τεχνική (διασωλήνωση μετά το βήμα $\pi$), υποδηλώνοντας μια πιο αποδοτική χρήση των slices. Η τεχνική ξετυλίγματος έχει την μεγαλύτερη απαίτηση σε πόρους.

3. Αύξηση ρυθμαπόδοσης: Και οι τρεις τεχνικές βελτιστοποίησης βελτιώνουν σημαντικά την ρυθμαπόδοση σε όλες τις συσκευές, με την τεχνική

διοχετεύσεων και ξετυλίγματος υλικού να παρέχει την καλύτερη απόδοση. Αυτό υποδηλώνει τη σημασία της επιλογής του σωστού σημείου για την τοποθέτηση του αγωγού στην αρχιτεκτονική για την μεγιστοποίηση της αποδοτικότητας.

4. Βελτίωση στην αποδοτικότητα: Η τεχνική διοχετεύσεων και ξετυλίγματος υλικού επιτυγχάνει την υψηλότερη αποδοτικότητα (Mbps/slices) σε όλα τα μεγέθη μπλοκ $r$, ενώ η τεχνική ξετυλίγματος δείχνει επίσης σημαντική βελτίωση στην αποδοτικότητα σε σύγκριση με την πρώτη προτεινόμενη τεχνική διασωλήνωσης. Αυτό υπογραμμίζει πως η βελτιστοποίηση της θέσης του αγωγού και η εφαρμογή του ξετυλίγματος μπορούν να οδηγήσουν σε αποδοτικότερη χρήση των πόρων και καλύτερη απόδοση του αλγορίθμου.

5. Συνολικές επιδράσεις στην ρυθμαπόδοση και την αποδοτικότητα: Και οι τρεις τεχνικές βελτιστοποίησης παρέχουν σημαντικές βελτιώσεις στην ρυθμαπόδοση και την αποδοτικότητα των FPGA συσκευών, επιτρέποντας την ανάπτυξη πιο προηγμένων ψηφιακών συστημάτων. Η σωστή εφαρμογή και συνδυασμός αυτών των τεχνικών οδηγει στην μέγιστη βελτιστοποίηση τόσο της ρυθμαπόδοσης όσο και της αποδοτικότητας, ανάλογα με τις ειδικές απαιτήσεις και περιορισμούς του κάθε σχεδίου.

## 6.5 Απαντήσεις των ερευνητικών ερωτημάτων

1. Ποιες είναι οι διάφορες υλοποιήσεις του αλγορίθμου SHA-3 σε FPGA και πώς αξιολογούνται ως προς την απόδοση, την αρχιτεκτονική, την επιτάχυνση και τις προκλήσεις που αντιμετωπίζουν;

   Η ρυθμαπόδοση αποτελεί κρίσιμη μετρική αξιολόγησης, καθώς καθορίζει την ταχύτητα με την οποία ο αλγόριθμος μπορεί να εκτελεστεί και να επεξεργαστεί δεδομένα, ενώ η κατανάλωση πόρων εκφράζει τον βαθμό αποδοτικής χρήσης των διαθέσιμων στοιχείων της συσκευής FPGA. Η συνολική αποδοτικότητα συνδέεται με την ικανότητα της εκάστοτε υλοποίησης να επιτυγχάνει τη μέγιστη δυνατή απόδοση με την ελάχιστη κατανάλωση πόρων, εξασφαλίζοντας ισορροπία μεταξύ κόστους και επιδόσεων.

   Οι παραδοσιακές σειριακές υλοποιήσεις του SHA-3 είναι πιο απλές στον σχεδιασμό και ευκολότερες στην υλοποίηση σε FPGA. Παρότι δεν προσφέρουν την υψηλότερη δυνατή ρυθμαπόδοση, μπορούν να είναι αποδοτικές ως

προς την κατανάλωση πόρων και επαρκείς για εφαρμογές με περιορισμένες απαιτήσεις. Ωστόσο, οι υλοποιήσεις αυτές υστερούν ως προς την επιτάχυνση, γεγονός που τις καθιστά λιγότερο κατάλληλες για εφαρμογές υψηλών επιδόσεων.

Αντίθετα, οι στρατηγικές διασωλήνωσης, ξετυλίγματος και ο συνδυασμός τους (διασωλήνωση/ξετύλιγμα) αποτελούν καθοριστικούς μηχανισμούς για την επίτευξη σημαντικής επιτάχυνσης και βελτίωσης της συνολικής απόδοσης. Η διασωλήνωση μειώνει την κρίσιμη διαδρομή και επιτρέπει υψηλότερες συχνότητες λειτουργίας, ενώ το ξετύλιγμα αυξάνει τον παραλληλισμό εκτελώντας πολλαπλούς γύρους ανά κύκλο ρολογιού. Ο συνδυασμός των δύο τεχνικών μπορεί να προσφέρει τις μέγιστες δυνατότητες επιτάχυνσης, οδηγώντας σε υλοποιήσεις με εξαιρετικά υψηλή ρυθμαπόδοση και ταχύτερη εκτέλεση.

Η αρχιτεκτονική των υλοποιήσεων του SHA-3 σε FPGA οφείλει να ισορροπεί ανάμεσα στη ρυθμαπόδοση, την κατανάλωση πόρων και την αποδοτικότητα. Οι βασικές προκλήσεις περιλαμβάνουν την εξισορρόπηση μεταξύ ταχύτητας και πολυπλοκότητας, καθώς και τη βελτιστοποίηση της χρήσης των περιορισμένων πόρων.

Για εφαρμογές που απαιτούν τη μέγιστη δυνατή απόδοση και ελάχιστο χρόνο εκτέλεσης, οι τεχνικές διασωλήνωσης, ξετυλίγματος και ο συνδυασμός τους συνιστούν την καταλληλότερη επιλογή, παρά την αυξημένη κατανάλωση πόρων. Από την άλλη πλευρά, σε περιβάλλοντα με περιορισμένη διαθεσιμότητα πόρων, οι σειριακές υλοποιήσεις μπορούν να αποτελέσουν μια πιο οικονομική λύση, αν και υστερούν σε όρους επιτάχυνσης και ρυθμαπόδοσης.

2. Ποιες μέθοδοι και τεχνικές επιτάχυνσης μπορούν να βελτιώσουν σημαντικά την απόδοση του κρίσιμου μονοπατιού της συνάρτησης κατακερματισμού SHA-3 σε περιβάλλοντα FPGA;

   Η βελτίωση της απόδοσης του κρίσιμου μονοπατιού αποτελεί θεμελιώδη προϋπόθεση για την αύξηση της ρυθμαπόδοσης και της συνολικής αποδοτικότητας του συστήματος. Η εκμετάλλευση του ενδογενή παραλληλισμού του αλγορίθμου SHA-3, ειδικότερα σε υλοποιήσεις με FPGA, επιτρέπει την ταυτόχρονη επεξεργασία δεδομένων σε διαφορετικά στάδια του αλγορίθμου, γεγονός που οδηγεί σε ουσιαστική επιτάχυνση και σε μείωση του συνολικού χρόνου εκτέλεσης. Η ανάλυση και η στοχευμένη

βελτιστοποίηση της κρίσιμης διαδρομής, η οποία καθορίζει τη μέγιστη συχνότητα λειτουργίας του κυκλώματος, είναι κομβικής σημασίας, καθώς η μείωση της καθυστέρησης σε αυτό το σημείο αποδίδει άμεσα επιτάχυνση και βελτίωση της απόδοσης.

Αναλύοντας τις διάφορες προσεγγίσεις, μπορούμε να καταλήξουμε στα εξής συμπεράσματα:

Η διασωλήνωση είναι μία από τις πιο διαδεδομένες τεχνικές για την βελτίωση της απόδοσης σε FPGA. Αυτή η τεχνική αυξάνει την αποδοτικότητα της επεξεργασίας με τη διαίρεση του υπολογισμού σε διάφορα στάδια, επιτρέποντας την ταυτόχρονη επεξεργασία διαφορετικών δεδομένων. Κάθε στάδιο της διασωλήνωσης εκτελεί μέρος του υπολογισμού και στη συνέχεια περνά το αποτέλεσμα στο επόμενο στάδιο. Αυτό μειώνει τον συνολικό χρόνο απαιτούμενο για την επεξεργασία μιας σειράς δεδομένων, καθώς κάθε στοιχείο δεδομένων μετακινείται μέσω του αγωγου σε κάθε κύκλο ρολογιού.

Το ξετύλιγμα του βρόχου αυξάνει την ταχύτητα εκτέλεσης με τη μείωση του αριθμού των επαναλήψεων του βρόχου, εκτελώντας πολλαπλές εντολές του βρόχου ταυτόχρονα. Το ξετύλιγμα του βρόχου μειώνει τον αριθμό των επαναλήψεων ενός βρόχου αυξάνοντας τον αριθμό των εντολών εντός κάθε επανάληψης. Αυτό μειώνει τον συνολικό αριθμό των κύκλων ρολογιού απαιτούμενων για την εκτέλεση του αλγορίθμου, καθώς μειώνεται ο χρόνος που αφιερώνεται σε διαχειριστικές λειτουργίες του βρόχου, όπως η εκτίμηση της συνθήκης επανάληψης και η ενημέρωση των μεταβλητών ελέγχου. Κατά αυτόν τον τρόπο, η εκτέλεση του αλγορίθμου γίνεται πιο αποτελεσματική, καθώς το ξετύλιγμα επιτρέπει την παράλληλη επεξεργασία πολλαπλών εντολών του βρόχου σε κάθε επανάληψη.

Η ταυτόχρονη εφαρμογή των δύο τεχνικών μπορεί να προσφέρει έναν ιδιαίτερα αποδοτικό τρόπο για την επιτάχυνση της απόδοσης των κρυπτογραφικών αλγορίθμων σε περιβάλλοντα FPGA. Η διασωλήνωση μειώνει την καθυστέρηση επιτρέποντας την ταυτόχρονη επεξεργασία διαφορετικών δεδομένων, ενώ το ξετύλιγμα βελτιώνει τη απόδοση μέσω της μείωσης του αριθμού των κύκλων ρολογιού απαιτούμενων για την εκτέλεση του βρόχου.

3. Πώς συγκρίνονται οι νέες υλοποιήσεις του SHA-3 σε FPGA από άποψη ρυθμαπόδοσης, αποδοτικότητας και επιτάχυνσης με άλλες υφιστάμενες

λύσεις; Ποια είναι τα πλεονεκτήματα και οι περιορισμοί τους σε σχέση με τις υπάρχουσες υλοποιήσεις;

Η ανάλυση των νέων σχεδίων του SHA-3 σε FPGA από άποψη ρυθμαπόδοσης, επιτάχυνσης και αποδοτικότητας αποκαλύπτει μια ενδιαφέρουσα εξέλιξη στην τεχνολογία κρυπτογράφησης και στις υλοποιήσεις σε υλικό. Τα νέα σχέδια παρουσιάζουν βελτιωμένη ρυθμαπόδοση και σημαντική επιτάχυνση λόγω της εφαρμογής προηγμένων τεχνικών βελτιστοποίησης όπως η διασωλήνωση, το ξετύλιγμα του βρόχου και η παραλληλοποίηση. Αυτές οι τεχνικές επιτρέπουν την ταχύτερη επεξεργασία δεδομένων, μείωση της καθυστέρησης και επιτάχυνση της εκτέλεσης σε σύγκριση με παλαιότερες υλοποιήσεις. Οι νέες υλοποιήσεις τείνουν να είναι πιο αποτελεσματικές από πλευράς χρήσης πόρων, καθώς οι σύγχρονες τεχνολογίες και στρατηγικές βελτιστοποίησης μειώνουν την ανάγκη για μεγάλο αριθμό λογικών πυλών και μνήμης. Αυτό επιτρέπει την υλοποίηση πιο σύνθετων λειτουργιών σε μικρότερα και λιγότερο κοστοβόρα FPGA, επιτυγχάνοντας ταυτόχρονα επιτάχυνση στην εκτέλεση.

Τα νέα σχέδια προσφέρουν μια σειρά από σημαντικά πλεονεκτήματα που τα καθιστούν ιδιαίτερα ελκυστικά για μια ποικιλία εφαρμογών στον τομέα της κρυπτογραφίας και της ασφάλειας δεδομένων. Η αυξημένη ρυθμαπόδοση και η επιτάχυνση επιτρέπουν την ταχύτερη επεξεργασία δεδομένων, μειώνοντας τον χρόνο απόκρισης και ενισχύοντας την ικανότητα χειρισμού μεγάλων όγκων δεδομένων. Η μειωμένη κατανάλωση πόρων και η καλύτερη ενεργειακή αποδοτικότητα σημαίνουν ότι τα νέα σχέδια μπορούν να εκτελεστούν σε λιγότερο ισχυρό υλικό ή να επιτρέπουν την προσθήκη περισσότερων λειτουργιών σε έναν δεδομένο υλικοτεχνικό πόρο, προσφέροντας μια κόστος-αποδοτική λύση για πολλά συστήματα.

Η ευελιξία στον σχεδιασμό και η δυνατότητα για προσαρμοσμένες βελτιστοποιήσεις είναι επίσης σημαντικά πλεονεκτήματα για την επίτευξη υψηλής ρυθμαπόδοσης και επιτάχυνσης σε διάφορα σενάρια χρήσης, από τα οποία μπορεί να απαιτούνται διαφορετικές βελτιστοποιήσεις και προσαρμογές. Επιπλέον, αυτή η ευελιξία μπορεί να επιτρέψει την ενσωμάτωση του SHA-3 σε πιο πολύπλοκα συστήματα, όπου η ασφάλεια δεδομένων πρέπει να εξισορροπηθεί με άλλες λειτουργικές απαιτήσεις, όπως η ταχύτητα επεξεργασίας, η ενεργειακή αποδοτικότητα και η χρήση πόρων. Η δυνατότητα να προσαρμόζεται και να βελτιστοποιείται ο σχεδιασμός σύμφωνα με τις ειδικές ανάγκες κάθε εφαρμογής παρέχει μια ισχυρή βάση

για την ανάπτυξη ασφαλών, αποδοτικών και κόστους-αποδοτικών λύσεων με παράλληλη επιτάχυνση της επεξεργασίας.

Τέλος, η δυνατότητα για εξατομικευμένες λύσεις προσφέρει μια σημαντική αξία για την έρευνα και την ανάπτυξη στον τομέα της κρυπτογραφίας, καθώς επιτρέπει την εξερεύνηση νέων ιδεών και προσεγγίσεων με ταχύτερους ρυθμούς, μειωμένο κόστος και επιτυγχανόμενη επιτάχυνση των πειραματικών υλοποιήσεων. Αυτή η δυνατότητα για προσαρμογή δίνει την ευκαιρία για τη βελτίωση της αποδοτικότητας και την ενίσχυση της επιτάχυνσης των κρυπτογραφικών συστημάτων, ενδυναμώνοντας την προστασία των ψηφιακών δεδομένων στη σύγχρονη εποχή της πληροφορικής.

Παρόλα αυτά, οι νέες υλοποιήσεις μπορεί να αντιμετωπίζουν περιορισμούς. Οι απαιτήσεις για υψηλή ρυθμαπόδοση και επιτάχυνση συχνά οδηγούν σε αυξημένη πολυπλοκότητα σχεδιασμού. Η ανάπτυξη προηγμένων λύσεων μπορεί να αυξήσει το κόστος, τόσο σε όρους χρόνου όσο και πόρων, σε σύγκριση με πιο απλές και καθιερωμένες υλοποιήσεις. Επιπλέον, αν και η επιτάχυνση βελτιώνει την απόδοση, σε ορισμένες περιπτώσεις μπορεί να επιφέρει υψηλότερη ενεργειακή κατανάλωση. Η εξισορρόπηση μεταξύ επιτάχυνσης, ρυθμαπόδοσης και ενεργειακής αποδοτικότητας αποτελεί σημαντική πρόκληση, ιδίως για εφαρμογές που λειτουργούν σε περιβάλλοντα με περιορισμένη διαθεσιμότητα ενέργειας.

## 6.6 Προτάσεις για μελλοντική έρευνα

Η διαρκής πρόοδος στον τομέα της τεχνολογίας FPGA αποτελεί τον καταλύτη για την ανάπτυξη και εφαρμογή προηγμένων τεχνικών βελτιστοποίησης, που στοχεύουν στη μεγιστοποίηση της αποδοτικότητας και της ρυθμαπόδοσης των ψηφιακών συστημάτων. Η συνεχής ανάγκη για μείωση της κατανάλωσης ενέργειας και την επιτάχυνση των υπολογιστικών διεργασιών καθιστά την έρευνα σε νέες στρατηγικές βελτιστοποίησης όχι μόνο επιθυμητή, αλλά και απαραίτητη.

Οι μελλοντικές έρευνες θα πρέπει να επικεντρώνονται στην ανάπτυξη καινοτόμων τεχνικών, προκειμένου να διαχειριστούν αποτελεσματικά τους πόρους των FPGA και να βελτιστοποιήσουν την απόδοση των εφαρμογών. Η εξερεύνηση των δυνατοτήτων της αυτοματοποιημένης σχεδίασης και των ευφυών συστημάτων μπορεί να παρέχει

νέες λύσεις για την αντιμετώπιση των προκλήσεων που εμφανίζονται στην ψηφιακή επεξεργασία και την κρυπτογραφία.

Παράλληλα, η ανάπτυξη τεχνολογιών που στοχεύουν στην περαιτέρω μείωση της κατανάλωσης ενέργειας και στην αύξηση της ενεργειακής αποδοτικότητας των FPGA είναι κρίσιμη για την υλοποίηση βιώσιμων και περιβαλλοντικά φιλικών ψηφιακών λύσεων. Η αξιοποίηση προηγμένων τεχνικών για τη μείωση της κατανάλωσης ενέργειας χωρίς να επηρεάζεται αρνητικά η απόδοση, ανοίγει νέους δρόμους για την ανάπτυξη ενεργειακά αποδοτικών και υψηλής απόδοσης ψηφιακών συστημάτων.

Η βελτίωση της αποδοτικότητας στην επεξεργασία και στη διαχείριση των δεδομένων στον αλγόριθμο SHA-3 μπορεί επίσης να οδηγήσει σε μεγαλύτερη ευελιξία στην εφαρμογή του σε διάφορα συστήματα και πλατφόρμες, προσφέροντας έτσι μια πιο αποδοτική λύση σε ένα ευρύ φάσμα εφαρμογών. Ενθαρρύνενεται η περαιτέρω έρευνα και ανάπτυξη πάνω στις προτεινόμενες τεχνικές, με στόχο την αναζήτηση νέων τρόπων για την ακόμη πιο αποδοτική χρήση της δυναμικότητας των FPGA συσκευών. Η ενσωμάτωση προηγμένων τεχνικών σχεδίασης και βελτιστοποίησης στον αλγόριθμο SHA-3 ανοίγει νέους δρόμους για την επίτευξη υψηλότερων επιδόσεων σε πολλαπλές εφαρμογές ασφαλείας και κρυπτογραφίας, παρέχοντας έτσι μια σταθερή βάση για την αντιμετώπιση των αυξανόμενων απαιτήσεων σε επεξεργαστική ισχύ.

Επιπλέον, η δυνατότητα για την περαιτέρω μείωση του κόστους και της κατανάλωσης ενέργειας, μέσω της αύξησης της αποδοτικότητας των συσκευών FPGA, παραμένει ένας σημαντικός παράγοντας που πρέπει να ληφθεί υπόψη στις μελλοντικές έρευνες. Η έμφαση στην εξερεύνηση νέων τεχνικών και στρατηγικών που μπορούν να ενσωματωθούν στην αρχιτεκτονική των FPGA, ενδεχομένως να προσφέρει ακόμη μεγαλύτερες βελτιώσεις σε ρυθμαπόδοση και αποδοτικότητα, ανοίγοντας έτσι το δρόμο για την ανάπτυξη πιο προηγμένων και ενεργειακά αποδοτικών λύσεων κρυπτογραφίας.

Η επέκταση της προτεινόμενης μεθοδολογίας βελτιστοποίησης σε άλλες οικογένειες αλγορίθμων, όπως οι επαναληπτικοί block cipher αλγόριθμοι (π.χ., AES), ανοίγει νέους ορίζοντες για την ερευνητική δραστηριότητα στον τομέα της κρυπτογραφίας και της ασφάλειας των δεδομένων. Η δυνατότητα εφαρμογής των προτεινόμενων βελτιστοποιήσεων σε πληθώρα αλγορίθμων με παρόμοιες αρχιτεκτονικές και

απαιτήσεις επεξεργασίας επιτρέπει τη συστηματική μελέτη και αξιοποίηση της αποδοτικότητας και της ρυθμαπόδοσης σε διάφορες εφαρμογές.

Η διαδικασία εξερεύνησης των ιδιαιτεροτήτων και των προκλήσεων που συνδέονται με τους διάφορους αλγορίθμους επιτρέπει την κατανόηση των βασικών παραγόντων που επηρεάζουν την αποδοτικότητα και την ρυθμαπόδοση, προσφέροντας παράλληλα νέες προοπτικές για την επίλυση τεχνικών προβλημάτων. Η προσέγγιση αυτή μπορεί να οδηγήσει στην ανάπτυξη ενός ευρύτερου φάσματος εφαρμογών, βελτιώνοντας την οικονομικότητα και την ευελλιξία των συστημάτων. Παράλληλα, η επέκταση των τεχνικών βελτιστοποίησης σε άλλους τομείς εκτός της κρυπτογραφίας μπορεί να αποκαλύψει νέες εφαρμογές και προκλήσεις, ενισχύοντας την κατανόηση και την εφαρμογή των αλγορίθμων σε ποικίλους υπολογιστικούς τομείς.

Η δυνατότητα για εφαρμογή σε πολλούς τομείς και η τροποποίηση της προτεινόμενης μεθοδολογίας ενισχύει την ευελιξία και την προσαρμοστικότητα των συστημάτων, ανοίγοντας τον δρόμο για την ανάπτυξη πιο αποδοτικών λύσεων στην επεξεργασία δεδομένων και στην κρυπτογραφία. Η εξερεύνηση και η αξιοποίηση νέων τεχνικών και μεθοδολογιών βελτιστοποίησης ανοίγει νέους ορίζοντες για την επιστημονική έρευνα και την τεχνολογική ανάπτυξη, προάγοντας την καινοτομία και την πρόοδο σε παγκόσμιο επίπεδο. Τέλος, η έμφαση στη συνεχή βελτίωση και επέκταση των υπάρχουσων τεχνολογιών αποτελεί κεντρικό στόχο για την επίτευξη υψηλότερων επιδόσεων, συμβάλλοντας στην ανάπτυξη πιο αποδοτικών και οικονομικά βιώσιμων τεχνολογικών εφαρμογών.

# Βιβλιογραφία


[1] Monica Borda and Monica Borda. Cryptography Basics. *Fundamentals in Information Theory and Coding*, pages 121–208, 2011.

[2] Robert G Underwood. Introduction to Cryptography. In *Cryptography for Secure Encryption*, pages 1–8. Springer, 2022.

[3] Rajkumar Banoth and Rekha Regar. An Introduction to Classical and Modern Cryptography. In *Classical and Modern Cryptography for Beginners*, pages 1–46. Springer, 2023.

[4] John F Dooley. *A brief history of cryptology and cryptographic algorithms*. Springer, 2013.

[5] William Easttom and William Easttom. History of Cryptography to the 1800s. *Modern Cryptography: Applied Mathematics for Encryption and Information Security*, pages 1–26, 2021.

[6] Sirapat Boonkrong and Sirapat Boonkrong. Introduction to Cryptography. *Authentication and Access Control: Practical Cryptography Methods and Tools*, pages 1–30, 2021.

[7] Dyala R Ibrahim, Je Sen Teh, and Rosni Abdullah. An overview of visual cryptography techniques. *Multimedia Tools and Applications*, 80:31927–31952, 2021.

[8] Ajay Kumar and Sunita Garhwal. State-of-the-art survey of quantum cryptography. *Archives of Computational Methods in Engineering*, 28:3831–3868, 2021.

[9] Johannes Buchmann. *Introduction to cryptography*, volume 335. Springer, 2004.

[10] Ralph C Merkle. A certified digital signature. In *Conference on the Theory and Application of Cryptology*, pages 218–238. Springer, 1989.

[11] Whitfield Diffie and Martin E Hellman. Privacy and Authentication: An Introduction to Cryptography. In *Democratizing Cryptography: The Work of Whitfield Diffie and Martin Hellman*, pages 431–514. ACM New York, NY, 2022.

[12] Jonathan Katz. *Digital signatures*, volume 1. Springer, 2010.

[13] Khaled Salah Mohamed and Khaled Salah Mohamed. Cryptography concepts: Confidentiality. *New Frontiers in Cryptography: Quantum, Blockchain, Lightweight, Chaotic and DNA*, pages 13–39, 2020.

[14] Faisal Abbasi and Pawan Singh. Cryptography: Security and Integrity of Data Management. *Journal of Management and Service Science (JMSS)*, 1(2):1–9, 2021.

[15] Amir Jalaly Bidgoly, Hamed Jalaly Bidgoly, and Zeynab Arezoumand. A survey on methods and challenges in EEG based authentication. *Computers & Security*, 93:101788, 2020.

[16] Sirapat Boonkrong. *Authentication and access control: practical cryptography methods and tools*. Springer, 2021.

[17] Moustafa Mamdouh, Ali Ismail Awad, Ashraf AM Khalaf, and Hesham FA Hamed. Authentication and identity management of IoHT devices: achievements, challenges, and future directions. *Computers & Security*, 111: 102491, 2021.

[18] Ning Xie, Zhuoyuan Li, and Haijun Tan. A survey of physical-layer authentication in wireless communications. *IEEE Communications Surveys & Tutorials*, 23(1):282–310, 2020.

[19] Jin Sun, Xiaomin Yao, Shangping Wang, and Ying Wu. Non-repudiation storage and access control scheme of insurance data based on blockchain in IPFS. *IEEE Access*, 8:155145–155155, 2020.

[20] Fei Chen, Jiahao Wang, Jianqiang Li, Yang Xu, Cheng Zhang, and Tao Xiang. TrustBuilder: A non-repudiation scheme for IoT cloud applications. *Computers & Security*, 116:102664, 2022.

[21] Ronald L Rivest. MD4 message digest algorithm. Technical report, MIT Laboratory for Computer Science, 1990.

[22] Ronald Rivest. The MD5 message-digest algorithm. Technical report, MIT Laboratory for Computer Science, 1992.

[23] Xiaoyun Wang and Hongbo Yu. How to break MD5 and other hash functions. In *Annual international conference on the theory and applications of cryptographic techniques*, pages 19–35. Springer, 2005.

[24] Xiaoyun Wang, Yiqun Lisa Yin, and Hongbo Yu. Finding collisions in the full SHA-1. In *Advances in Cryptology–CRYPTO 2005: 25th Annual International Cryptology Conference, Santa Barbara, California, USA, August 14-18, 2005. Proceedings 25*, pages 17–36. Springer, 2005.

[25] Marc Stevens, Elie Bursztein, Pierre Karpman, Ange Albertini, and Yarik Markov. The first collision for full SHA-1. In *Advances in Cryptology–CRYPTO 2017: 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20–24, 2017, Proceedings, Part I 37*, pages 570–596. Springer, 2017.

[26] Raffaele Martino and Alessandro Cilardo. SHA-2 acceleration meeting the needs of emerging applications: A comparative survey. *IEEE Access*, 8:28415–28436, 2020.

[27] Yang Jun, Ding Jun, Li Na, and Guo Yixiong. FPGA implementation of SHA-224/256 algorithm oriented Digital Signature. In *2010 International Conference on Challenges in Environmental Science and Computer Engineering*, volume 2, pages 63–66. IEEE, 2010.

[28] Quynh Dang. Changes in federal information processing standard (FIPS) 180-4, secure hash standard. *Cryptologia*, 37(1):69–73, 2013.

[29] Hans Delfs, Helmut Knebl, and Helmut Knebl. *Introduction to cryptography*, volume 2. Springer, 2002.

[30] Morris J Dworkin. SHA-3 standard: Permutation-based hash and extendable-output functions. Technical report, National Institute of Standards and Technology (NIST), 2015.

[31] Shu-jen Chang, Ray Perlner, William E Burr, Meltem Sönmez Turan, John M Kelsey, Souradyuti Paul, and Lawrence E Bassham. Third-round report of the SHA-3 cryptographic hash algorithm competition. *NIST Interagency Report*, 7896:121, 2012.

[32] Salwa M Serag Eldin, Ahmed Sedik, Sultan S Alshamrani, and Ahmed M Ayoup. Cancellable Multi-Biometric Feature Veins Template Generation Based on SHA-3 Hashing. *Computers, Materials & Continua*, 75(1), 2023.

[33] Christof Paar and Jan Pelzl. *Understanding cryptography: a textbook for students and practitioners*. Springer Science & Business Media, 2009.

[34] Itai Dinur, Orr Dunkelman, and Adi Shamir. Improved practical attacks on round-reduced Keccak. *Journal of cryptology*, 27:183–209, 2014.

[35] Sandip Ghoshal, Pradosh Bandyopadhyay, Surojit Roy, and Monalisa Baneree. A journey from md5 to sha-3. In *Trends in Communication, Cloud, and Big Data: Proceedings of 3rd National Conference on CCB, 2018*, pages 107–112. Springer, 2020.

[36] Chin-Chen Chang and Ya-Fen Chang. Signing a digital signature without using one-way hash functions and message redundancy schemes. *IEEE Communications Letters*, 8(8):485–487, 2004.

[37] Shai Halevi and Hugo Krawczyk. Strengthening digital signatures via randomized hashing. In *Annual International Cryptology Conference*, pages 41–59. Springer, 2006.

[38] Moni Naor and Moti Yung. Universal one-way hash functions and their cryptographic applications. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, pages 33–43, 1989.

[39] Kannan Balasubramanian. Hash functions and their applications. In *Algorithmic Strategies for Solving Complex Problems in Cryptography*, pages 66–77. IGI Global, 2018.

[40] Dexi Wang, Yu Jiang, Houbing Song, Fei He, Ming Gu, and Jiaguang Sun. Verification of implementations of cryptographic hash functions. *IEEE Access*, 5:7816–7825, 2017.

[41] Jonathan Katz and Andrew Y Lindell. Aggregate message authentication codes. In *Cryptographers' Track at the RSA Conference*, pages 155–169. Springer, 2008.

[42] Bart Preneel and Paul C Van Oorschot. On the security of iterated message authentication codes. *IEEE Transactions on Information theory*, 45(1):188–199, 1999.

[43] P Karthik, P Shanthibala, Akashdeep Bhardwaj, Salil Bharany, Heejung Yu, and Yousaf Bin Zikria. A novel subset-based polynomial design for enhancing the security of short message-digest with inflated avalanche and random

responses. *Journal of King Saud University-Computer and Information Sciences*, 35(1):310–323, 2023.

[44] Leila Megouache, Abdelhafid Zitouni, and Mahieddine Djoudi. Ensuring user authentication and data integrity in multi-cloud environment. *Human-centric Computing and information sciences*, 10:1–20, 2020.

[45] Tarak Nandy, Mohd Yamani Idna Bin Idris, Rafidah Md Noor, Laiha Mat Kiah, Lau Sian Lun, Nor Badrul Annuar Juma'at, Ismail Ahmedy, Norjihan Abdul Ghani, and Sananda Bhattacharyya. Review on security of internet of things authentication mechanism. *IEEE Access*, 7:151054–151089, 2019.

[46] Joobeom Yun, Fayozbek Rustamov, Juhwan Kim, and Youngjoo Shin. Fuzzing of Embedded Systems: A Survey. *ACM Computing Surveys*, 55(7):1–33, 2022.

[47] Charalampos Manifavas, George Hatzivasilis, Konstantinos Fysarakis, and Yannis Papaefstathiou. A survey of lightweight stream ciphers for embedded systems. *Security and Communication Networks*, 9(10):1226–1246, 2016.

[48] Alexander Barkalov, Larysa Titarenko, and Małgorzata Mazurkiewicz. *Foundations of embedded systems*, volume 195. Springer, 2019.

[49] William A Johnson, Sheikh Ghafoor, and Stacy Prowell. A taxonomy and review of remote attestation schemes in embedded systems. *IEEE Access*, 9: 142390–142410, 2021.

[50] Tanmaya Mishra, Thidapat Chantem, and Ryan Gerdes. Survey of control-flow integrity techniques for real-time embedded systems. *ACM Transactions on Embedded Computing Systems (TECS)*, 21(4):1–32, 2022.

[51] Xinwen Li, Li Sun, and Christine A Rochester. Embedded system and smart embedded wearable devices promote youth sports health. *Microprocessors and Microsystems*, 83:104019, 2021.

[52] Peter Marwedel. *Embedded system design: embedded systems foundations of cyber-physical systems, and the internet of things*. Springer Nature, 2021.

[53] Natassya BF Silva, Daniel F Pigatto, Paulo S Martins, and Kalinka RLJC Branco. Case studies of performance evaluation of cryptographic algorithms for an embedded system and a general purpose computer. *Journal of Network and Computer Applications*, 60:130–143, 2016.

[54] Tim Güneysu, Vadim Lyubashevsky, and Thomas Pöppelmann. Practical lattice-based cryptography: A signature scheme for embedded systems. In *Cryptographic Hardware and Embedded Systems–CHES 2012: 14th International Workshop, Leuven, Belgium, September 9-12, 2012. Proceedings 14*, pages 530–547. Springer, 2012.

[55] Stefan Mangard, François-Xavier Standaert, et al. *Cryptographic Hardware and Embedded Systems: CHES 2010*. Springer, 2010.

[56] Prateek Sikka, Abhijit R Asati, and Chandra Shekhar. Speed optimal FPGA implementation of the encryption algorithms for telecom applications. *Microprocessors and Microsystems*, 79:103324, 2020.

[57] Guido Marco Bertoni and Jean-Sébastien Coron. *Cryptographic Hardware and Embedded Systems-CHES 2013*. Springer, 2013.

[58] Jean-Philippe Aumasson, Antony Vennard, and AG Teserakt. Cryptography in industrial embedded systems: our experience of needs and constraints. In *NIST Lightweight Cryptography Workshop*, 2019.

[59] David D Hwang, Patrick Schaumont, Kris Tiri, and Ingrid Verbauwhede. Securing embedded systems. *IEEE Security & Privacy*, 4(02):40–49, 2006.

[60] Rajdeep Chakraborty, Uttam Kr Mondal, Asish Debnath, Utpal Ghosh, and Bibek Bikash Roy. Lightweight micro-architecture for IoT & FPGA security. *International Journal of Information Technology*, 15(7):3899–3905, 2023.

[61] Arkan Alkamil and Darshika G Perera. Towards dynamic and partial reconfigurable hardware architectures for cryptographic algorithms on embedded devices. *IEEE Access*, 8:221720–221742, 2020.

[62] Mohamed Maazouz, Abdelmoughni Toubal, Billel Bengherbia, Oussama Houhou, and Noureddine Batel. FPGA implementation of a chaos-based image encryption algorithm. *Journal of King Saud University-Computer and Information Sciences*, 34(10):9926–9941, 2022.

[63] Miguel Morales-Sandoval, Luis Armando Rodriguez Flores, Rene Cumplido, Jose Juan Garcia-Hernandez, Claudia Feregrino, and Ignacio Algredo. A compact fpga-based accelerator for curve-based cryptography in wireless sensor networks. *Journal of Sensors*, 2021:1–13, 2021.

[64] Sa'ed Abed, Reem Jaffal, Bassam J Mohd, and Mohammad Al-Shayeji. An analysis and evaluation of lightweight hash functions for blockchain-based IoT devices. *Cluster Computing*, 24:3065–3084, 2021.

[65] Mohammed Omar Awadh Al-Shatari, Fawnizu Azmadi Hussin, Azrina Abd Aziz, Gunawan Witjaksono, and Xuan-Tu Tran. FPGA-based lightweight hardware architecture of the PHOTON hash function for IoT edge devices. *IEEE Access*, 8:207610–207618, 2020.

[66] Gokulnath Rajendran, Writam Banerjee, Anupam Chattopadhyay, and Mohamed M Sabry Aly. Application of resistive random access memory in hardware security: A review. *Advanced Electronic Materials*, 7(12):2100536, 2021.

[67] Arpan Jati, Naina Gupta, Anupam Chattopadhyay, and Somitra Kumar Sanadhya. A configurable crystals-kyber hardware implementation with side-channel protection. *ACM Transactions on Embedded Computing Systems*, 2021.

[68] Nabihah Ahmad and SM Rezaul Hasan. A new ASIC implementation of an advanced encryption standard (AES) crypto-hardware accelerator. *Microelectronics Journal*, 117:105255, 2021.

[69] Argirios Sideris, Theodora Sanida, and Minas Dasygenis. Hardware acceleration of SHA-256 algorithm using NIOS-II processor. In *2019 8th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, pages 1–4. IEEE, 2019.

[70] Theodora Sanida, Argyrios Sideris, and Minas Dasygenis. Accelerating the AES algorithm using opencl. In *2020 9th International conference on modern circuits and systems technologies (MOCAST)*, pages 1–4. IEEE, 2020.

[71] Simon Friedmann, Johannes Schemmel, Andreas Grübl, Andreas Hartel, Matthias Hock, and Karlheinz Meier. Demonstrating hybrid learning in a flexible neuromorphic hardware system. *IEEE transactions on biomedical circuits and systems*, 11(1):128–142, 2016.

[72] Tran Sy Nam, Hoang Van Thuc, and Bui Duy Hieu. A Hardware Architecture of NIST Lightweight Cryptography applied in IPSec to Secure High-throughput Low-latency IoT Networks. *IEEE Access*, 2023.

[73] Johannes Buchmann, Evangelos Karatsiolis, Alexander Wiesmaier, and Evangelos Karatsiolis. *Introduction to public key infrastructures*, volume 36. Springer, 2013.

[74] Pita Jarupunphol and Wipawan Buathong. Secure Electronic Transactions (SET): A case of secure system project failures. *International Journal of Engineering and Technology*, 5(2):278, 2013.

[75] Behnam Kamali, Robert Alexander Bennett, and Dyani Camika Cox. Understanding WiMAX: An IEEE-802.16 standard-based wireless technology. *IEEE Potentials*, 31(5):23–27, 2012.

[76] Hugo Krawczyk. LFSR-based hashing and authentication. In *Annual International Cryptology Conference*, pages 129–139. Springer, 1994.

[77] Marc Fischlin, Anja Lehmann, and Daniel Wagner. Hash function combiners in TLS and SSL. In *Cryptographers' Track at the RSA Conference*, pages 268–283. Springer, 2010.

[78] Mohamed Khalil-Hani, Vishnu P Nambiar, and MN Marsono. Hardware Acceleration of OpenSSL cryptographic functions for high-performance Internet Security. In *2010 International Conference on Intelligent Systems, Modelling and Simulation*, pages 374–379. IEEE, 2010.

[79] Wassim El-Hajj. The most recent SSL security attacks: origins, implementation, evaluation, and suggested countermeasures. *Security and Communication Networks*, 5(1):113–124, 2012.

[80] Tandoh Lawrence, Fagen Li, Ikram Ali, Charles R Haruna, Michael Y Kpiebaareh, and Tandoh Christopher. A computationally efficient HMAC-based authentication scheme for network coding. *Telecommunication Systems*, pages 1–23, 2022.

[81] Robert Walther, Carsten Weinhold, and Michael Roitzsch. RATLS: Integrating Transport Layer Security with Remote Attestation. In *International Conference on Applied Cryptography and Network Security*, pages 361–379. Springer, 2022.

[82] Hojin Choi and Seog Chung Seo. Optimization of PBKDF2 using HMAC-SHA2 and HMAC-LSH families in CPU environment. *IEEE Access*, 9:40165–40177, 2021.

[83] Francesc Sebé, Josep M Miret, Jordi Pujolàs, and Jordi Puiggalí. Simple and efficient hash-based verifiable mixing for remote electronic voting. *Computer communications*, 33(6):667–675, 2010.

[84] Yijun Yang, Fei Chen, Xiaomei Zhang, Jianping Yu, and Peng Zhang. Research on the hash function structures and its application. *Wireless Personal Communications*, 94:2969–2985, 2017.

[85] Hala Hassan, Ali Ibrahim El-Desouky, Abdelhameed Ibrahim, El-Sayed M El-Kenawy, and Reham Arnous. Enhanced QoS-based model for trust assessment in cloud computing environment. *IEEE Access*, 8:43752–43763, 2020.

[86] Vu Khanh Quy, Vi Hoai Nam, Dao Manh Linh, Nguyen Tien Ban, and Nguyen Dinh Han. A survey of QoS-aware routing protocols for the MANET-WSN convergence scenarios in IoT networks. *Wireless Personal Communications*, 120(1):49–62, 2021.

[87] Di Wu, Xin Luo, Mingsheng Shang, Yi He, Guoyin Wang, and Xindong Wu. A data-characteristic-aware latent factor model for web services QoS prediction. *IEEE Transactions on Knowledge and Data Engineering*, 34(6):2525–2538, 2020.

[88] Renzo Andri, Lukas Cavigelli, Davide Rossi, and Luca Benini. YodaNN: An ultra-low power convolutional neural network accelerator based on binary weights. In *2016 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, pages 236–241. IEEE, 2016.

[89] Safiullah Khan, Wai-Kong Lee, and Seong Oun Hwang. A flexible Gimli hardware implementation in FPGA and its application to RFID authentication protocols. *IEEE Access*, 9:105327–105340, 2021.

[90] Arish Sateesan, Jelle Biesmans, Thomas Claesen, Jo Vliegen, and Nele Mentens. Optimized algorithms and architectures for fast non-cryptographic hash functions in hardware. *Microprocessors and Microsystems*, 98:104782, 2023.

[91] Ata Elahi and Alex Cushman. Introduction to Cryptography. In *Computer Networks: Data Communications, Internet and Security*, pages 293–322. Springer, 2023.

[92] Robert G Underwood et al. *Cryptography for Secure Encryption*. Springer, 2022.

[93] Marc STEVENS. Cryptanalysis of SHA-1. *Symmetric Cryptography, Volume 2: Cryptanalysis and Future Directions*, page 181, 2024.

[94] Gurvir Kaur, Kuldeepak Singh, and Harsimranjit Singh Gill. Chaos-based joint speech encryption scheme using SHA-1. *Multimedia tools and applications*, 80: 10927–10947, 2021.

[95] Zeyad A Al-Odat, Samee U Khan, and Eman Al-Qtiemat. A modified secure hash design to circumvent collision and length extension attacks. *Journal of Information Security and Applications*, 71:103376, 2022.

[96] Richard H Preston. Applying Grover's Algorithm to Hash Functions: A Software Perspective. *IEEE Transactions on Quantum Engineering*, 3:1–10, 2022.

[97] Nada Lachtar, Abdulrahman Abu Elkhail, Anys Bacha, and Hafiz Malik. A cross-stack approach towards defending against cryptojacking. *IEEE Computer Architecture Letters*, 19(2):126–129, 2020.

[98] Hoai Luan Pham, Thi Hong Tran, Tri Dung Phan, Vu Trung Duong Le, Duc Khai Lam, and Yasuhiko Nakashima. Double SHA-256 hardware architecture with compact message expander for bitcoin mining. *IEEE Access*, 8:139634–139646, 2020.

[99] L Shane John Paul, Carlton Gracias, Anurag Desai, V Thanikaiselvan, S Suba Shanthini, and Amirtharajan Rengarajan. A novel colour image encryption scheme using dynamic DNA coding, chaotic maps, and SHA-2. *Multimedia Tools and Applications*, 81(26):37873–37894, 2022.

[100] FIPS Pub. Secure hash standard (shs). *Fips pub*, 180(4), 2012.

[101] Secure Hash Standard. Secure hash standard. *FIPS PUB*, pages 180–1, 1995.

[102] Ricardo Chaves, Leonel Sousa, Nicolas Sklavos, Apostolos P Fournaris, Georgina Kalogeridou, Paris Kitsos, and Farhana Sheikh. Secure hashing: Sha-1, sha-2, and sha-3. *Circuits and systems for security and privacy*, pages 105–132, 2016.

[103] Guido Bertoni, Joan Daemen, Michaël Peeters, and GV Assche. The keccak reference. *Submission to NIST (Round 3)*, 13:14–15, 2011.

[104] Abhilash Chakraborty, Anupam Biswas, and Ajoy Kumar Khan. SRIJAN: Secure Randomized Internally Joined Adjustable Network for one-way hashing. *Journal of Information Security and Applications*, 81:103717, 2024.

[105] Ahmed Maache and Abdesattar Kalache. Design and Implementation of a flexible Multi-purpose Cryptographic System on low cost FPGA. *International journal of electrical and computer engineering systems*, 14(1):45–58, 2023.

[106] Lu Li, Qi Tian, Guofeng Qin, Shuaiyu Chen, and Weijia Wang. Compact Instruction Set Extensions for Dilithium. *ACM Transactions on Embedded Computing Systems*, 2024.

[107] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. Cryptographic sponges. *online] http://sponge. noekeon. org*, 2011.

[108] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. The making of KECCAK. *Cryptologia*, 38(1):26–60, 2014.

[109] Dong-Chan Kim, Deukjo Hong, Jung-Keun Lee, Woo-Hwan Kim, and Daesung Kwon. LSH: A new fast secure hash function family. In *Information Security and Cryptology-ICISC 2014: 17th International Conference, Seoul, South Korea, December 3-5, 2014, Revised Selected Papers 17*, pages 286–313. Springer, 2015.

[110] María Naya-Plasencia, Andrea Röck, and Willi Meier. Practical analysis of reduced-round Keccak. In *International Conference on Cryptology in India*, pages 236–254. Springer, 2011.

[111] Ali Alzahrani and Fayez Gebali. Multi-core dataflow design and implementation of secure hash algorithm-3. *IEEE Access*, 6:6092–6102, 2018.

[112] John Kelsey, Shu-jen Chang, and Ray Perlner. SHA-3 derived functions: cSHAKE, KMAC, TupleHash, and ParallelHash. *NIST special publication*, 800: 185, 2016.

[113] Hemendra Rawat and Patrick Schaumont. Vector instruction set extensions for efficient computation of keccak. *IEEE Transactions on Computers*, 66(10): 1778–1789, 2017.

[114] Elena Andreeva, Bart Mennink, and Bart Preneel. Open problems in hash function security. *Designs, Codes and Cryptography*, 77:611–631, 2015.

[115] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. Keccak sponge function family main document. *Submission to NIST (Round 2)*, 3(30): 320–337, 2009.

[116] Itai Dinur, Paweł Morawiecki, Josef Pieprzyk, Marian Srebrny, and Michał Straus. Cube attacks and cube-attack-like cryptanalysis on the round-reduced Keccak sponge function. In *Advances in Cryptology–EUROCRYPT 2015: 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I 34*, pages 733–761. Springer, 2015.

[117] Ismail San and Nuray At. Compact Keccak hardware architecture for data integrity and authentication on FPGAs. *Information Security Journal: A Global Perspective*, 21(5):231–242, 2012.

[118] Stéphanie Kerckhof, François Durvaux, Nicolas Veyrat-Charvillon, Francesco Regazzoni, Guerric Meurice de Dormale, and François-Xavier Standaert. Compact FPGA implementations of the five SHA-3 finalists. In *Smart Card Research and Advanced Applications: 10th IFIP WG 8.8/11.2 International Conference, CARDIS 2011, Leuven, Belgium, September 14-16, 2011, Revised Selected Papers 10*, pages 217–233. Springer, 2011.

[119] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. Sponge-based pseudo-random number generators. In *Cryptographic Hardware and Embedded Systems, CHES 2010: 12th International Workshop, Santa Barbara, USA, August 17-20, 2010. Proceedings 12*, pages 33–47. Springer, 2010.

[120] Argyrios Sideris and Minas Dasygenis. Enhancing the Hardware Pipelining Optimization Technique of the SHA-3 via FPGA. *Computation*, 11(8):152, 2023.

[121] Ahmad Abusukhon, Zeyad Mohammad, and Ali Al-Thaher. An authenticated, secure, and mutable multiple-session-keys protocol based on elliptic curve cryptography and text-to-image encryption algorithm. *Concurrency and Computation: Practice and Experience*, 34(4):e6649, 2022.

[122] Tarunpreet Bhatia, Anil Kumar Verma, and Gaurav Sharma. Towards a secure incremental proxy re-encryption for e-healthcare data sharing in mobile cloud computing. *Concurrency and Computation: Practice and Experience*, 32(5):e5520, 2020.

[123] Xiaoxiao Chi, Chao Yan, Hao Wang, Wajid Rafique, and Lianyong Qi. Amplified locality-sensitive hashing-based recommender systems with privacy protection. *Concurrency and Computation: Practice and Experience*, 34(14):e5681, 2022.

[124] Hongsheng Hu, Gillian Dobbie, Zoran Salcic, Meng Liu, Jianbing Zhang, Lingjuan Lyu, and Xuyun Zhang. Differentially private locality sensitive hashing based federated recommender system. *Concurrency and Computation: Practice and Experience*, page e6233, 2021.

[125] Jiwoo Bang, Chungyong Kim, Eun-Kyu Byun, Hanul Sung, Jaehwan Lee, and Hyeonsang Eom. Accelerating I/O performance of ZFS-based Lustre file system in HPC environment. *The Journal of Supercomputing*, pages 1–27, 2022.

[126] Shi Zhang, Jin Huang, Ruliang Xiao, Xin Du, Ping Gong, and Xinhong Lin. Toward more efficient locality-sensitive hashing via constructing novel hash function cluster. *Concurrency and Computation: Practice and Experience*, 33(20): e6355, 2021.

[127] Ivica Nikolić and Alex Biryukov. Collisions for step-reduced SHA-256. In *International Workshop on Fast Software Encryption*, pages 1–15. Springer, 2008.

[128] Somitra Kumar Sanadhya and Palash Sarkar. New collision attacks against up to 24-step SHA-2. In *International conference on cryptology in India*, pages 91–103. Springer, 2008.

[129] Jian Guo, Guohong Liao, Guozhen Liu, Meicheng Liu, Kexin Qiao, and Ling Song. Practical collision attacks against round-reduced SHA-3. *Journal of Cryptology*, 33(1):228–270, 2020.

[130] Shunrong Jiang, Xiaoyan Zhu, and Liangmin Wang. An efficient anonymous batch authentication scheme based on HMAC for VANETs. *IEEE Transactions on Intelligent Transportation Systems*, 17(8):2193–2204, 2016.

[131] Jörg Schwenk. Attacks on SSL and TLS. In *Guide to Internet Cryptography: Security Protocols and Real-World Attack Implications*, pages 267–328. Springer, 2022.

[132] Hua-Lei Yin, Yao Fu, Chen-Long Li, Chen-Xun Weng, Bing-Hong Li, Jie Gu, Yu-Shuo Lu, Shan Huang, and Zeng-Bing Chen. Experimental quantum secure network with digital signatures and encryption. *National Science Review*, 10(4): nwac228, 2023.

[133] Rashidah Funke Olanrewaju, Burhan Ul Islam Khan, Mohd Mueen Ul Islam Mattoo, Farhat Anwar, Anis Nurashikin Bt Nordin, and Roohie Naaz Mir. Securing electronic transactions via payment gateways–a systematic review.

*International Journal of Internet Technology and Secured Transactions*, 7(3): 245–269, 2017.

[134] Terence Spies. Public key infrastructure. In *Computer and Information Security Handbook*, pages 691–711. Elsevier, 2017.

[135] David Goz, Georgios Ieronymakis, Vassilis Papaefstathiou, Nikolaos Dimou, Sara Bertocco, Francesco Simula, Antonio Ragagnin, Luca Tornatore, Igor Coretti, and Giuliano Taffoni. Performance and energy footprint assessment of FPGAs and GPUs on HPC systems using astrophysics application. *Computation*, 8(2):34, 2020.

[136] Juan Ruiz-Rosero, Gustavo Ramirez-Gonzalez, and Rahul Khanna. Field programmable gate array applications—A scientometric review. *Computation*, 7 (4):63, 2019.

[137] Fahad Siddiqui, Sam Amiri, Umar Ibrahim Minhas, Tiantai Deng, Roger Woods, Karen Rafferty, and Daniel Crookes. FPGA-Based Processor Acceleration for Image Processing Applications. *Journal of Imaging*, 5(1):16, 2019.

[138] Konstantinos Kalaitzis, Evripidis Sotiriadis, Ioannis Papaefstathiou, and Apostolos Dollas. Evaluation of external memory access performance on a High-End FPGA hybrid computer. *Computation*, 4(4):41, 2016.

[139] Argyrios Sideris, Theodora Sanida, Antonios Chatzisavvas, Michael Dossis, and Minas Dasygenis. High Throughput of Image Processing with Keccak Algorithm using Microprocessor on FPGA. In *2022 7th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM)*, pages 1–4. IEEE, 2022.

[140] Tan Nguyen, Colin MacLean, Marco Siracusa, Douglas Doerfler, Nicholas J Wright, and Samuel Williams. FPGA-based HPC accelerators: An evaluation on performance and energy efficiency. *Concurrency and Computation: Practice and Experience*, 34(20):e6570, 2022.

[141] Zeyad A Al-Odat, Mazhar Ali, Assad Abbas, and Samee U Khan. Secure hash algorithms and the corresponding fpga optimization techniques. *ACM Computing Surveys (CSUR)*, 53(5):1–36, 2020.

[142] Khai-Minh Ma, Duc-Hung Le, Cong-Kha Pham, and Trong-Thuc Hoang. Design of an SoC Based on 32-Bit RISC-V Processor with Low-Latency Lightweight Cryptographic Cores in FPGA. *Future Internet*, 15(5):186, 2023.

[143] Soufiane El Moumni, Mohamed Fettach, and Abderrahim Tragha. High throughput implementation of SHA3 hash algorithm on field programmable gate array (FPGA). *Microelectronics Journal*, 93:104615, 2019.

[144] Ming Ming Wong, Jawad Haj-Yahya, Suman Sau, and Anupam Chattopadhyay. A new high throughput and area efficient SHA-3 implementation. In *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–5. IEEE, 2018.

[145] Arshad Aziz et al. A low-power SHA-3 designs using embedded digital signal processing slice on FPGA. *Computers & Electrical Engineering*, 55:138–152, 2016.

[146] George Provelengios, Paris Kitsos, Nicolas Sklavos, and Christos Koulamas. FPGA-based design approaches of keccak hash function. In *2012 15th Euromicro Conference on Digital System Design*, pages 648–653. IEEE, 2012.

[147] Hassen Mestiri, Fatma Kahri, Mouna Bedoui, Belgacem Bouallegue, and Mohsen Machhout. High throughput pipelined hardware implementation of the KECCAK hash function. In *2016 International Symposium on Signal, Image, Video and Communications (ISIVC)*, pages 282–286. IEEE, 2016.

[148] Magnus Sundal and Ricardo Chaves. Efficient FPGA implementation of the SHA-3 hash function. In *2017 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, pages 86–91. IEEE, 2017.

[149] Lenos Ioannou, Harris E Michail, and Artemios G Voyiatzis. High performance pipelined FPGA implementation of the SHA-3 hash algorithm. In *2015 4th Mediterranean Conference on Embedded Computing (MECO)*, pages 68–71. IEEE, 2015.

[150] George S Athanasiou, George-Paris Makkas, and Georgios Theodoridis. High throughput pipelined FPGA implementation of the new SHA-3 cryptographic hash algorithm. In *2014 6th International Symposium on Communications, Control and Signal Processing (ISCCSP)*, pages 538–541. IEEE, 2014.

[151] Kris Gaj, Ekawat Homsirikamol, Marcin Rogawski, Rabia Shahid, and Malik Umar Sharif. Comprehensive evaluation of high-speed and medium-speed

implementations of five SHA-3 finalists using Xilinx and Altera FPGAs. *Cryptology ePrint Archive*, 2012.

[152] Pietro Nannipieri, Matteo Bertolucci, Luca Baldanzi, Luca Crocetti, Stefano Di Matteo, Francesco Falaschi, Luca Fanucci, and Sergio Saponara. SHA2 and SHA-3 accelerator design in a 7 nm technology within the European Processor Initiative. *Microprocessors and Microsystems*, 87:103444, 2021.

[153] Hassen Mestiri and Imen Barraj. High-Speed Hardware Architecture Based on Error Detection for KECCAK. *Micromachines*, 14(6):1129, 2023.

[154] Brian Baldwin, Andrew Byrne, Liang Lu, Mark Hamilton, Neil Hanley, Maire O'Neill, and William P Marnane. FPGA implementations of the round two SHA-3 candidates. In *2010 International Conference on Field Programmable Logic and Applications*, pages 400–407. IEEE, 2010.

[155] Kentaro Katayama, Hidetoshi Matsumura, Hiroaki Kameyama, Shinichi Sazawa, and Yasuhiro Watanabe. An FPGA-accelerated high-throughput data optimization system for high-speed transfer via wide area network. In *2017 International Conference on Field Programmable Technology (ICFPT)*, pages 211–214. IEEE, 2017.

[156] Information Technology Laboratory Computer Security Division. Example Values - Cryptographic Standards and Guidelines: CSRC, 2016. URL `https://nist.gov/itl/csd`.

[157] Harris Michail, Athanasios Kakarountas, Athanasios Milidonis, and Costas Goutis. A top-down design methodology for ultrahigh-performance hashing cores. *IEEE Transactions on Dependable and Secure computing*, 6(4):255–268, 2008.

[158] Xilinx Inc. Xilinx Power Estimator v2018.2. User Guide. https://docs.xilinx.com/v/u/2018.2-English/ug440-xilinx-power-estimator, 2018.

[159] Argyrios Sideris, Theodora Sanida, and Minas Dasygenis. Hardware acceleration design of the SHA-3 for high throughput and low area on FPGA. *Journal of Cryptographic Engineering*, pages 1–13, 2024.

[160] Sergiy Gnatyuk, Vasyl Kinzeryavyy, Karina Kyrychenko, Khalicha Yubuzova, Marek Aleksander, and Roman Odarchenko. Secure hash function constructing

for future communication systems and networks. In *Advances in Artificial Systems for Medicine and Education II 2*, pages 561–569. Springer, 2020.

[161] Song Guo, Deze Zeng, and Yang Xiang. Chameleon hashing for secure and privacy-preserving vehicular communications. *IEEE Transactions on Parallel and Distributed Systems*, 25(11):2794–2803, 2013.

[162] Muhammad Rehan Anwar, Desy Apriani, and Irsa Rizkita Adianita. Hash algorithm in verification of certificate data integrity and security. *Aptisi Transactions on Technopreneurship (ATT)*, 3(2):181–188, 2021.

[163] Lianhua Chi and Xingquan Zhu. Hashing techniques: A survey and taxonomy. *ACM Computing Surveys (CSUR)*, 50(1):1–36, 2017.

[164] Mesala M Sravani and S Ananiah Durai. Attacks on cryptosystems implemented via VLSI: A review. *Journal of Information Security and Applications*, 60:102861, 2021.

[165] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. Keccak. In *Advances in Cryptology–EUROCRYPT 2013: 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings 32*, pages 313–314. Springer, 2013.

[166] Todor Mladenov and Saeid Nooshabadi. Implementation of reconfigurable SHA-2 hardware core. In *APCCAS 2008-2008 IEEE Asia Pacific Conference on Circuits and Systems*, pages 1802–1805. IEEE, 2008.

[167] Lin Li, Shaoyu Lin, Shuli Shen, Kongcheng Wu, Xiaochao Li, and Yihui Chen. High-throughput and area-efficient fully-pipelined hashing cores using BRAM in FPGA. *Microprocessors and Microsystems*, 67:82–92, 2019.

[168] Hoai Luan Pham, Thi Hong Tran, Vu Trung Duong Le, and Yasuhiko Nakashima. A high-efficiency fpga-based multimode sha-2 accelerator. *IEEE Access*, 10: 11830–11845, 2022.

[169] Paul D Rosero-Montalvo, Zsolt István, and Wilmar Hernandez. A Survey of Trusted Computing Solutions Using FPGAs. *IEEE Access*, 2023.

[170] Marc Rothmann and Mario Porrmann. A survey of domain-specific architectures for reinforcement learning. *IEEE Access*, 10:13753–13767, 2022.

[171] Miroslav Knezevic, Kazuyuki Kobayashi, Jun Ikegami, Shin'ichiro Matsuo, Akashi Satoh, Ünal Kocabas, Junfeng Fan, Toshihiro Katashita, Takeshi Sugawara, Kazuo Sakiyama, et al. Fair and consistent hardware evaluation of fourteen round two SHA-3 candidates. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 20(5):827–840, 2011.

[172] Jori Winderickx, An Braeken, Dave Singelée, and Nele Mentens. In-depth energy analysis of security algorithms and protocols for the Internet of Things. *Journal of Cryptographic Engineering*, 12(2):137–149, 2022.

[173] Joshua R Templin and Jason R Hamlet. A new power-aware FPGA design metric. *Journal of Cryptographic Engineering*, 5:1–11, 2015.

[174] Juan J Rodríguez-Andina, Maria D Valdes-Pena, and Maria J Moure. Advanced features and industrial applications of FPGAs—A review. *IEEE Transactions on Industrial Informatics*, 11(4):853–864, 2015.

[175] Bernhard Jungk and Marc Stöttinger. Serialized lightweight SHA-3 FPGA implementations. *Microprocessors and Microsystems*, 71:102857, 2019.

[176] Yi Yang, Debiao He, Neeraj Kumar, and Sherali Zeadally. Compact hardware implementation of a SHA-3 core for wireless body sensor networks. *IEEE Access*, 6:40128–40136, 2018.

[177] Argyrios Sideris, Theodora Sanida, and Minas Dasygenis. High Throughput Implementation of the Keccak Hash Function Using the Nios-II Processor. *Technologies*, 8(1):15, 2020.

[178] Argyrios Sideris, Theodora Sanida, and Minas Dasygenis. High throughput pipelined implementation of the SHA-3 cryptoprocessor. In *2020 32nd International Conference on Microelectronics (ICM)*, pages 1–4. IEEE, 2020.

[179] Alia Arshad, Arshad Aziz, et al. Compact implementation of SHA3-512 on FPGA. In *2014 Conference on Information Assurance and Cyber Security (CIACS)*, pages 29–33. IEEE, 2014.

[180] Bernhard Jungk and Marc Stöttinger. Among slow dwarfs and fast giants: A systematic design space exploration of KECCAK. In *2013 8th International Workshop on Reconfigurable and Communication-Centric Systems-on-Chip (ReCoSoC)*, pages 1–8. IEEE, 2013.

[181] Kashif Latif, Arshad Aziz, and Athar Mahboob. Look-up table based implementations of SHA-3 finalists: JH, Keccak and Skein. *KSII Transactions on Internet and Information Systems (TIIS)*, 6(9):2388–2404, 2012.

[182] Kris Gaj, Ekawat Homsirikamol, and Marcin Rogawski. Fair and comprehensive methodology for comparing hardware performance of fourteen round two SHA-3 candidates using FPGAs. In *Cryptographic Hardware and Embedded Systems, CHES 2010: 12th International Workshop, Santa Barbara, USA, August 17-20, 2010. Proceedings 12*, pages 264–278. Springer, 2010.

[183] Arshad Aziz, Kashif Latif, et al. Resource Efficient Implementation of the Keccak, Skein & JH Algorithms on a Reconfigurable Platform. *Cankaya University Journal of Science and Engineering*, 13(1):40–57, 2016.

[184] Rourab Paul and Sandeep Shukla. Partitioned security processor architecture on FPGA platform. *IET Computers & Digital Techniques*, 12(5):216–226, 2018.

[185] Atefeh Gholipour and Sattar Mirzakuchaki. High-speed implementation of the Keccak hash function on FPGA. *International Journal of Advanced Computer Science*, 2(8):303–307, 2012.

[186] Doan Van Hieu and Lam Duc Khai. A Fast Keccak Hardware Design for High Performance Hashing System. In *2021 15th International Conference on Advanced Computing and Applications (ACOMP)*, pages 162–168. IEEE, 2021.

[187] Fatimazahraa Assad, Mohamed Fettach, Fadwa El Otmani, and Abderrahim Tragha. High-performance FPGA implementation of the secure hash algorithm 3 for single and multi-message processing. *International Journal of Electrical and Computer Engineering (IJECE)*, 12(2):1324–1333, 2022.

[188] Oscar Ferraz, Srinivasan Subramaniyan, Ramesh Chinthala, João Andrade, Joseph R Cavallaro, Soumitra K Nandy, Vitor Silva, Xinmiao Zhang, Madhura Purnaprajna, and Gabriel Falcao. A survey on high-throughput non-binary LDPC decoders: ASIC, FPGA, and GPU architectures. *IEEE Communications Surveys & Tutorials*, 24(1):524–556, 2021.

[189] Kamyar Mohajerani, Richard Haeussler, Rishub Nagpal, Farnoud Farahmand, Abubakr Abdulgadir, Jens-Peter Kaps, and Kris Gaj. FPGA Benchmarking of round 2 candidates in the NIST lightweight cryptography standardization process: methodology, metrics, tools, and results. *Cryptology ePrint Archive*, 2020.

[190] Argyrios Sideris, Theodora Sanida, and Minas Dasygenis. A Novel Hardware Architecture for Enhancing the Keccak Hash Function in FPGA Devices. *Information*, 14(9):475, 2023.

[191] N Madhusudhana Reddy, G Ramesh, Srinivasa Babu Kasturi, D Sharmila, G Gopichand, and L Thomas Robinson. Secure data storage and retrieval system using hybridization of orthogonal knowledge swarm optimization and oblique cryptography algorithm in cloud. *Applied Nanoscience*, 13(3):2449–2461, 2023.

[192] Emmanuel A Adeniyi, Peace Busola Falola, Mashael S Maashi, Mohammed Aljebreen, and Salil Bharany. Secure sensitive data sharing using RSA and ElGamal cryptographic algorithms with hash functions. *Information*, 13(10): 442, 2022.

[193] Jameel Almalki, Waleed Al Shehri, Rashid Mehmood, Khalid Alsaif, Saeed M Alshahrani, Najlaa Jannah, and Nayyar Ahmed Khan. Enabling Blockchain with IoMT Devices for Healthcare. *Information*, 13(10):448, 2022.

[194] Ashwini Kore and Shailaja Patil. Cross layered cryptography based secure routing for IoT-enabled smart healthcare system. *Wireless Networks*, pages 1–15, 2022.

[195] Manju Khari, Aditya Kumar Garg, Amir H Gandomi, Rashmi Gupta, Rizwan Patan, and Balamurugan Balusamy. Securing data in Internet of Things (IoT) using cryptography and steganography techniques. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 50(1):73–80, 2019.

[196] Alireza Sadeghi-Nasab and Vahid Rafe. A comprehensive review of the security flaws of hashing algorithms. *Journal of Computer Virology and Hacking Techniques*, pages 1–16, 2022.

[197] Nimish Mishra, SK Hafizul Islam, and Sherali Zeadally. A comprehensive review on collision-resistant hash functions on lattices. *Journal of Information Security and Applications*, 58:102782, 2021.

[198] Stefania Loredana Nita and Marius Iulian Mihailescu. Hash Functions. In *Cryptography and Cryptanalysis in Java: Creating and Programming Advanced Algorithms with Java SE 17 LTS and Jakarta EE 10*, pages 101–112. Springer, 2022.

[199] Kazuki Nakamura, Koji Hori, and Shoichi Hirose. Algebraic Fault Analysis of SHA-256 Compression Function and Its Application. *Information*, 12(10):433, 2021.

[200] Gorjan Alagic, Daniel Apon, David Cooper, Quynh Dang, Thinh Dang, John Kelsey, Jacob Lichtinger, Carl Miller, Dustin Moody, Rene Peralta, et al. Status report on the third round of the NIST post-quantum cryptography standardization process. *US Department of Commerce, NIST*, 2022.

[201] Young Beom Kim, Taek-Young Youn, and Seog Chung Seo. Chaining optimization methodology: a new sha-3 implementation on low-end microcontrollers. *Sustainability*, 13(8):4324, 2021.

[202] An Braeken. Highly efficient symmetric key based authentication and key agreement protocol using Keccak. *Sensors*, 20(8):2160, 2020.

[203] Thibaut Vandervelden, Ruben De Smet, Kris Steenhaut, and An Braeken. SHA 3 and Keccak variants computation speeds on constrained devices. *Future Generation Computer Systems*, 128:28–35, 2022.

[204] Julián Caba, María Díaz, Jesús Barba, Raúl Guerra, Soledad Escolar, and Sebastián López. Low-power hyperspectral anomaly detector implementation in cost-optimized FPGA devices. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:2379–2393, 2022.

[205] Argyrios Sideris, Theodora Sanida, and Minas Dasygenis. High throughput pipelined implementation of the SHA-3 cryptoprocessor. In *2020 32nd International Conference on Microelectronics (ICM)*, pages 1–4. IEEE, 2020.

[206] F Assad, F Elotmani, M Fettach, and A Tragha. An optimal hardware implementation of the KECCAK hash function on virtex-5 FPGA. In *2019 International Conference on Systems of Collaboration Big Data, Internet of Things & Security (SysCoBIoTS)*, pages 1–5. IEEE, 2019.

[207] Hachem Bensalem, Yves Blaquière, and Yvon Savaria. An efficient OpenCL-Based implementation of a SHA-3 co-processor on an FPGA-centric platform. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 2022.

[208] Jubin Mitra and Tapan K Nayak. An FPGA-based phase measurement system. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 26(1):133–142, 2017.

[209] Muzaffar Rao, Thomas Newe, Ian Grout, and Avijit Mathur. High speed implementation of a SHA-3 core on Virtex-5 and Virtex-6 FPGAs. *Journal of Circuits, Systems and Computers*, 25(07):1650069, 2016.

[210] Fatma Kahri, Hassen Mestiri, Belgacem Bouallegue, and Mohsen Machhout. High speed FPGA implementation of cryptographic KECCAK hash function crypto-processor. *Journal of Circuits, Systems and Computers*, 25(04):1650026, 2016.

[211] Riccardo Della Sala, Davide Bellizia, and Giuseppe Scotti. High-Throughput FPGA-Compatible TRNG Architecture Exploiting Multistimuli Metastable Cells. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 69(12):4886–4897, 2022.

[212] Wang Jinpeng, Zhang Teng, Zhang Bo, Zhao Xin, et al. An Innovative FPGA Implementations of the Secure frequency hopping communication system based on the improved ZUC algorithm. *IEEE Access*, 10:54634–54648, 2022.

[213] Amer Aljaedi, Sajjad Shaukat Jamal, Muhammad Rashid, Adel R Alharbi, Mohammed Alotaibi, and Dalal J Alanazi. Area-Efficient Realization of Binary Elliptic Curve Point Multiplication Processor for Cryptographic Applications. *Applied Sciences*, 13(12):7018, 2023.

[214] Binh Kieu-Do-Nguyen, Cuong Pham-Quoc, Ngoc-Thinh Tran, Cong-Kha Pham, and Trong-Thuc Hoang. Low-Cost Area-Efficient FPGA-Based Multi-Functional ECDSA/EdDSA. *Cryptography*, 6(2):25, 2022.

# Παράρτημα Α΄

# Δημοσιεύσεις

Κατά τη διάρκεια της διδακτορικής διατριβής δημοσιεύθηκαν οι ακόλουθες εργασίες σε Περιοδικά, Συνέδρια και Κεφάλαια Βιβλίων, αντίστοιχα.

## Α΄.1 Περιοδικά

**[Π5]** Sideris, A., Sanida, T. and Dasygenis, M. (2024). Hardware acceleration design of the SHA-3 for high throughput and low area on FPGA. Journal of Cryptographic Engineering, 1-13.

**[Π2]** Sanida, T., Sanida, M. V., Sideris, A., & Dasygenis, M. (2024). Enhancing Pulmonary Diagnosis in Chest X-rays through Generative AI Techniques. J, 7(3), 302-318.

**[Π3]** Sanida, M. V., Sanida, T., Sideris, A. and Dasygenis, M. (2024). An advanced deep learning framework for multi-class diagnosis from chest x-ray images. J, 7(1), 48-71.

**[Π4]** Sanida, T., Sanida, M. V., Sideris, A., & Dasygenis, M. (2024). Optimizing Lung Condition Categorization through a Deep Learning Approach to Chest X-ray Image Analysis. BioMedInformatics, 4(3), 2002-2021.

**[Π5]** Sanida, T., Sideris, A., Sanida, M. V. and Dasygenis, M. (2023). Tomato leaf disease identification via two–stage transfer learning approach. Smart Agricultural Technology, 5, 100275.

**[Π6]** Sideris, A., Sanida, T. and Dasygenis, M. (2023). A Novel Hardware Architecture for Enhancing the Keccak Hash Function in FPGA Devices. Information, 14(9), 475.

**[Π7]** Sideris, A. and Dasygenis, M. (2023). Enhancing the Hardware Pipelining Optimization Technique of the SHA-3 via FPGA. Computation, 11(8), 152.

**[Π8]** Sanida, T., Tabakis, I. M., Sanida, M. V., Sideris, A. and Dasygenis, M. (2023). A Robust Hybrid Deep Convolutional Neural Network for COVID-19 Disease Identification from Chest X-ray Images. Information, 14(6), 310.

**[Π9]** Sanida, M. V., Sanida, T., Sideris, A. and Dasygenis, M. (2023). An Efficient hybrid cnn classification model for tomato crop disease. Technologies, 11(1), 10.

**[Π10]** Sideris, A., Sanida, T., Tsiktsiris, D. and Dasygenis, M. (2022). Acceleration of Image Processing with SHA-3 (Keccak) Algorithm using FPGA. J. Eng. Res. Sci, 1, 20-28.

**[Π11]** Sanida, T., Tsiktsiris, D., Sideris, A. and Dasygenis, M. (2022). A heterogeneous implementation for plant disease identification using deep learning. Multimedia Tools and Applications, 81(11), 15041-15059.

**[Π12]** Sanida, T., Sideris, A., Tsiktsiris, D. and Dasygenis, M. (2022). Lightweight neural network for COVID-19 detection from chest X-ray images implemented on an embedded system. Technologies, 10(2), 37.

**[Π13]** Sideris, A., Sanida, T. and Dasygenis, M. (2020). High throughput implementation of the keccak hash function using the Nios-ii processor. Technologies, 8(1), 15.

## Α΄.2 Συνέδρια

**[Σ1]** Sanida, T., Sideris, A., Sanida, M. V., Dossis, M., & Dasygenis, M. (2024, September). Accelerating CNNs for Pneumonia Disease Diagnosis via Heterogeneous FPGA Systems. In 2024 9th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM) (pp. 159-162). IEEE.

**[Σ2]** Sanida, M. V., Sanida, T., Sideris, A., Dossis, M., & Dasygenis, M. (2024, September). Fake News Detection Approach Using Hybrid Deep Learning Framework. In 2024 9th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM) (pp. 81-84). IEEE.

**[Σ3]** Sanida, T., Sideris, A., Sanida, M. V., Dossis, M. and Dasygenis, M. (2023, November). An Efficiency CNN Solution for Olive Disease Management Through FPGA. In 2023 8th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM) (pp. 1-4). IEEE.

**[Σ4]** Sanida, T., Sanida, M. V., Sideris, A., Dossis, M. and Dasygenis, M. (2023, November). Efficient Categorization of Pneumonia Diagnosis Using Low-Power Embedded Devices. In 2023 8th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM) (pp. 1-4). IEEE.

**[Σ5]** Sideris, A., Sanida, T., Sanida, M. V., Dossis, M. and Dasygenis, M. (2023, November). Accelerate Processing of Image with the Keccak-512 Algorithm on Cryptoprocessor. In 2023 8th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM) (pp. 1-4). IEEE.

**[Σ6]** Sanida, T., Sideris, A., Sanida, M. V., Dossis, M. and Dasygenis, M. (2023, November). Acceleration of GANs for Potato Crop Disease Identification via FPGA. In 2023 8th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM) (pp. 1-4). IEEE.

**[Σ7]** Sanida, T., Sanida, M. V., Sideris, A. and Dasygenis, M. (2023, June). A Lightweight CNN Model for Tomato Crop Diseases on Heterogeneous Embedded System. In 2023 12th International Conference on Modern Circuits and Systems Technologies (MOCAST) (pp. 1-4). IEEE.

**[Σ8]** Sanida, T., Sideris, A., Chatzisavvas, A., Dossis, M. and Dasygenis, M. (2022, September). Radiography Images with Transfer Learning on Embedded System. In 2022 7th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM) (pp. 1-4). IEEE.

**[Σ9]** Sideris, A., Sanida, T., Chatzisavvas, A., Dossis, M. and Dasygenis, M. (2022, September). High Throughput of Image Processing with Keccak Algorithm using Microprocessor on FPGA. In 2022 7th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM) (pp. 1-4). IEEE.

**[Σ10]** Sideris, A., Sanida, T., Tsiktsiris, D. and Dasygenis, M. (2022). Image hashing based on sha-3 implemented on fpga. In Recent Advances in Manufacturing Modelling and Optimization: Select Proceedings of RAM 2021 (pp. 521-530). Singapore: Springer Nature Singapore.

**[Σ11]** Tsiktsiris, D., Sanida, T., Sideris, A. and Dasygenis, M. (2022). Accelerated Defective Product Inspection on the Edge Using Deep Learning. In Recent Advances in Manufacturing Modelling and Optimization: Select Proceedings of RAM 2021 (pp. 185-191). Singapore: Springer Nature Singapore.

**[Σ12]** Sanida, T., Tsiktsiris, D., Sideris, A. and Dasygenis, M. (2021, July). A Heterogeneous Lightweight Network for Plant Disease Classification. In 2021 10th International Conference on Modern Circuits and Systems Technologies (MOCAST) (pp. 1-4). IEEE.

**[Σ13]** Sideris, A., Sanida, T. and Dasygenis, M. (2020, December). High throughput pipelined implementation of the sha-3 cryptoprocessor. In 2020 32nd International Conference on Microelectronics (ICM) (pp. 1-4). IEEE.

**[Σ14]** Sanida, T., Sideris, A. and Dasygenis, M. (2020, September). Accelerating the AES algorithm using opencl. In 2020 9th International conference on modern circuits and systems technologies (MOCAST) (pp. 1-4). IEEE.

**[Σ15]** Sanida, T., Sideris, A. and Dasygenis, M. (2020, September). A heterogeneous implementation of the Sobel edge detection filter using OpenCL. In 2020 9th International Conference on Modern Circuits and Systems Technologies (MOCAST) (pp. 1-4). IEEE.

**[Σ16]** Sideris, A., Sanida, T. and Dasygenis, M. (2019, November). Hardware acceleration of the aes algorithm using nios-ii processor. In 2019 Panhellenic Conference on Electronics and Telecommunications (PACET) (pp. 1-5). IEEE.

**[Σ17]** Sideris, A., Sanida, T. and Dasygenis, M. (2019, May). Hardware acceleration of SHA-256 algorithm using NIOS-II processor. In 2019 8th International

Conference on Modern Circuits and Systems Technologies (MOCAST) (pp. 1-4). IEEE.

## Α΄.3 Κεφάλαια Βιβλίων

**[KB1]** Sideris, A., Tsiktsiris, D., Ziouzios, D. and Dasygenis, M. (2019). Smart grid hardware security. IoT for Smart Grids: Design Challenges and Paradigms, 85-113.

**[KB2]** Ziouzios, D., Sideris, A., Tsiktsiris, D. and Dasygenis, M. (2019). Smart-Grid Modelling and Simulation. IoT for Smart Grids: Design Challenges and Paradigms, 43-54.