ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ Τ.Ε

Μεταπτυχιακό Πρόγραμμα Σπουδών: Τεχνολογίες & Συστήματα Ευρυζώνικών Εφαρμογών & Υπηρεσιών

ΜΕΛΕΤΗ ΚΑΙ ΥΛΟΠΟΙΗΣΗ ΠΑΡΑΓΩΓΗΣ ΚΛΕΙΔΙΩΝ ΑΣΦΑΛΕΙΑΣ ΣΤΗΝ ΑΣΥΡΜΑΤΗ ΤΕΧΝΟΛΟΓΙΑ BLUETOOTH

HARDWARE IMPLEMENTATION OF BLUETOOTH SECURITY KEY GENERATION

ΦΙΛΙΠΠΟΣ ΠΙΡΠΙΛΙΔΗΣ
Μεταπτυχιακός Φοιτητής

Επιβλέπων Καθηγητής : ΠΑΡΑΣΚΕΥΑΣ ΚΙΤΣΟΣ

ΠΕΡΙΕΧΟΜΕΝΑ

Σελίδα

ΛΙΣΤΑ ΕΙΚΟΝΩΝ

1. ΤΕΧΝΟΛΟΓΙΑ BLUETOOTH

1. ΕΙΣΑΓΩΓΗ

Η ασύρματη τεχνολογία Bluetooth είναι μια τεχνολογία που επιτρέπει την επικοινωνία σε μικρές αποστάσεις. Η τεχνολογία Bluetooth μπορεί να μεταδώσει σήματα σε μικρές αποστάσεις ανάμεσα σε τηλέφωνα, υπολογιστές και άλλες συσκευές. Με δεδομένο ότι το μέσο μετάδοσης όλων των ασύρματων συσκευών είναι κοινό, ένα μεγάλο ζήτημα να όχι το μεγαλύτερο είναι η ασφάλεια επικοινωνίας μεταξύ των συσκευών. Στην εργασία έγινε υλοποίηση σε υλικό της πιστοποίησης αυθεντικότητας χρηστή αλλά και της παραγωγής κλειδιών κρυπτογράφησης που απαιτείται για την κρυπτογράφηση των δεδομένων πριν αυτά μεταβούν στο κανάλι επικοινωνίας. Η υλοποίηση έγινε σύμφωνα με τις προδιαγραφές της έκδοσης Bluetooth 5.0 ενώ για μετρήσεις και αποτελέσματα χρησιμοποιήθηκε η συσκευή FPGA Zedboard της Xilinx .

2. ΤΙ ΕΙΝΑΙ BLUETOOTH

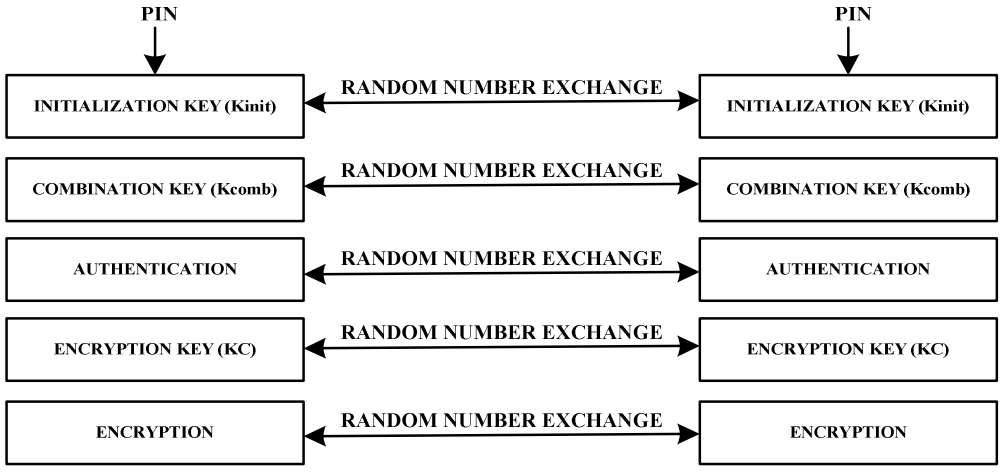
Το Bluetooth όπως αναφέραμε και νωρίτερα είναι μια ασύρματη τεχνολογία μικρών αποστάσεων για ασύρματα προσωπικά δίκτυα υπολογιστών. Η τεχνολογία αυτή μπορεί να μεταδώσει σήματα μέσω μικροκυμάτων σε ψηφιακές συσκευές. Συνοπτικά, το Bluetooth είναι ένα πρωτόκολλο το οποίο παρέχει ασύρματη επικοινωνία ανάμεσα σε κινητά τηλέφωνα, προσωπικούς υπολογιστές, εκτυπωτές και άλλες συσκευές μέσω μιας ασφαλούς, φθηνής και παγκοσμίως διαθέσιμης ραδιοσυχνότητας μικρής εμβέλειας. Το Bluetooth κατανέμεται στο φυσικό επίπεδο, υπό-επίπεδο MAC και προαιρετικά στο υπό-επίπεδο LLC του OSI.

3. ΙΣΤΟΡΙΑ BLUETOOTH

Η εταιρία Ericsson έθεσε τις βάσεις για την ανάπτυξη μιας τεχνολογίας πολύ μικρής εμβέλειας με σκοπό την ασύρματη και ad hoc δικτύωση ετερογενών φορητών συσκευών. Η ιδέα ξεκίνησε από την επιθυμία απαλλαγής από τα καλώδια RS-232 στα τέλη της δεκαετίας 1990. Το 1994 πέντε εταιρίες Ericsson , Nokia, Toshiba, Intel και IBM ανακοίνωσαν την πρώτη έκδοση του Bluetooth. Οι σχεδιαστές ήταν βέβαιοι ότι το νέο πρότυπο θα επικρατούσε και θα έφερνε ακόμη πιο κοντά τους ανθρώπους και τις συσκευές τους. Σαν πρότυπο υιοθετήθηκε από την IEEE με ονομασία 802.15. Το 2018 η έκδοση του Bluetooth είναι η v5. Οι βασικές αλλαγές από την v1 έως την v5 αφορούν υποστήριξη πρωτοκόλλων γρήγορου ρυθμού μετάδοσης και χαμηλής κατανάλωσης.

4. ΑΣΦΑΛΕΙΑ BLUETOOTH

Ένα από τα βασικά ζητήματα που απασχολεί τις ασύρματες τεχνολογίες είναι η ασφάλεια των δεδομένων που μεταδίδονται. Αν και ασύρματη τεχνολογία μικρών αποστάσεων, η ασφάλεια στο Bluetooth είναι εξίσου σημαντική. Η διαδικασία ασφάλειας σε γενικά πλαίσια δεν διαφέρει από τις υπόλοιπες ασύρματες επικοινωνίες καθώς σαν τρόπος ασφάλισης των δεδομένων χρησιμοποιεί την κρυπτογράφηση. Το πιο πολύπλοκο και ταυτόχρονα σημαντικό κομμάτι στην ασφάλεια του Bluetooth είναι η πιστοποίηση και η παραγωγή κλειδιών κρυπτογράφησης. Τα κλειδιά αυτά χρησιμοποιούνται προκειμένου να μπορούν οι συσκευές να κρυπτογραφούν και να αποκρυπτογραφούν τα δεδομένα που στέλνουν και λαμβάνουν, αντίστοιχα. Στην εικόνα 1 βλέπουμε ονομαστικά τις διαδικασίες που λαμβάνουν χώρα από την τοποθέτηση του αριθμού PIN ως την διαδικασία της κρυπτογράφησης. Στην πτυχιακή εργασία ασχοληθήκαμε με τις διαδικασίες AUTHENTICATION και ENCRYPTION KEY.



Εικόνα 1: Διαδικασίες ασφάλειας Bluetooth

Δεδομένου ότι το Bluetooth είναι σε θέση να επικοινωνήσει με σχεδόν οποιαδήποτε άλλη Bluetooth συσκευή και το μέσο μετάδοσης είναι κοινό για όλες τις ασύρματες συσκευές, η ασφάλεια είναι μια σημαντική ανησυχία. Εντούτοις, πρέπει να αναφερθεί ότι η ασφάλεια του Bluetooth έχει σχεδιαστεί ώστε να πληρούνται όλες οι απαιτήσεις προστασίας και εμπιστευτικότητας πληροφοριών. Οι υπηρεσίες ασφαλείας εφαρμόζονται στο στρώμα σύνδεσης προσφέροντας πιστοποίηση της συσκευής και κρυπτογράφηση των δεδομένων πριν αυτά οδηγηθούν στο κανάλι επικοινωνίας. Οι τρείς βασικοί παράμετροι που χρησιμοποιούνται στην ασφάλεια Bluetooth είναι:

Η διεύθυνση της συσκευής Bluetooth (BD_ADDR) με μήκος 48 bits και είναι μοναδική σε κάθε Bluetooth συσκευή. Στο επίπεδο του προγραμματιστή αυτή η τιμή αναπαριστάται με 12 δεκαεξαδικούς χαρακτήρες(6 bytes).

Το κλειδί πιστοποίησης K, το οποίο έχει μήκος 128 bits και χρησιμοποιείται για σκοπούς πιστοποίησης της συσκευής.

Το κλειδί κρυπτογράφησης Kc το οποίο υπολογίζεται από hash συναρτήσεις και χρησιμοποιείται έπειτα σαν κλειδί στην κρυπτογράφηση των δεδομένων.

Η τιμή RAND η οποία είναι μια τυχαία τιμή των 128 bits που παράγει η συσκευή, η χρήση της οποίας είναι παρόμοια με αυτή στα συστήματα κινητών επικοινωνιών με αποτέλεσμα να μην έχει καθορισμένη τιμή.

1. ΚΑΤΑΣΤΑΣΕΙΣ ΑΣΦΑΛΕΙΑΣ BLUETOOTH

Κάθε συσκευή Bluetooth μπορεί να λειτουργήσει σε 3 καταστάσεις ασφάλειας.

1. Η κατάσταση ασφαλείας 1. Είναι μια μη ασφαλής κατάσταση, στην οποία η συσκευή δεν ξεκινά καμία διαδικασία ασφάλειας. Αυτή η κατάσταση επιτρέπει σε άλλες συσκευές Bluetooth να πραγματοποιούν σύνδεση στη συγκεκριμένη συσκευή.
2. Η κατάσταση ασφάλειας 2. Εφαρμόζεται στο στρώμα υπηρεσιών. Εδώ η συσκευή δεν κάνει κάποια διαδικασία ασφάλειας ώσπου να δημιουργηθεί το κανάλι επικοινωνίας(Discoverable level). Η έναρξη των διαδικασιών ασφαλείας εξαρτάται από τις απαιτήσεις είτε του ίδιου του καναλιού είτε των υπηρεσιών που θα χρησιμοποιηθούν. Στην κατάσταση αυτή ευνοείται η δημιουργία ευέλικτων πολιτικών ασφαλείας, που επιτρέπουν την ταυτόχρονη λειτουργία εφαρμογών, οι οποίες απαιτούν ως ένα βαθμό ασφάλεια με απλά πρωτόκολλα που δεν περιέχουν σχεδόν καμία ασφάλεια.
3. Η κατάσταση ασφάλειας 3 είναι η πιο ασφαλής κατάσταση επειδή εφαρμόζεται στο στρώμα σύνδεσης. Στη συγκεκριμένη κατάσταση η συσκευή ξεκινά τις διαδικασίες ασφάλειας πριν ολοκληρωθεί η σύνδεση.

2. ΕΠΙΠΕΔΑ ΑΣΦΑΛΕΙΑΣ BLUETOOTH

Μια συσκευή κατηγοριοποιεί τις άλλες συσκευές σε τρία διαφορετικά επίπεδα ασφαλείας. Στο πρώτο επίπεδο εντάσσονται οι έμπιστες συσκευές που σημαίνει ότι η συσκευή έχει περάσει την διαδικασία πιστοποίησης στο παρελθόν και το κλειδί είναι

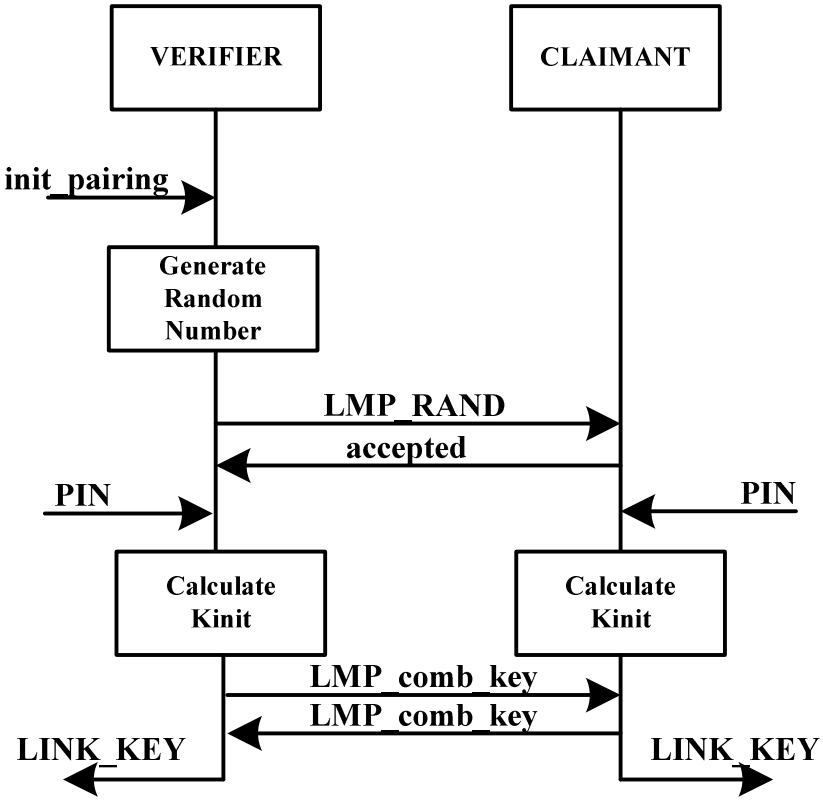
αποθηκευμένο. Αυτές οι συσκευές έχουν αποθηκευτεί στη βάση δεδομένων ως έμπιστες. Στο δεύτερο επίπεδο βρίσκονται οι μη-έμπιστες συσκευές. Αυτές οι συσκευές δεν έχουν περάσει την διαδικασία πιστοποίησης, όμως έχουν δημιουργήσει και αποθηκέυσει το κλειδί σύνδεσης(LINK_KEY). Στο τρίτο επίπεδο βρίσκονται οι άγνωστες συσκευές οι οποίες δεν έχουν δημιουργήσει ούτε το κλειδί σύνδεσης ούτε έχουν περάσει την διαδικασία πιστοποίησης και για αυτό αντιμετωπίζονται ως μη-έμπιστες συσκευές. Επίσης, υπάρχουν τέσσερα είδη υπηρεσιών που σχετίζονται με την ασφάλεια του Bluetooth. Στο πρώτο είδος ανήκουν οι υπηρεσίες που απαιτούν εξουσιοδότηση και πιστοποίηση της ταυτότητας της συσκευής. Η αυτόματη πρόσβαση παρέχεται μόνο στις έμπιστες συσκευές ενώ οι υπόλοιπες θα πρέπει να εξουσιοδοτηθούν. Η διαδικασία εξουσιοδότησης εμπεριέχει πάντα την διαδικασία πιστοποίησης, ώστε να διαπιστωθεί αν η απομακρυσμένη συσκευή είναι αυτή που ισχυρίζεται ότι είναι. Το δεύτερο είδος υπηρεσιών αφορά τις υπηρεσίες που απαιτούν πιστοποίηση πριν την πρόσβαση. Το τρίτο είδος αφορά τις υπηρεσίες που απαιτούν κρυπτογράφηση των δεδομένων πριν εγκριθεί η πρόσβαση στις υπηρεσίες, ενώ στο τέταρτο είδος ανήκουν οι υπηρεσίες οι οποίες σχετίζονται με τις ¨ ανοιχτές ¨ συσκευές στις οποίες για την πρόσβαση δεν απαιτείται καμία διαδικασία ασφάλειας.

5. ΔΙΑΔΙΚΑΣΙΕΣ ΑΣΦΑΛΕΙΑΣ

1. ΖΕΥΞΗ

Για να κατανοήσουμε καλύτερα τον μηχανισμό ασφάλειας του Bluetooth ας πάμε ένα επίπεδο πίσω από την συγκεκριμένη υλοποίηση. Στην εικόνα 2 φαίνεται η διαδικασία ζεύξης μέχρι την δημιουργία του κλειδιού σύνδεσης. Στην διαδικασία ζεύξης

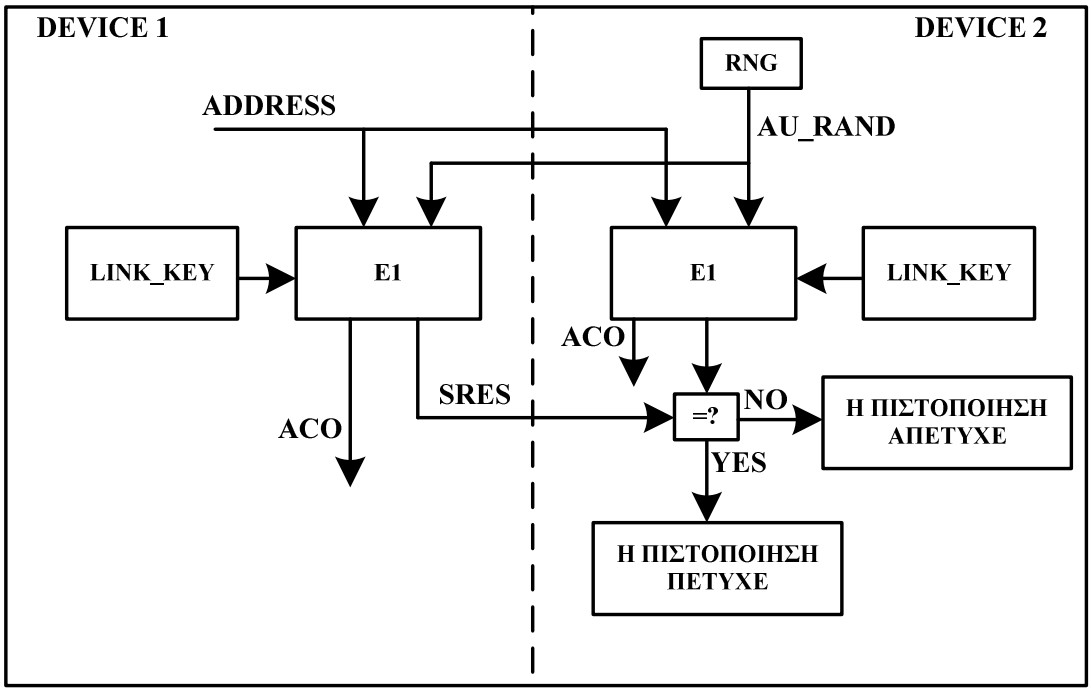
οι συσκευές μοιράζονται ΄΄μυστικά΄΄ το κλειδί σύνδεσης. Το κλειδί σύνδεσης χρησιμοποιείται για να πιστοποιηθεί η ταυτότητα μιας συσκευής. Στη συνέχεια το κλειδί σύνδεσης μαζί με παραμέτρους που έχουν δημιουργηθεί από τη διαδικασία πιστοποίησης χρησιμοποιούνται σαν είσοδος για τη δημιουργία του κλειδιού κρυπτογράφησης. Τα δεδομένα που θα ανταλλάξουν οι συσκευές από εκεί και έπειτα θα κρυπτογραφηθούν με το κλειδί κρυπτογράφησης πριν αυτά δρομολογηθούν στο κανάλι επικοινωνίας.



Εικόνα 2: Διαδικασία Παραγωγής Κλειδιού Σύνδεσης

2. ΠΙΣΤΟΠΟΙΗΣΗ ΤΑΥΤΟΤΗΤΑΣ

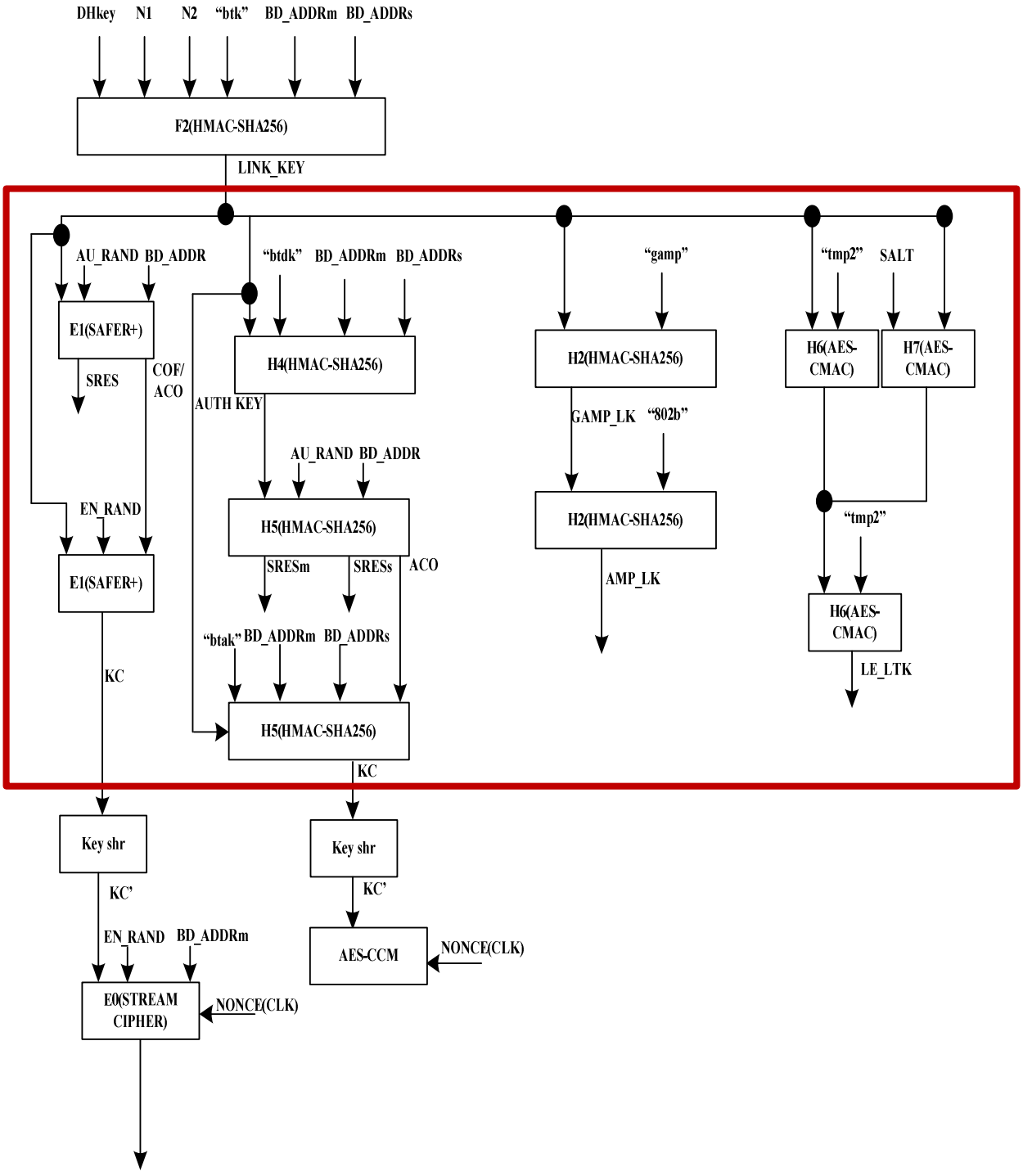
Στην εικόνα 3 βλέπουμε την διαδικασία Πιστοποίησης ταυτότητας. Αφού δημιουργηθεί το κλειδί σύνδεσης τότε κλειδί σύνδεσης διεύθυνση συσκευής και μια τιμή AU_RAND των 128 bit επεξεργάζονται και παράγουν μια 128 bit τιμή στην έξοδο. Αυτή με την σειρά της χωρίζεται σε 2 μέρη. Τα 32 πιο σημαντικά bits αποτελούν την μεταβλητή SRES η οποία στέλνεται στην συσκευή. Αν το αποτέλεσμα της λογικής πράξης XOR μεταξύ του SRES που λήφθηκε και του SRES που υπολογίστηκε είναι ίση με το 0 τότε η συσκευή είναι αυτή που ισχυρίζεται ότι είναι. Τα υπόλοιπα bits ανήκουν στην μεταβλητή ACO η οποία θα χρησιμοποιηθεί στη συνέχεια για την δημιουργία του κλειδιού κρυπτογράφησης.



Εικόνα 3: Πιστοποίηση Ταυτότητας

3. ΔΗΜΙΟΥΡΓΙΑ ΚΛΕΙΔΙΩΝ ΑΣΦΑΛΕΙΑΣ

Η διαχείριση των κλειδιών περιλαμβάνει την δημιουργία, αποθήκευση και διανομή κλειδιών με τελικό σκοπό την κρυπτογράφηση των δεδομένων πριν αυτά δρομολογηθούν στο κανάλι επικοινωνίας. Στην παρούσα εργασία έγινε υλοποίηση της διαδικασίας πιστοποίησης και παραγωγής κλειδιών κρυπτογράφησης. Ένα από τα βασικά κλειδιά που χρησιμοποιεί το Bluetooth είναι το κλειδί σύνδεσης. Το κλειδί σύνδεσης παράγεται από μια συνάρτηση που ονομάζεται f2 και χρησιμοποιεί τον αλγόριθμο κατακερματισμού sha-256 σε hmac λειτουργία. Οι αλγόριθμοι που χρησιμοποιούνται στις διαδικασίες πιστοποίησης και παραγωγής κλειδιών είναι : μια βελτιστοποιημένη έκδοση του συμμετρικού αλγόριθμου κρυπτογράφησης SAFER+, η συνάρτηση κατακερματισμού SHA256 σε HMAC λειτουργία καθώς και ο αλγόριθμος AES σε CMAC λειτουργία. Αξίζει να σημειωθεί ότι το Bluetooth 5 φέρνει μαζί του και 2 χαρακτηριστικά (LE και AMP) τα οποία υιοθέτησε από τις εκδόσεις v3 και v4 και επηρεάζουν την ιεραρχία της ασφάλειας του Bluetooth. Το LE(Low Energy) αφορά πρωτόκολλα χαμηλής κατανάλωσης ενώ το AMP(Alternative MAC/PHY) αφορά πρωτόκολλα υψηλής ταχύτητας. Αυτά τα δύο χαρακτηριστικά επηρεάζουν την υλοποίηση της ιεραρχίας με την παραγωγή μοναδικών κλειδιών για τις συγκεκριμένες λειτουργίες. Στην εικόνα 4 βλέπουμε την ιεραρχία παραγωγής κλειδιών ανάλογα με την λειτουργία ασφάλειας όπως μας δόθηκαν από τις επίσημες προδιαγραφές του Bluetooth 5.0. Με κόκκινο περίγραμμα είναι οι διαδικασίες που υλοποιήθηκαν και θα αναλύσουμε παρακάτω.



Εικόνα 4: Ιεραρχία Παραγωγής Κλειδιών

4. ΑΣΦΑΛΕΙΑ AMP

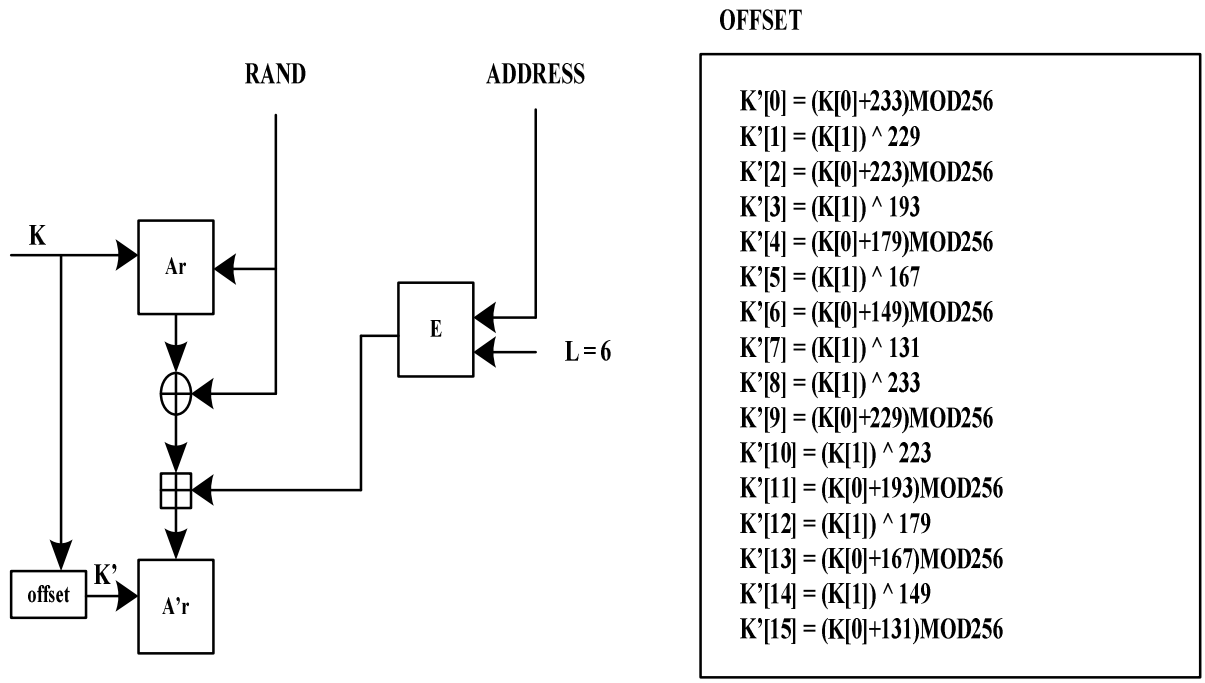
Η λειτουργία παραγωγής του κλειδιού AMP ξεκινά κάθε φορά που το κλειδί σύνδεσης δημιουργείται ή αλλάζει. Όταν το AMP κλειδί δημιουργηθεί όλα τα κλειδιά που σχετίζονται με τη συσκευή στο άλλο άκρο αφαιρούνται από τη βάση δεδομένων. Επίσης τερματίζεται οποιαδήποτε σύνδεση είναι ανοιχτή με τη συγκεκριμένη συσκευή. Όπως φαίνεται και στην εικόνα 4, για τη δημιουργία του κλειδιού AMP χρησιμοποιείται δύο φορές η συνάρτηση h2, η οποία με τη σειρά της χρησιμοποιεί τον αλγόριθμο sha-256 σε hmac λειτουργία.

5. ΑΣΦΑΛΕΙΑ LE

Ο μηχανισμός ζεύξης που υπάρχει στην τεχνολογία Bluetooth από την έκδοση v4 και έπειτα και είναι γνωστή και ως LE Legacy Pairing έχει κάποιες διαφορές στην ασφάλεια σε σχέση με την απλή ασφαλή ζεύξη. Το σύστημα είναι παρόμοιο με αυτό της απλής ασφαλής ζεύξης από πλευράς χρήστη και οι διαφορές αφορούν την ποιότητα και την προστασία που παρέχει. Στην εικόνα 4 βλέπουμε ότι, η παραγωγή του κλειδιού σε αυτή την λειτουργία χρησιμοποιεί τον αλγόριθμο AES-CMAC σε 2 συναρτήσεις, την h6 και την h7 από τις οποίες η h6 χρησιμοποιείται δύο φορές.

6. Η ΣΥΝΑΡΤΗΣΗ ΠΙΣΤΟΠΟΙΗΣΗΣ E1

Η συνάρτηση πιστοποίησης E1 είναι ένας ασφαλής κώδικας πιστοποίησης. Η συνάρτηση χρησιμοποιεί τον αλγόριθμο κρυπτογράφησης SAFER+, ο οποίος είναι μία βελτιστοποιημένη έκδοση του αλγόριθμου SAFER-SK128, και είναι δωρεάν διαθέσιμος. Η συνάρτηση στο εσωτερικό της έχει μία υπό-συνάρτηση που ονομάζεται Ar και χρησιμοποιείται δύο φορές προκειμένου να υπολογίσει το αποτέλεσμα. Η συνάρτηση A'r είναι μια αλλαγμένη έκδοση από την Ar και η βασική τους διαφορά αφορά την διαδικασία Key Schedule του αλγορίθμου SAFER+, η οποία εξηγείται βαθύτερα παρακάτω. Η είσοδος address μετατρέπεται σε 128 bits και τα bytes της τιμής address επαναλαμβάνονται όσες φορές χρειαστεί ξεκινώντας από το πιο σημαντικό byte προκειμένου να σχηματιστεί η 128 bit τιμή στην έξοδο του E. Το κλειδί πριν τοποθετηθεί στην συνάρτηση A'r θα υποστεί μια μικρή επεξεργασία. Στην εικόνα 5 βλέπουμε το εσωτερικό της συνάρτησης E1 καθώς και την επεξεργασία που γίνεται στο κλειδί που εισέρχεται στην υπό-συνάρτηση A'r.



Εικόνα 5: Ιεραρχία Παραγωγής Κλειδιών

7. Η ΣΥΝΑΡΤΗΣΗ ΠΑΡΑΓΩΓΗΣ ΚΛΕΙΔΙΟΥ Ε3

Το κλειδί Kc που υπολογίζεται από την συνάρτηση Ε3 χρησιμοποιείται σαν κλειδί κρυπτογράφησης στα δεδομένα που αποστέλλονται στο κανάλι επικοινωνίας. Η συνάρτηση έχει την ακόλουθη μορφή :

$$E_3: \{0,1\}^{128} x \{0,1\}^{128} x \{0,1\}^{96} \rightarrow \{0,1\}^{128}$$

$$(K, RAND, COF) \rightarrow Hash(K, RAND, COF, 12)$$

Η τιμή COF υπολογίζεται από την συνάρτηση Ε1 που περιγράψαμε παραπάνω και έχει μέγεθος 96 bits. Η RAND είναι μια ψευδό-τυχαία τιμή των 128 bits που παράγεται σε μία από της δυο συσκευές που πρόκειται να επικοινωνήσουν ενώ η τιμή Κ των 128 bits είναι το κλειδί σύνδεσης.

8. Η ΣΥΝΑΡΤΗΣΗ H2

Ο ορισμός της απλής ζεύξης αφορά την μέθοδο παραγωγής με κλειδί το AMP Key, η οποία χρησιμοποιεί την συνάρτηση H2 που με τη σειρά της χρησιμοποιεί τον αλγόριθμο sha-256 σε HMAC μορφή. Οι είσοδοι στην συνάρτηση αυτή είναι οι ακόλουθοι :

W – 256 bits

KeyID -32 bits

L – 8 bits

Η έξοδος υπολογίζεται σύμφωνα με τον παρακάτω τύπο:

$$h2(W, keyID, L) = HMAC - SHA - 256_w(keyID)/2^{256-(L*8)}$$

9. Η ΣΥΝΑΡΤΗΣΗ H3

Σε αυτή τη συνάρτηση υπολογίζεται το κλειδί που θα χρησιμοποιηθεί για την κρυπτογράφηση των δεδομένων με τον αλγόριθμο AES. Ο πυρήνας της συνάρτησης είναι ο αλγόριθμος κατακερματισμού sha-256 σε HMAC μορφή.

Οι είσοδοι στην συνάρτηση h3 είναι :

Είσοδος	**Μέγεθος σε bits**
T	128
keyID	32
A_1	48
A_2	48
ACO	64

Όπου T είναι το κλειδί σύνδεσης, KeyID μια 32 bit σταθερή τιμή(0x6274616b), A1 η διεύθυνση BD_ADDDRm, A2 η διεύθυνση BD_ADDRs και ACO τα 64 λιγότερο σημαντικά bits του αποτελέσματος της συνάρτησης H5.

Η έξοδος υπολογίζεται από την ακόλουθη συνάρτηση.

$$h3(W, keyID, A_1, A_2, ACO) = HMAC - SHA - 256_T(keyID||A_1||A_2||ACO)/2^{128}$$

10. Η ΣΥΝΑΡΤΗΣΗ H4

Η H4 συνάρτηση χρησιμοποιείται για να υπολογίσει το κλειδί πιστοποίησης το οποίο με τη σειρά του είναι είσοδος στην συνάρτηση H5 προκειμένου να ολοκληρωθεί η διαδικασία πιστοποίησης όταν πρόκειται για ασφαλή σύνδεση μεταξύ των συσκευών. Η συνάρτηση όπως και η H3 χρησιμοποιεί τον αλγόριθμο sha-256 σε HMAC μορφή.

Οι είσοδοι της συνάρτησης H4 :

Είσοδος	**Μέγεθος σε bits**
T	128
keyID	32
A_1	48
A_2	48

Όπου T είναι το κλειδί σύνδεσης, KeyID μια 32 bit σταθερή τιμή(0x6274646b), A1 η διεύθυνση BD_ADDDRm και A2 η διεύθυνση BD_ADDRs.

Ενώ η έξοδος υπολογίζεται από την ακόλουθη συνάρτηση.

$$h4(W, keyID, A_1, A_2) = HMAC - SHA - 256_T(keyID||A_1||A_2|)/2^{128}$$

11. Η ΣΥΝΑΡΤΗΣΗ H5

Στις ασφαλείς συνδέσεις το δεύτερο και τελευταίο στάδιο της πιστοποίησης ταυτότητας αποτελεί η συνάρτηση H5 που στον πυρήνα της έχει τον αλγόριθμο sha-256 σε HMAC μορφή. Η H5 συνάρτηση υπολογίζει τις δύο τιμές SRES καθώς και την τιμή ACO που χρησιμοποιείται σαν είσοδος στην συνάρτηση H3 προκειμένου να υπολογιστεί το κλειδί κρυπτογράφησης. Ανάλογα με το αν η συσκευή είναι MASTER ή SLAVE το αντίστοιχο SRES στέλνεται στη συσκευή στο άλλο άκρο.

Οι είσοδοι της συνάρτησης H5 είναι :

Είσοδος	**Μέγεθος σε bits**
S	128
R_1	128
R_2	128

Όπου S είναι το κλειδί πιστοποίησης που υπολογίζεται από την συνάρτηση H4 ενώ R1 και R2 είναι δύο ψευδό-τυχαίες τιμές των 128 bit, οι οποίες παράγονται η μία στον MASTER και η άλλη στον SLAVE.

Η έξοδος της συνάρτησης υπολογίζεται από τον παρακάτω τύπο :

$$h5(W, R_1, R_2) = HMAC - SHA - 256_T(R_1||R_2)/2^{128}$$

12. Η ΣΥΝΑΡΤΗΣΗ H6 και H7

Αντίθετα με τις παραπάνω συναρτήσεις, η συγκεκριμένη συνάρτηση χρησιμοποιεί τον κρυπτογραφικό αλγόριθμο AES σε CMAC μορφή και χρησιμοποιείται για την μετατροπή του τύπου του κλειδιού σύνδεσης.

Οι είσοδοι της συνάρτησης H6 είναι : μια μεταβλητή W των 128 bits που χρησιμοποιείται σαν κλειδί στον AES-CMAC καθώς και μια σταθερά τιμή των 32 bits KeyID(“tmp2”).

Η έξοδος υπολογίζεται από τον παρακάτω τύπο :

$$h6(W, keyID) = AES - CMAC_W(keyID)$$

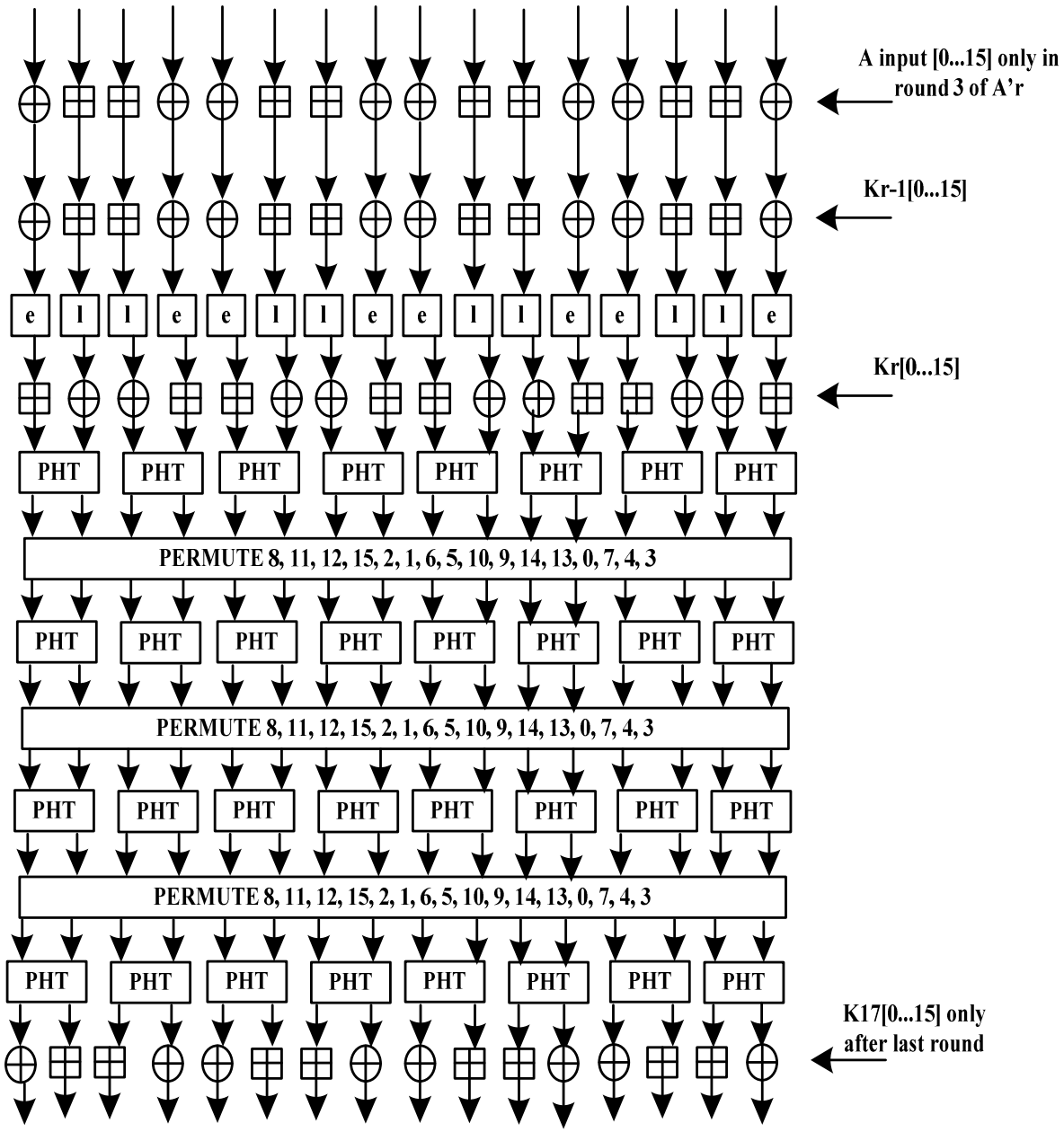
Η συνάρτηση H7 είναι παρόμοια με την H6 χρησιμοποιώντας τον ίδιο αλγόριθμο AES- CMAC. Η βασική διαφορά με την H6 είναι στις εισόδους της. Σαν κλειδί παίρνει την μεταβλητή SALT ενώ η μεταβλητή W αντιμετωπίζεται σαν Plaintext.

Η έξοδος της H7 υπολογίζεται από τον ακόλουθο τύπο:

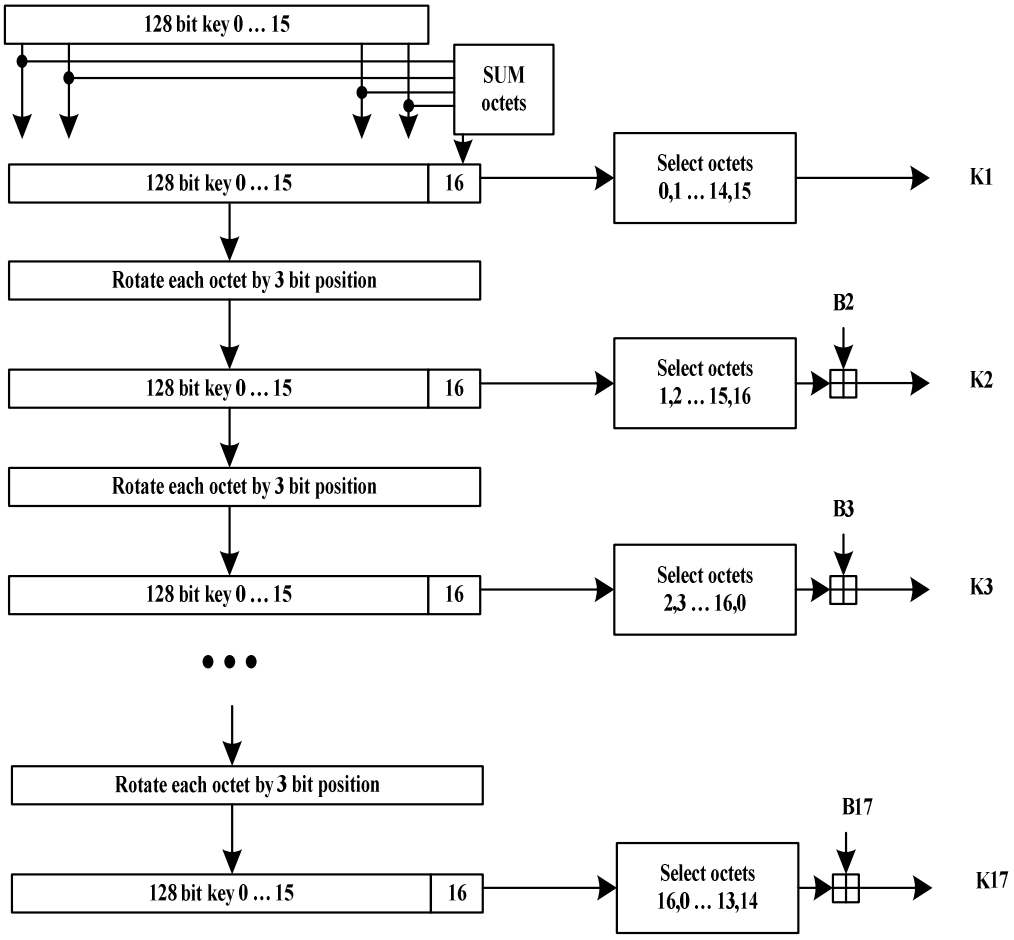
$$h7(SALT, W) = AES - CMAC_{SALT}(W)$$

13. Ο ΚΡΥΠΤΟΓΡΑΦΙΚΟΣ ΑΛΓΟΡΙΘΜΟΣ SAFER+

Ο SAFER+ είναι ένας αλγόριθμος και ανήκει στην οικογένεια των block ciphers. Ο SAFER+ περιλαμβάνει χαρακτηριστικά από τους αλγόριθμους SAFER-K64, SAFER K-128 και SAFER-SK128 και αναπτύχθηκε από τον James L. Massey. Ο αλγόριθμος αυτός δεν μοιάζει με αυτούς που έχουν Feistel δομή αλλά ούτε και με εκείνους τους αλγορίθμους που χρησιμοποιούν την τεχνική της αντικατάστασης(Substitution - Ciphers). Πιο συγκεκριμένα, ο SAFER+ χρησιμοποιεί πρώτον μια διαφορετική τεχνική που την ονομάζει PHT(Pseudo-Hadamard-Transformation) και δεύτερον χρησιμοποιεί επιπλέον σταθερές, τις Bias-Factors που τις προσθέτει στην διαδικασία παραγωγής υπό-κλειδιών. Με αυτόν τον τρόπο επιτυγχάνεται η αποφυγή παραγωγής αδύναμων κλειδιών. Κάθε γύρος χρησιμοποιεί 128 bit υπό-κλειδιά που παράγονται από ένα μυστικό κλειδί των 128 bits. Ο SAFER+ αποτελεί κομμάτι της ασφάλειας του Bluetooth από την πρώτη έκδοση v1 και χρησιμοποιείται στις συναρτήσεις E1 και E3 στις διαδικασίες πιστοποίησης και παραγωγής κλειδιού κρυπτογράφησης. Στην εικόνα 6 βλέπουμε τον γύρο του Safer+ που επαναλαμβάνεται 8 φορές καθώς και τις πράξεις που γίνονται στις συναρτήσεις “e” και “l”, ενώ στην εικόνα 7 βλέπουμε την επεξεργασία του μυστικού κλειδιού για την παραγωγή των υπό-κλειδιών.



Εικόνα 6: Ο επαναλαμβανόμενος γύρος του Safer+



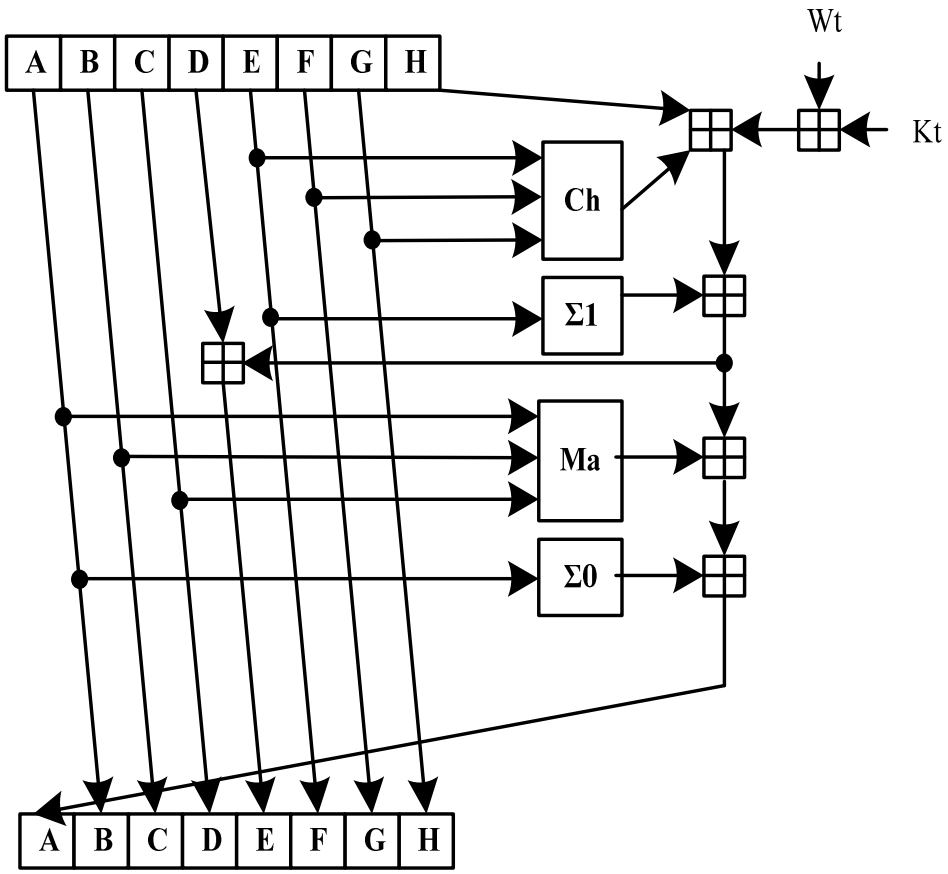
Εικόνα 7: Η παραγωγή υπό-κλειδιών του Safer+

14. Η ΣΥΝΑΡΤΗΣΗ ΚΑΤΑΚΕΡΜΑΤΙΣΜΟΥ SHA-256

Ο SHA-256 είναι μία μονόδρομη συνάρτηση κατακερματισμού από τις πιο ισχυρές που συναντάμε σε σύγχρονες εφαρμογές και είναι ευρύτερα χρησιμοποιούμενος σε μηχανισμούς MAC για πιστοποίηση μηνυμάτων. Αξίζει να σημειωθεί ότι χρησιμοποιείται σαν βασικός πυρήνας στο bitcoin ένα από τα πιο διαδεδομένα κρυπτό-νομίσματα στο κόσμο.

Όπως σε όλες τις μονόδρομες συναρτήσεις κατακερματισμού έτσι και εδώ έχουμε μια είσοδο ανεξάρτητου μήκους ενώ ο αλγόριθμος παράγει μια έξοδο συγκεκριμένου

μήκους ίση με 256 bits. Ο SHA-256 λέγεται ασφαλής καθώς είναι υπολογιστικά ανέφικτο να βρεθεί το αρχικό μήνυμα με βάση την έξοδο των 256 bits ή έστω να βρεθούν δύο διαφορετικά μηνύματα που να παράγουν την ίδια τιμή εξόδου. Έτσι η παραμικρή αλλαγή στα bytes ενός μηνύματος θα οδηγήσει σε εξαιρετικά διαφορετική τιμή εξόδου την οποία θα μπορούσε εύκολα ο παραλήπτης να διαπιστώσει συγκρίνοντας το hash που λαμβάνει με το hash που παράγει. Στην εικόνα 8 βλέπουμε το πώς επεξεργάζεται ο αλγόριθμος τα δεδομένα.



Εικόνα 8: Δομή SHA-256

Αφού αρχικοποιηθούν οι 32 bit τιμές A, B, C, D, E, F, G, H τότε ξεκινά η επανάληψη του αλγορίθμου. Ο αλγόριθμος επεξεργάζεται το μήνυμα σε blocks των 512 bit, με το τελευταίο block να περνά την διαδικασία Padding. Ο συνολικός αριθμός επαναλήψεων είναι ίσος με 64 ενώ οι συναρτήσεις που χρησιμοποιούνται στον γύρο περιγράφονται παρακάτω.

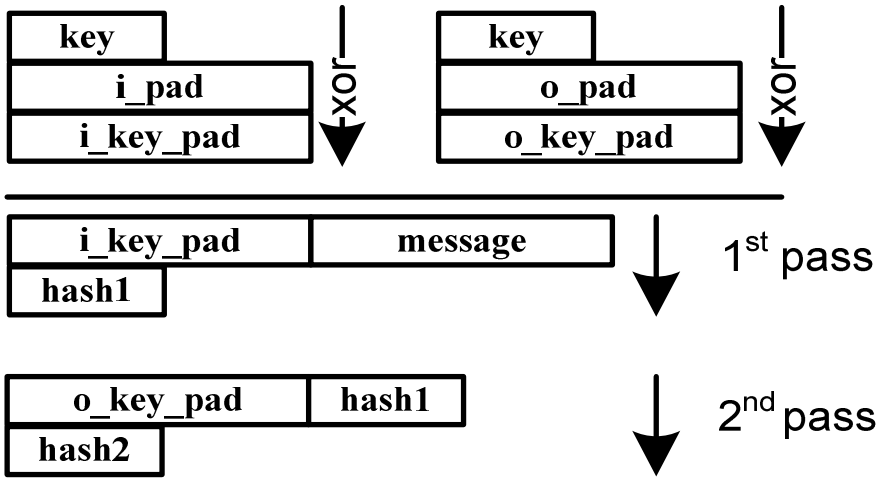
$Ch_t(X,Y,Z) = (X^Y) \oplus ((\dashv X)\hat{}\ Z)$, για t=0 έως t=63

$Maj_t(X,Y,Z) = (X\ \hat{}\ Y) \oplus (X\hat{}Z) \oplus (Y\hat{}Z)$, για t=0 έως t=63

$\Sigma_0^{\{256\}}(X) = (X \ggg 2) \oplus (X \ggg 13) \oplus (X \ggg 22)$, για t=0 έως t=63

$\Sigma_1^{\{256\}}(X) = (X \ggg 6) \oplus (X \ggg 11) \oplus (X \ggg 25)$, για t=0 έως t=63

Τέλος, στην ασφάλεια του Bluetooth ο SHA256 χρησιμοποιείται σε MAC μορφή στην διαδικασία πιστοποίησης και παραγωγής κλειδιού κρυπτογράφησης όταν το σύστημα απαιτεί κρυπτογράφηση των δεδομένων με τον αλγόριθμο AES. Όταν οι συναρτήσεις κατακερματισμού χρησιμοποιούνται σε MAC μορφή, τότε το ονομάζουμε HMAC. Στην εικόνα 9 βλέπουμε το πώς ένα μήνυμα περνά την διαδικασία HMAC με τον αλγόριθμο SHA-256.

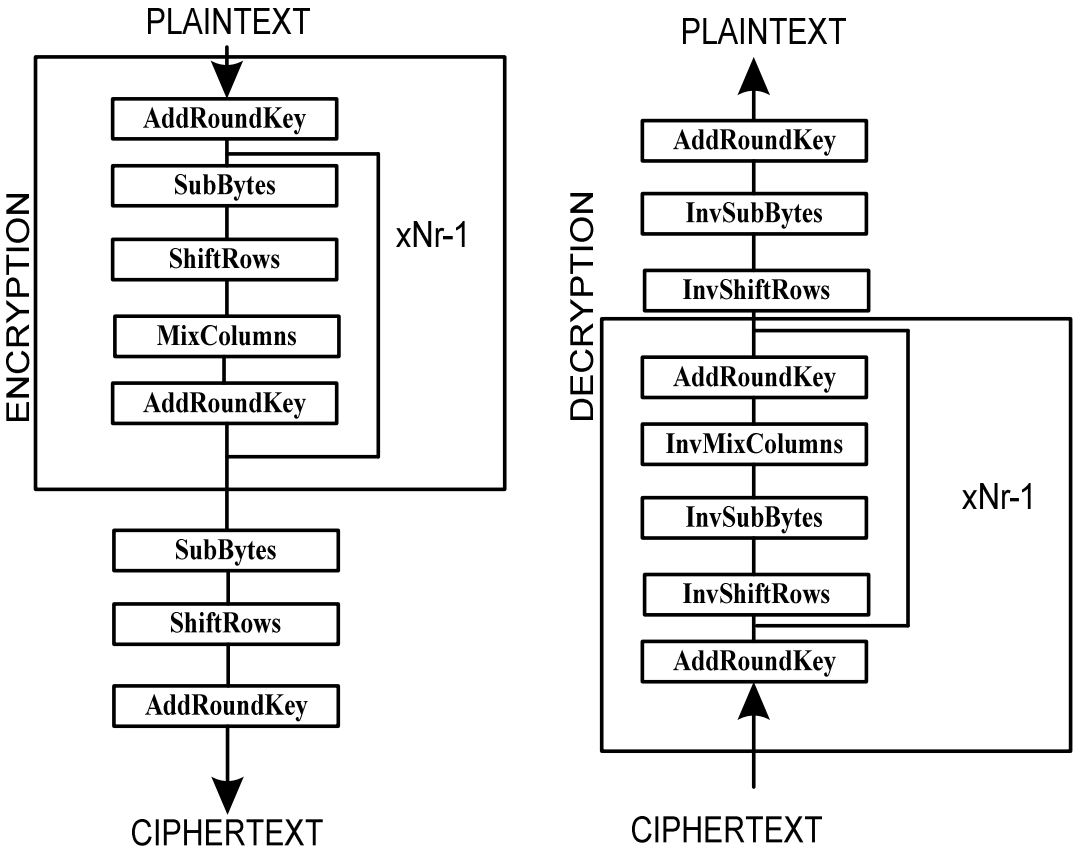


Εικόνα 9: Μετασχηματισμός HMAC

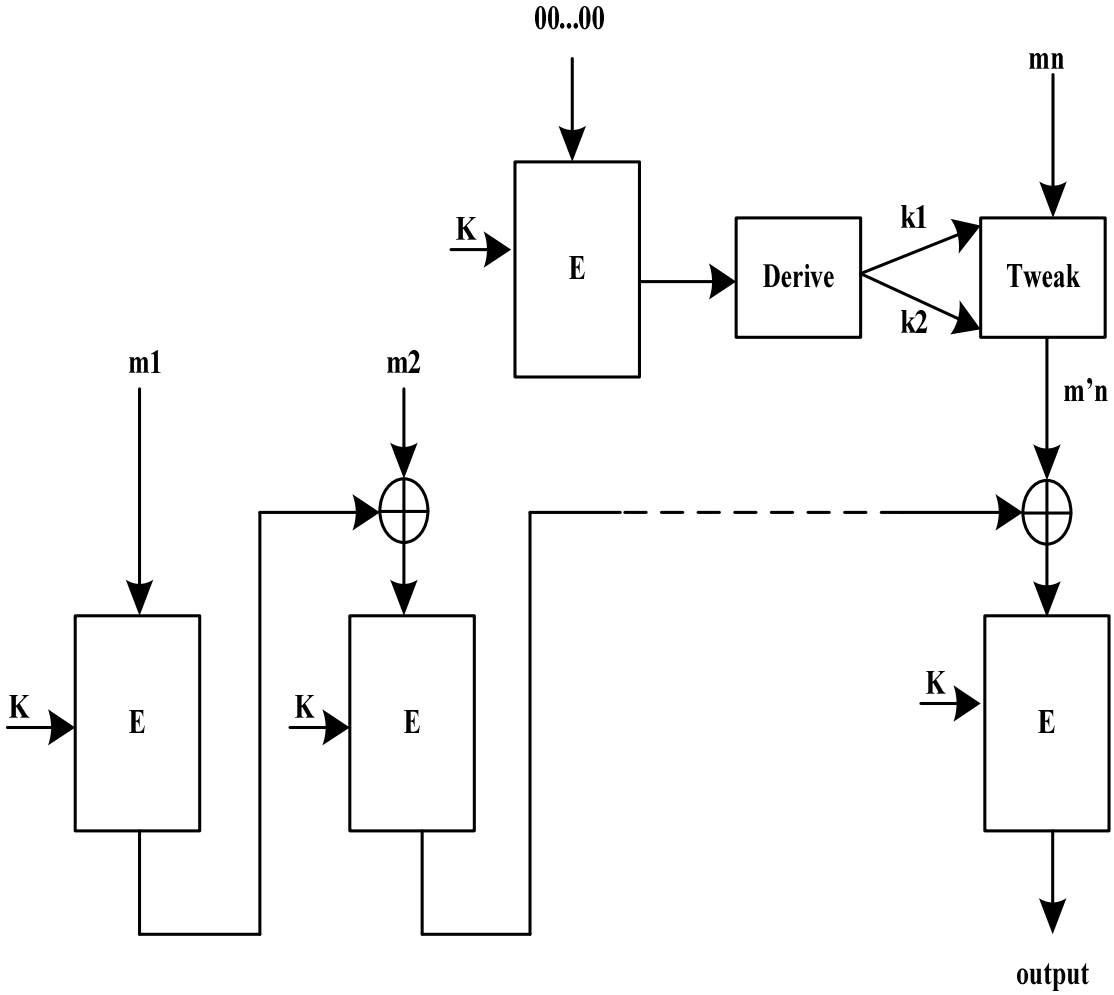
15. Ο ΚΡΥΠΤΟΓΡΑΦΙΚΟΣ ΑΛΓΟΡΙΘΜΟΣ AES

Ο AES αλγόριθμος δεν βασίζεται στα Feistel δίκτυα, αλλά στη χρήση blocks όπου κάθε block έχει συγκεκριμένο μήκος της τάξης των 16 bytes, ενώ τα κλειδιά που μπορούν να χρησιμοποιηθούν είναι μεταβλητού μήκους. Τα κλειδιά μπορούν να έχουν μέγεθος 128 ή 192 ή 256. Από το μέγεθος του κλειδιού εξαρτάται και ο αριθμός των γύρων. Πιο συγκεκριμένα, για κλειδί ίσο με 128 ο αριθμός των γύρων είναι ίσος με 10, για κλειδί ίσο με 192 είναι 12 ενώ για κλειδί ίσο με 256 είναι 14. Ο AES χρησιμοποιείται στις πιο σύγχρονες εφαρμογές και θεωρείται από τους πιο ισχυρούς αλγόριθμους έως σήμερα. Το μέγεθος του κλειδιού καθιστά αδύνατο να βρεθεί ένα κλειδί από το κρυπτό-κείμενο με την μέθοδο ωμής βίας, αφού για παράδειγμα για κλειδί ίσο με 128 bits υπάρχουν $3.4 * 10^{38}$ πιθανοί συνδυασμούς. Όσον αφορά την αποκρυπτογράφηση η διαδικασία είναι η αντίστροφη σε σχέση με την διαδικασία κρυπτογράφησης. Ο

αλγόριθμος AES στην ασφάλεια του Bluetooth χρησιμοποιείται σε μορφή CMAC. Ο AES-CMAC στην ουσία είναι μια MAC πιστοποίηση που χρησιμοποιεί τον αλγόριθμο AES. Δέχεται μια είσοδο μηνύματος μεταβλητού μήκους και παράγει μια έξοδο συγκεκριμένου μήκους. Το τελευταίο block εισόδου περνά την διαδικασία padding στην περίπτωση που δεν είναι 128 bits. Στην εικόνα 10 φαίνεται η δομή του AES ενώ στην εικόνα 11 βλέπουμε την δομή του AES-CMAC.



Εικόνα 10: Ο αλγόριθμος AES



Εικόνα 11: Πιστοποίηση AES-CMAC

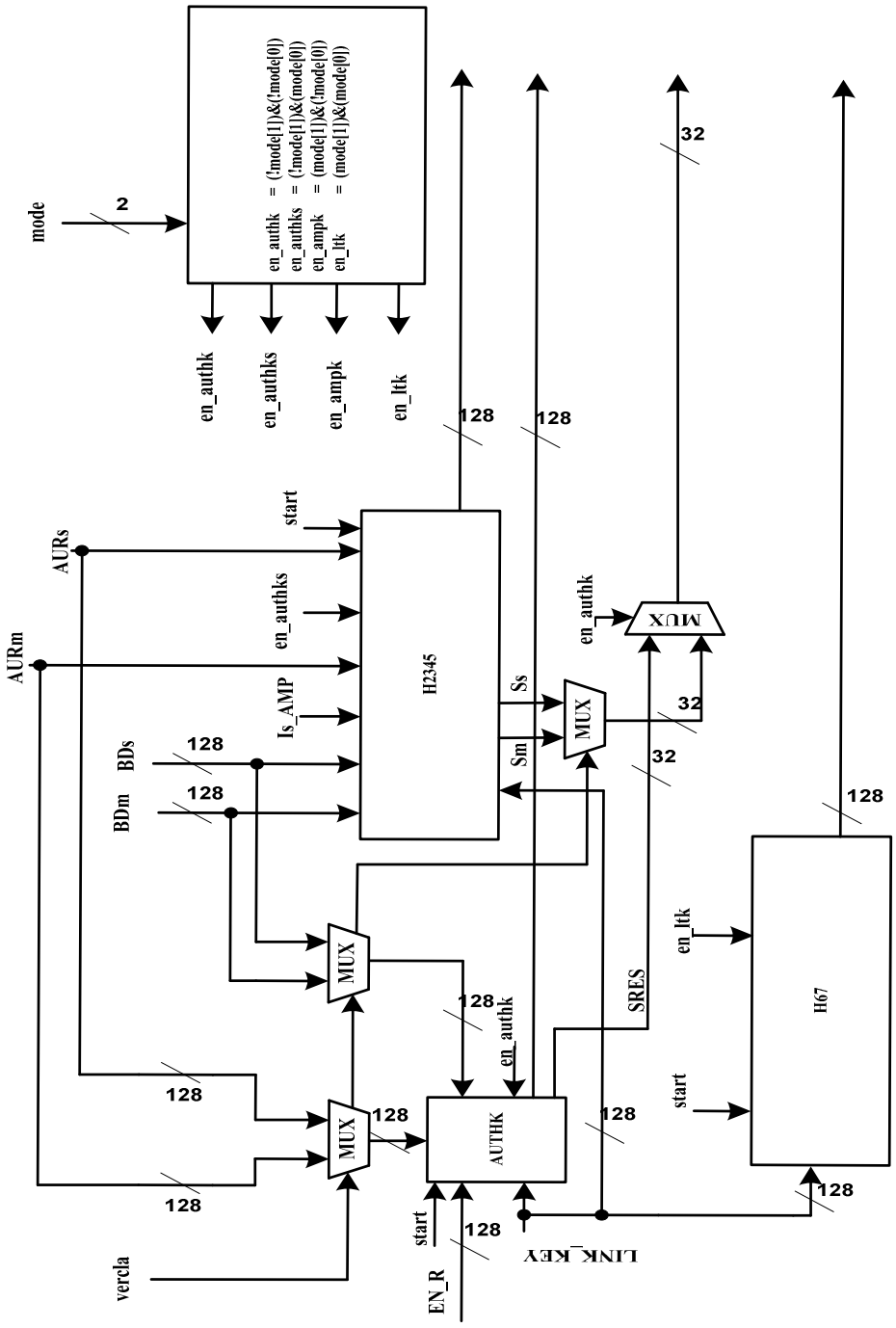
2. ΥΛΟΠΟΙΗΣΗ

1. ΕΙΣΑΓΩΓΗ

Όπως είδαμε και από την εικόνα 3 οι συναρτήσεις που χρησιμοποιούνται στην διαδικασία πιστοποίησης και παραγωγής κλειδιού κρυπτογράφησης βασίζονται στους αλγόριθμους κατακερματισμού και cipher HMAC-256, SAFER+ και AES-CMAC. Επίσης, παρατηρούμε ότι οι ενδιάμεσες συναρτήσεις ανάμεσα στις προαναφερθέντες διαδικασίες εξαρτώνται από τα αποτελέσματα των προηγούμενων συναρτήσεων. Για λόγους ελαχιστοποίησης των πόρων υλικού η υλοποίηση που έγινε χρησιμοποιεί την τεχνική επαναχρησιμοποίησης υλικού. Κάθε αλγόριθμος κρυπτογράφησης και κατακερματισμού έχει τοποθετηθεί από μια μόνο φορά στον κεντρικό μας πυρήνα, έτσι με τα κατάλληλα σήματα ελέγχου και έναν μικρό αριθμό flipflops καταφέραμε να χρησιμοποιήσουμε όσο το δυνατόν λιγότερο υλικό χωρίς να αυξήσουμε ιδιαίτερα των αριθμό κύκλων που απαιτούνται για την δημιουργία είτε του 32 bit σήματος SRES(πιστοποίηση ταυτότητας) είτε για το κλειδί κρυπτογράφησης KC . Παρακάτω περιγράφονται αναλυτικά οι αρχιτεκτονικές των αλγορίθμων που κατασκευάστηκαν καθώς επίσης και αποτελέσματα σύνθεσης και προσομοίωσης. Η κατασκευή των μονάδων έγινε στη γλώσσα περιγραφής υλικού verilog ενώ για τις μετρήσεις και την επαλήθευση ορθότητας της λειτουργίας χρησιμοποιήθηκε το FPGA της Xilinx Zeadboard.

2. Η ΜΟΝΑΔΑ KEYGEN

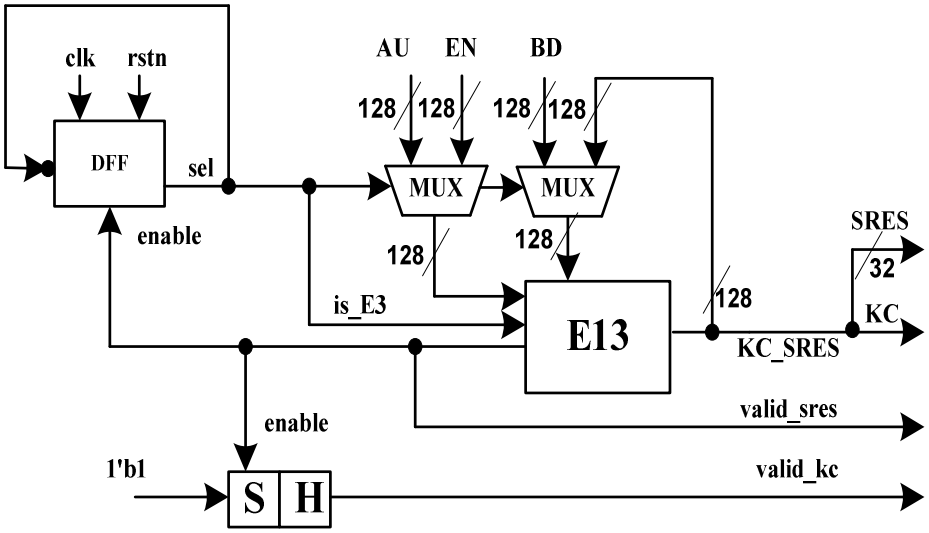
Η μονάδα KEYGEN αποτελεί τη πρώτη μονάδα στην ιεραρχία της υλοποίησης. Μέσα σε αυτή τοποθετούνται όλες οι μονάδες που χρειάζονται προκειμένου να υπολογιστούν : το κλειδί κρυπτογράφησης KC , το AMP Key και το Long Term LE key. Εκτός από τα σήματα εισόδου που χρειάζονται για τις συναρτήσεις, υπάρχουν και 2 επιπλέον σήματα : το σήμα vercla του ενός bit καθώς και το σήμα mode των 2 bits. Όταν η τιμή του σήματος vercla είναι "1" τότε ο επεξεργαστής μπαίνει σε λειτουργία claimant ενώ όταν είναι "0" τότε λειτουργεί σαν verifier. Οι δύο αυτές λειτουργίες αφορούν την διαδικασία πιστοποίησης. Πιο συγκεκριμένα, το σήμα vercla ελέγχει την έξοδο 2 πολυπλεκτών για να επιλεγεί η σωστή ψευδοτυχαία τιμή AU_RAND(σήμα AURm ή AURs) και η σωστή διεύθυνση BD_ADDR(σήμα BDm ή BDs) που αφορούν τις διαδικασίες πιστοποίησης και παραγωγής κλειδιού όταν λαμβάνει χώρα μια απλή ζεύξη μεταξύ των κόμβων. Το σήμα mode χρησιμοποιείται για να παράγει τα τέσσερα σήματα ενεργοποίησης (en_authk, en_authks, en_ampk, en_ltk), τα οποία το καθένα ξεχωριστά ενεργοποιεί μία από τις τρείς κύριες υπό-μονάδες, AUTHK, H2345 και H67. Στην εικόνα 12 φαίνεται το εσωτερικό της μονάδας KEYGEN, ενώ στο επάνω δεξιά μέρος της εικόνας βλέπουμε σε ψευδοκώδικα πως η μονάδα χειρίζεται τα δύο bits του σήματος mode για να παράγει τα σήματα ενεργοποίησης.



Εικόνα 12: Αρχιτεκτονική της Μονάδας KEYGEN

1. Η ΜΟΝΑΔΑ AUTHK

Η μονάδα επεξεργασίας AUTHK υλοποιεί την διαδικασία πιστοποίησης ταυτότητας σε μια απλή ασφαλή ζεύξη. Η μονάδα έχει σαν είσοδο τα σήματα των 128 bit AU, EN, BD και παράγει σαν έξοδο τα σήματα KC_SRES, valid_sres και valid_kc. Τα σήματα valid_sres και valid_kc ενεργοποιούνται όταν η πιστοποίηση ή η παραγωγή του κλειδιού κρυπτογράφησης βρίσκονται στην έξοδο αντίστοιχα. Πιο συγκεκριμένα, έπειτα από 34 κύκλους ρολογιού το σήμα valid_sres παίρνει την τιμή ένα και στα 32 πιο σημαντικά bits στην έξοδο KC_SRES βρίσκεται η τιμή SRES ενώ στους επόμενους 34 κύκλους ρολογιού το σήμα valid_kc θα πάρει την τιμή ένα και στο σήμα KC_SRES θα βρίσκεται το κλειδί κρυπτογράφησης. Δύο πολυπλέκτες ένα d-flipflop και ένας shifter χρησιμοποιούνται για να αλλάξουν τις εισόδους στην μονάδα E13 καθώς και την λειτουργία του. Το d-flipflop σε τιμή reset παράγει μηδέν στην έξοδο ενώ η έξοδος του είναι ταυτόχρονα και είσοδος του. Έτσι με μια πύλη not στην είσοδο και με enable σήμα το valid_sres παράγουμε ένα σήμα sel το οποίο ταλαντεύεται στις τιμές μηδέν και ένα ανά 34 παλμούς ρολογιού. Το σήμα sel χρησιμοποιείται σαν είσοδος στη E13 καθώς και σαν σήμα ελέγχου στους πολυπλέκτες. Τέλος, ο shifter αναλαμβάνει την παραγωγή του valid_kc όταν το σήμα valid από την μονάδα E13 έρθει σε κατάσταση ανερχόμενης παρυφής δύο φορές. Στην εικόνα 13 φαίνεται η αρχιτεκτονική της μονάδας AUTHK.



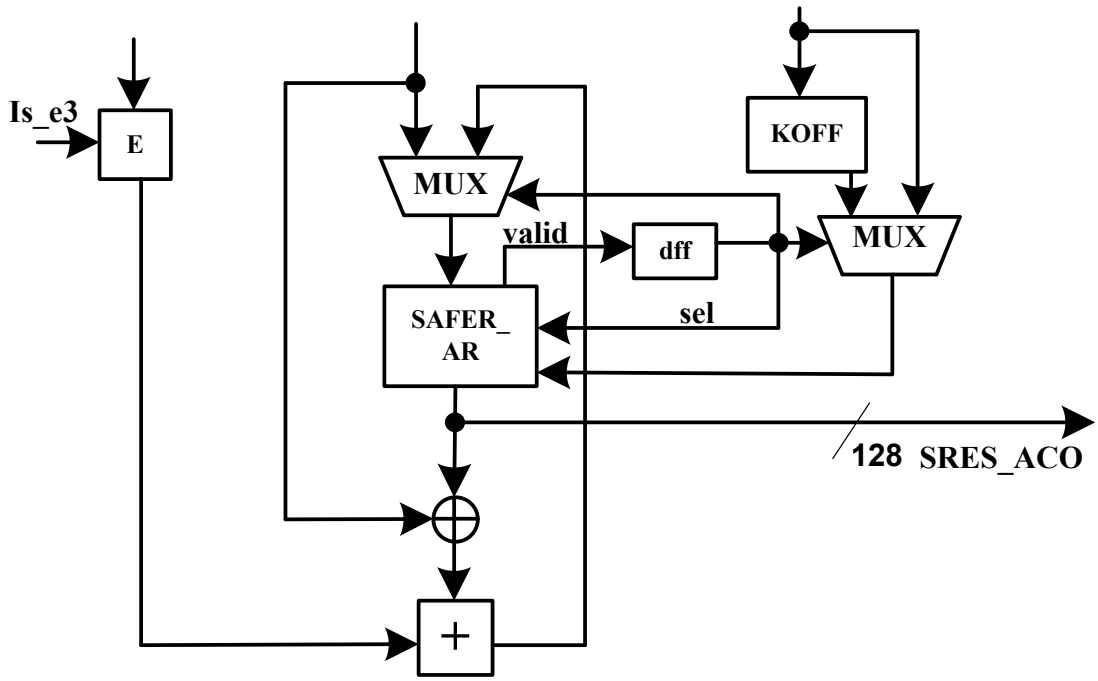
Εικόνα 13: Αρχιτεκτονική της Μονάδας AUTHK

2. Η ΜΟΝΑΔΑ E13

Η μονάδα e13 υλοποιεί τις συναρτήσεις E1 και E3 που αφορούν την διαδικασία πιστοποίησης και παραγωγής κλειδιού κρυπτογράφησης όταν τα δεδομένα πρόκειται να κρυπτογραφηθούν με stream cipher αλγόριθμο. Για το λόγο ότι οι δύο αυτές συναρτήσεις δεν έχουν πολύ μεγάλες διαφορές μεταξύ τους και καθώς η συνάρτηση E3 εξαρτάται από το αποτέλεσμα της συνάρτησης E1, απλοποιήσαμε την διαδικασία χρησιμοποιώντας μία μονάδα έναντι των δύο που απαιτούνται. Η αλλαγή μεταξύ της E1 και της E3 συνάρτησης αφορά την expansion διαδικασία η οποία υλοποιείται στην υπό-μονάδα E. Το σήμα εισόδου Is_e3 είναι υπεύθυνο για την αλλαγή της λειτουργίας από E1 σε E3.

Μία εξίσου παρόμοια ομοιότητα συναντάμε και στις υπό-συναρτήσεις Ar και A'r. Οι υπό-συναρτήσεις αυτές χρησιμοποιούν τον αλγόριθμο Sapher+, έτσι εδώ όπως και στην παραπάνω διαδικασία, η επεξεργασία της συνάρτησης A'r εξαρτάται από το

αποτέλεσμα της συνάρτησης Ar. Η υπό-μονάδα Sapher_Ar υλοποιεί αυτές τις δύο υπό-συναρτήσεις, ενώ το valid σήμα που μαρκάρει τα δεδομένα μας στην έξοδο θα περάσει από ένα D-FlipFlop με enable προκειμένου δημιουργηθεί το σήμα sel, το οποίο με την σειρά του θα χρησιμοποιηθεί σαν σήμα ελέγχου στον πολυπλέκτη εισόδου καθώς και σαν δείκτης στην μονάδα Sapher+ για την αλλαγή της συνάρτησης από Ar σε A'r. Το κλειδί που θα χρησιμοποιηθεί στην εκάστοτε συνάρτηση είναι η έξοδος από τον δεύτερο πολυπλέκτη ο οποίος με σήμα ελέγχου το σήμα sel θα επιλέξει ανάμεσα στο κλειδί που εισάγεται και στο κλειδί που επεξεργάζεται στην υπό-μονάδα KOFF. Οι υπό-μονάδες KOFF και E είναι μονάδες συνδυαστικής λογικής και δεν εξαρτώνται από FLIPFLOPs. Ο υπολογισμό της υπό-μονάδας KOFF περιγράφεται στην εικόνα 4, ενώ η υπό-μονάδα E επαναλαμβάνει τα bytes εισόδου ξεκινώντας από το πιο σημαντικό ώστε να φτιάξει ένα σήμα ίσο με 128 bits. Στην εικόνα 14 βλέπουμε το εσωτερικό της μονάδας E13, ενώ στην εικόνα 15 φαίνεται το εσωτερικό της μονάδας Sapher_Ar καθώς και η υλοποίηση του Sapher+ αλγορίθμου.



Εικόνα 14: Αρχιτεκτονική της Μονάδας E13

3. Η ΜΟΝΑΔΑ SAFER_AR

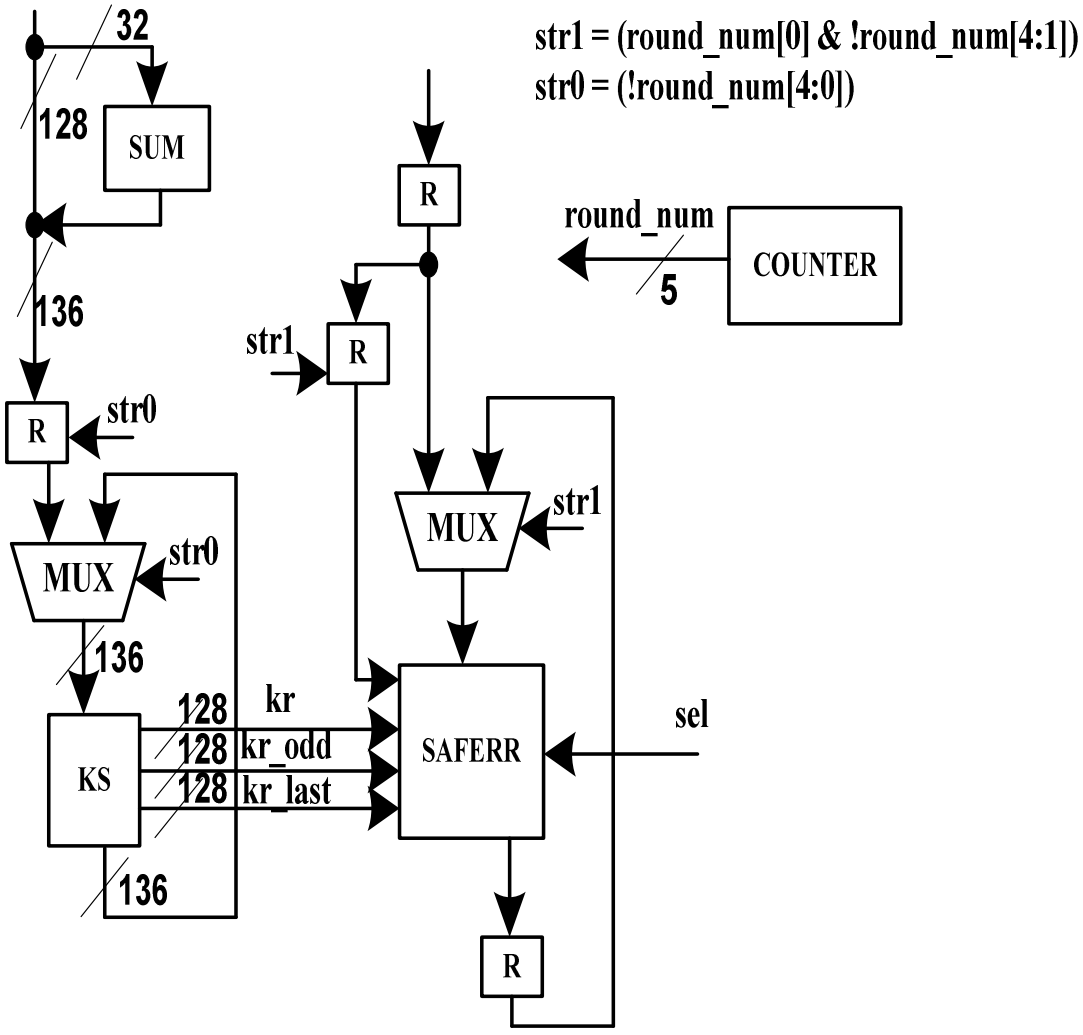
Όπως αναφέραμε και παραπάνω η μονάδα SAFER_AR αναλαμβάνει την υλοποίηση των συναρτήσεων Ar και A'r. Η μόνη διαφορά των συναρτήσεων Ar και A'r είναι ότι στον τρίτο γύρο του Safer+ στην συνάρτηση A'r γίνονται οι πράξεις πρόσθεσης και λογικής XOR ανάμεσα στην τιμή του πρώτου γύρου και του τρέχοντος γύρου, όπως είδαμε νωρίτερα στις προδιαγραφές του Bluetooth Safer+ (εικόνα 5). Στην μονάδα SAFER_AR ένας μετρητής μετρά από το μηδέν μέχρι το δεκαεπτά και λειτουργεί σαν δείκτης στους 16 γύρους του SAFER+. Τα δεδομένα αποθηκεύονται σε έναν register με enable σήμα το αποτέλεσμα τις λογικής πράξης : (round_num[0] & !round_num[4:1]) και κρατούνται στην έξοδο του register η οποία είναι μια επιπλέον είσοδος στην υπό-μονάδα

SAFERR, ενώ ένας πολυπλέκτης επιλέγει την έξοδό του ανάλογα με το αν ο γύρος του Safer+ είναι ο πρώτος ή όχι. Η έξοδός της μονάδας SAFERR φιλτράρεται από έναν register προκειμένου να δρομολογηθούν τα bytes εξόδου στην είσοδό του στον επόμενο κύκλο ρολογιού. Στην εικόνα 14 φαίνεται η αρχιτεκτονική της μονάδας SAFER_AR.

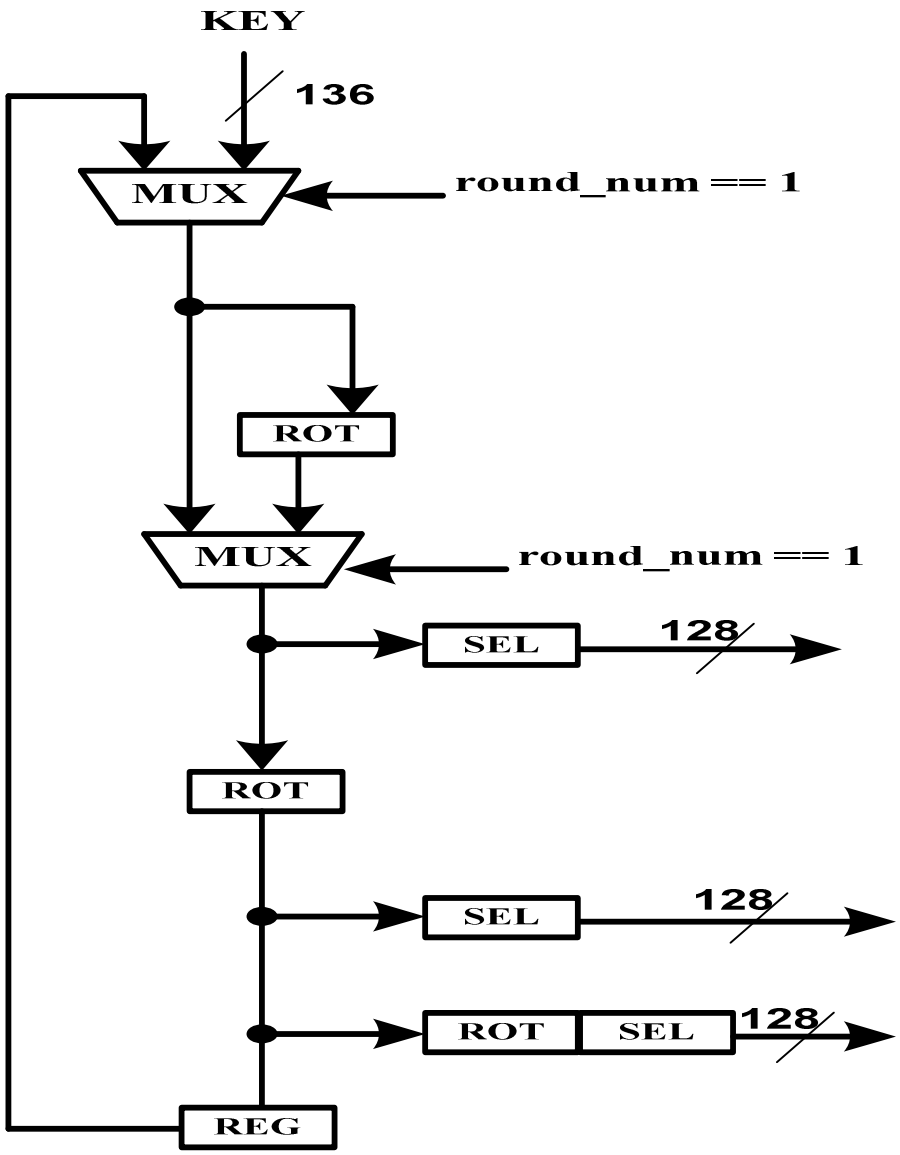
Η υπό-μονάδα SAFERR υλοποιεί με την σειρά της τον επαναλαμβανόμενο γύρο του Safer+ αλγόριθμου έχοντας σαν είσοδο τα δύο υποκλείδια που χρειάζονται για κάθε γύρο. Το τελευταίο κλειδί υπολογίζεται όταν ο δείκτης round_num είναι ίσος με τον δεκαδικό αριθμό 17. Τα δεδομένα εισόδου text καθώς και το σήμα sel δρομολογούνται από την υψηλότερη στην ιεραρχία μονάδα E13. Η λογική στο εσωτερικό του SAFERR είναι συνδυαστική ενώ οι υπό-μονάδες e και l είναι προϋπολογισμένες για όλες τις πιθανές εισόδους και υλοποιημένες σε ROM μνήμες. Ο πολλαπλασιασμός x2 μέσα στην PHT υλοποιείται με την διαδικασία της αριστερής ολίσθησης. Στην εικόνα 17 φαίνεται το εσωτερικό της μονάδας SAFERR και της υπό-μονάδας PHT.

Σύμφωνα με τις προδιαγραφές του αλγορίθμου Safer+ που είδαμε νωρίτερα το κλειδί για να επεξεργαστεί πρέπει πρώτα να περάσει μια μικρή προ-επεξεργασία. Την προ-επεξεργασία αυτή την αναλαμβάνει η υπό-μονάδα SUM κατά την οποία τα bytes 15,14,1 και 0 του εισερχόμενου κλειδιού περνάνε και επεξεργάζονται σε αυτήν. Η λογική της είναι συνδυαστική και το μόνο που κάνει είναι να προσθέτει τα bytes μεταξύ τους προκειμένου να παράγει ένα επιπλέον byte που προσκολλάται στο κύριο κλειδί μετατρέποντας το σε 17 bytes (136 bits). Ένας register συγχρονίζει το κλειδί πριν αυτό φιλτραριστεί από τον πολυπλέκτη για να περάσει στην μονάδα KS. Ο πολυπλέκτης με τη σειρά του αφήνει να περάσει στην έξοδο του το εισερχόμενο κλειδί μόνο όταν ο δείκτης

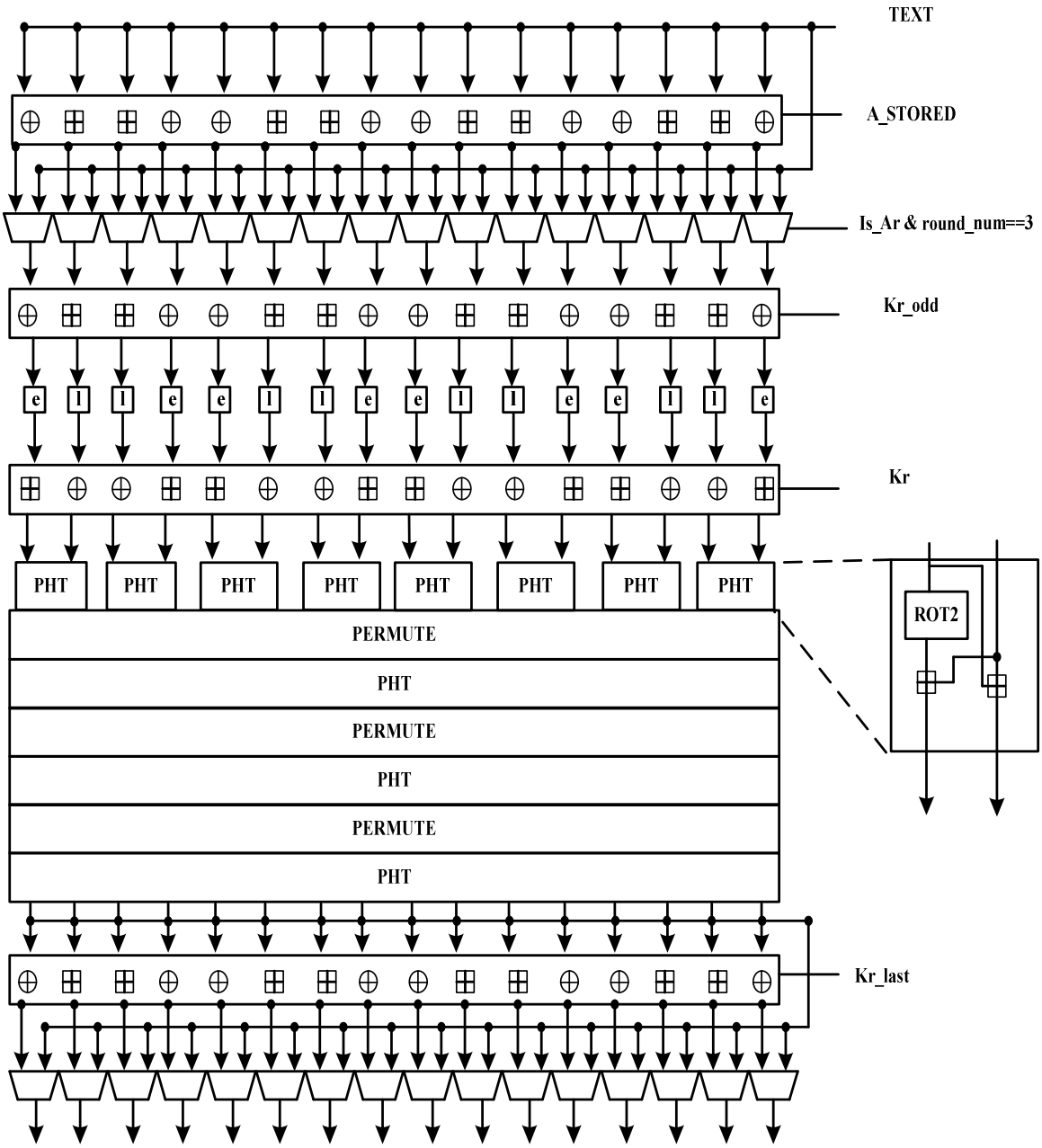
round_num είναι 1. Η μονάδα KS υλοποιεί την διαδικασία παραγωγής υποκλείδιων και το εσωτερικό της φαίνεται στην εικόνα 16.



Εικόνα 15: Αρχιτεκτονική της Μονάδας SAFER_AR



Εικόνα 16: Αρχιτεκτονική της υπό-μονάδας KS



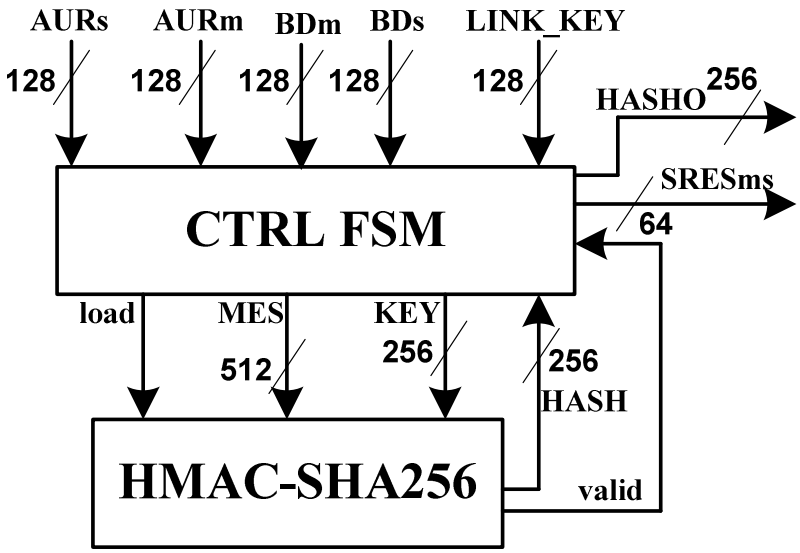
Εικόνα 17: Αρχιτεκτονική της Μονάδας SAFERR

4. Η ΜΟΝΑΔΑ H2345

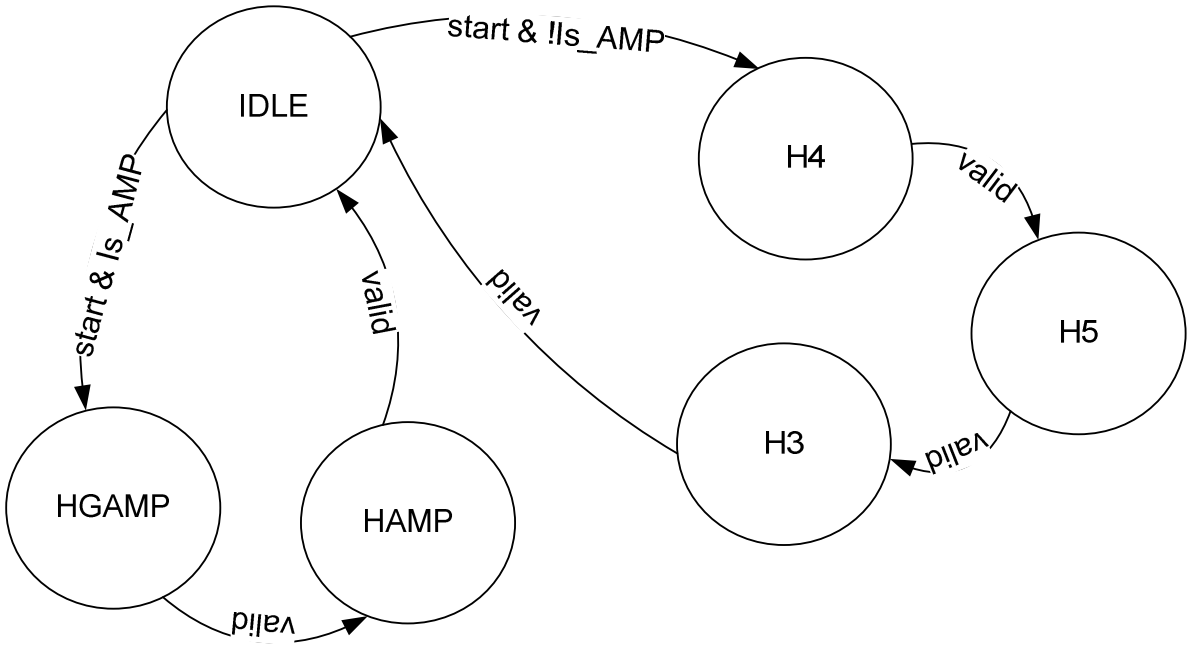
Η μονάδα αυτή αποτελεί την πιο πολύπλοκη διαδικασία στην υλοποίηση μας καθώς υλοποιεί την παραγωγή 2 κλειδιών, του κλειδιού κρυπτογράφησης όταν τα δεδομένα κρυπτογραφούνται με τον αλγόριθμο AES-CCM και του AMP_KEY που χρησιμοποιείται για να υποστηρίξει την ασφάλεια σε πρωτόκολλα υψηλότερου ρυθμού μετάδοσης. Αξίζει να σημειωθεί ότι στην συγκεκριμένη μονάδα έγιναν και οι περισσότερες βελτιστοποιήσεις καθώς οι δύο διαδικασίες χρησιμοποιούν συναρτήσεις που βασίζονται στον ίδιο αλγόριθμο, τον HMAC-SHA256 ο οποίος χρησιμοποιείται 5 φορές.

Πιο συγκεκριμένα, η μονάδα αυτή έχει σαν εισόδους τα σήματα LINK_KEY, AURm, AURs, BDm, BDs και ένα σήμα Is_AMP το οποίο όταν είναι ένα η λειτουργία της μονάδας δουλεύει για να παράγει το AMP_KEY ενώ όταν είναι μηδέν η μονάδα λειτουργεί για να παράγει το κλειδί κρυπτογράφησης. Στην εικόνα 18 βλέπουμε την αρχιτεκτονική της μονάδας H2345. Ο κεντρικός πυρήνας είναι η μονάδα HMAC-SHA256, η οποία υλοποιεί την διαδικασία του αλγορίθμου. Για τον σχηματισμό της κατάλληλης εισόδου χρησιμοποιούμε μία FSM, της οποίας η λειτουργία περιγράφεται στην εικόνα 19. Η FSM είναι υπεύθυνη εκτός από την συναρμολόγηση των εισόδων για την μονάδα HMAC-SHA256 να παράγει και το σήμα load. Οι σταθερές brle και btak που χρειάζονται είναι αποθηκευμένες μέσα στην FSM προκειμένου να χρησιμοποιηθούν όταν η κατάσταση βρίσκεται στην H4 ή H3 αντίστοιχα. Το load σήμα χρησιμοποιείται για να εισάγουμε το μήνυμα στον αλγόριθμο HMAC-SHA256. Όταν το σήμα load είναι ένα τότε εισάγεται το μήνυμα σε κομμάτια των 512 bits. Όταν το σήμα load γίνει 0 τότε ξεκινά και η επεξεργασία του αλγορίθμου. Η διαδικασία για την παραγωγή της

πιστοποίησης, δηλαδή της παραγωγής των σημάτων SRESm και SRESs απαιτεί 538 κύκλους ρολογιού, ενώ η παραγωγή του κλειδιού απαιτεί 804 κύκλους.



Εικόνα 18: Αρχιτεκτονική της Μονάδας H2345

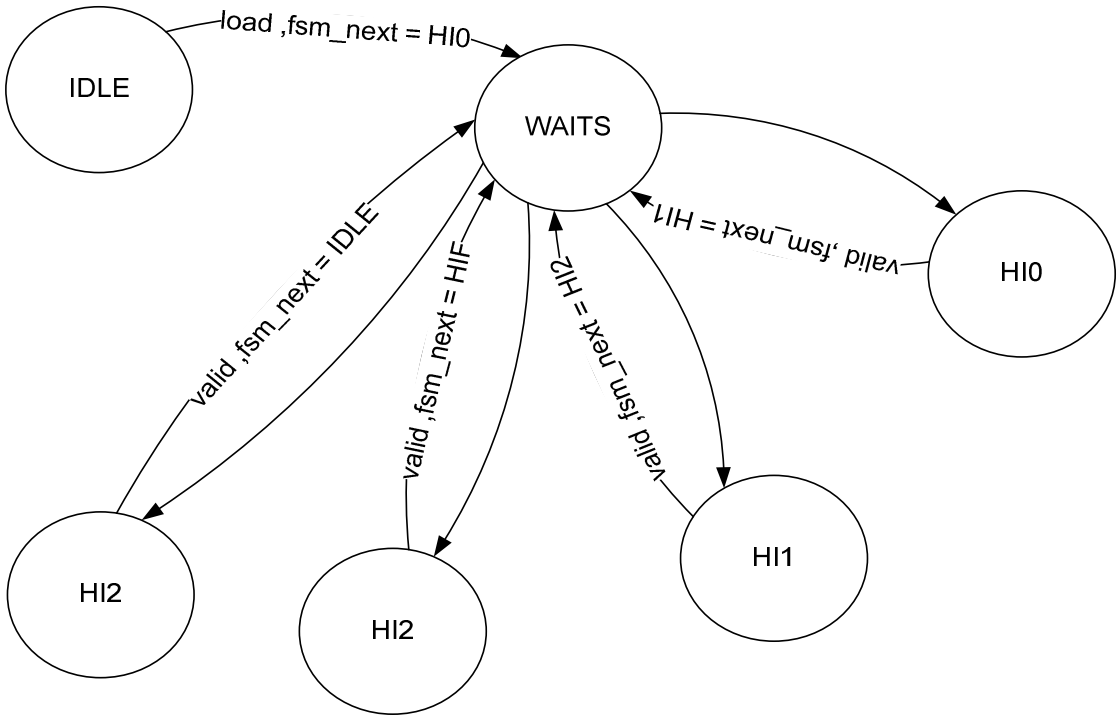


Εικόνα 19: Λειτουργία FSM στην μονάδα H2345

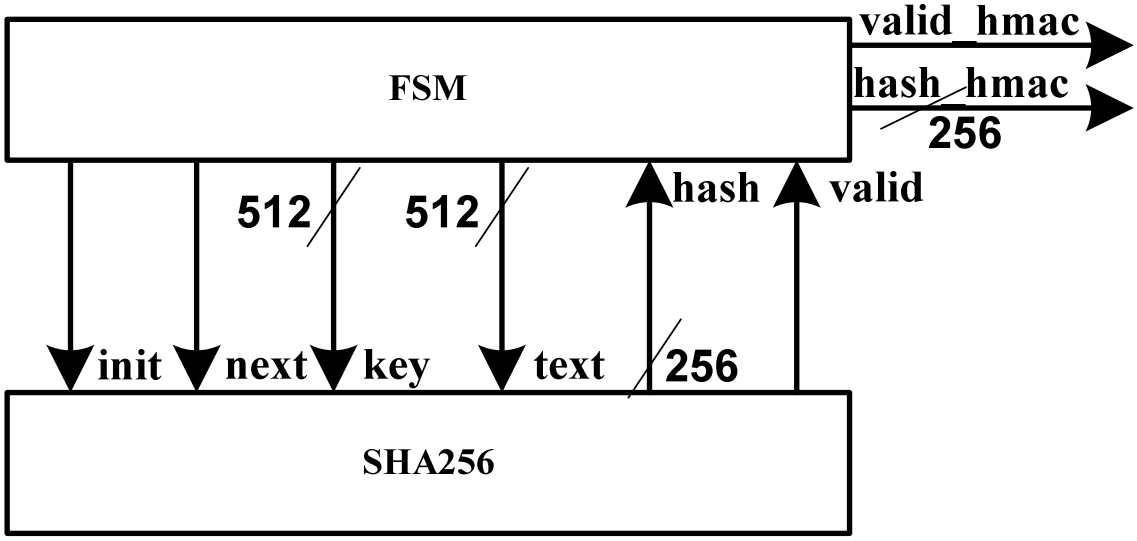
5. Η ΜΟΝΑΔΑ HMAC-SHA256

Η μονάδα αυτή όπως έχει αναφερθεί και παραπάνω χρησιμοποιείται για να υλοποιήσει τον αλγόριθμο SHA256 σε λειτουργία MAC. Η μονάδα HMAC-SHA256 αποτελείται από δύο υπό-μονάδες, την μονάδα SHA256 που υλοποιεί τον αλγόριθμο κατακερματισμού sha256 και μια μονάδα FSM η οποία αναλαμβάνει τον υπολογισμό των λογικών πράξεων XOR με της σταθερές i_pad και o_pad. Επίσης. διαχειρίζεται τα δεδομένα στην είσοδο text και στο κλειδί key που εισέρχονται στην υπό-μονάδα αυτή. Οι καταστάσεις HI0, HI1, HI2 και HIF λειτουργούν για να αλλάξουν τις εισόδους στη SHA256 ενώ η κατάσταση της FSM αλλάζει όταν το σήμα valid από την υπό-μονάδα SHA256 έρθει στην τιμή 1 δηλαδή όταν η hash τιμή από την SHA256 είναι έτοιμη. Ο σχεδιασμός της FSM σε αυτήν την μονάδα είναι διαφορετικός από την προηγούμενη για το λόγο ότι απαιτείται ένας επιπλέον κύκλος ρολογιού για να ξεκινήσει η διαδικασία παραγωγής hash του επόμενου block. Αυτό επιτυγχάνεται με την δημιουργία μιας ενδιάμεσης κατάστασης που την ονομάζουμε WAITS. Κάθε φορά που τελειώνει η δημιουργία ενός hash το σήμα valid αλλάζει την κατάσταση σε WAITS, ενώ ένας register των 3 bits κρατάει την τιμή της επόμενης κατάστασης. Εκτός από τις εισόδους text και key υπάρχουν δύο επιπλέον σήματα init και next που παίζουν καθοριστικό ρόλο στη σωστή λειτουργία του HMAC-SHA256. Το σήμα init αναλαμβάνει να ενημερώσει την υπό-μονάδα SHA256 ότι το block που εισέρχεται είναι το πρώτο, ενώ το σήμα next ότι πρόκειται για επόμενο block.. Τέλος, ο αριθμός των κύκλων που απαιτούνται στη συγκεκριμένη μονάδα εξαρτάται προφανώς από τον αριθμό των blocks του μηνύματος. Στη περίπτωση του Bluetooth όμως τα blocks έχουν συγκεκριμένο αριθμό σύμφωνα με τις προδιαγραφές των συναρτήσεων που αναλύσαμε νωρίτερα και για το λόγο αυτό το

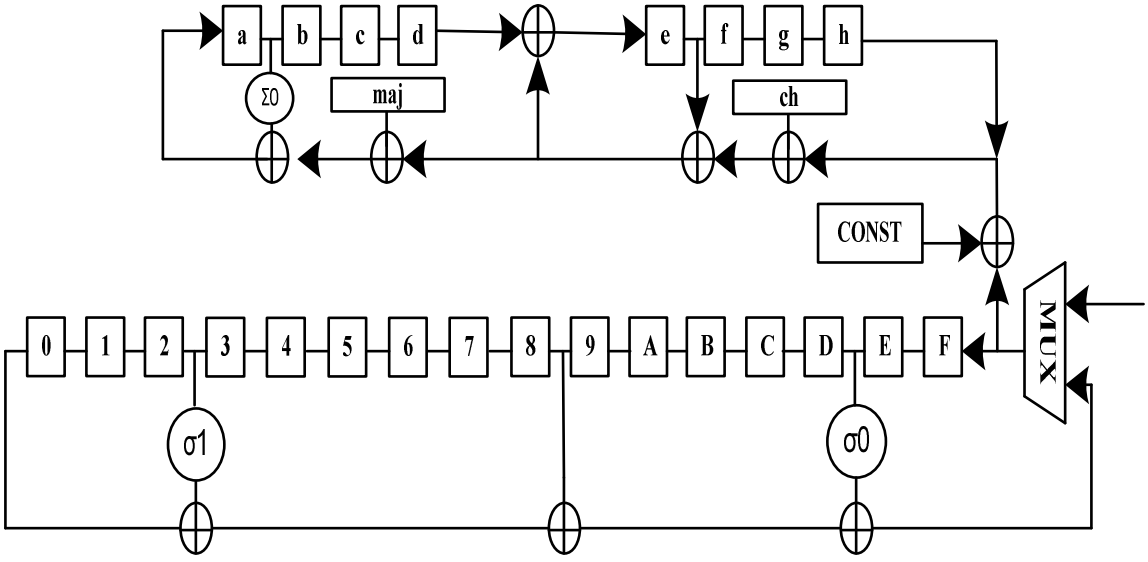
επίπεδο πολυπλοκότητας έχει συγκεκριμένο όριο. Οι εικόνες 21 και 22 δείχνουν τις μονάδες HMAC-SHA256 και SHA256 αντίστοιχα, ενώ η λειτουργία της FSM καθώς και η αρχιτεκτονική της υπό-μονάδας SHA256 φαίνεται στην εικόνα 20 και 22 αντίστοιχα.



Εικόνα 20: Λειτουργία FSM στην μονάδα HMAC-SHA256



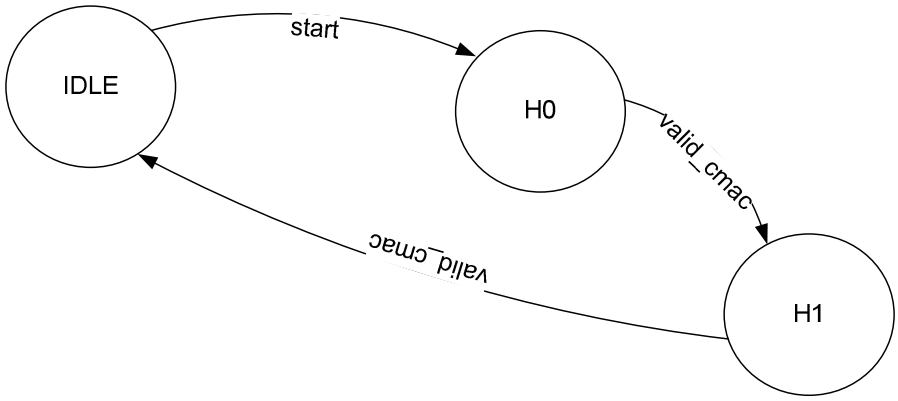
Εικόνα 21: Η μονάδα HMAC-SHA256



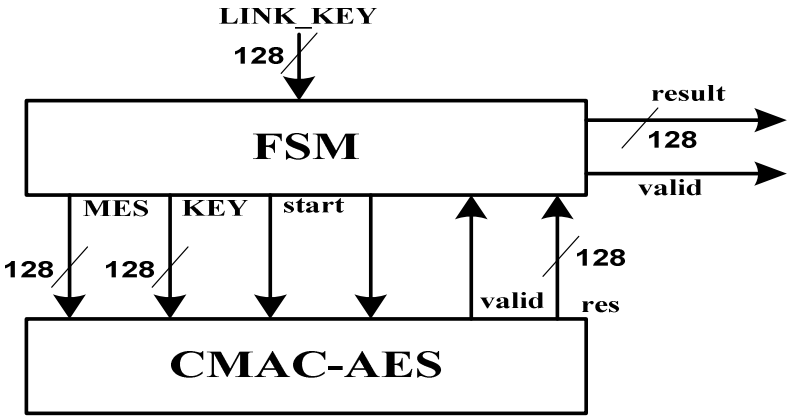
Εικόνα 22: Η μονάδα SHA256

6. Η ΜΟΝΑΔΑ H76

Η μονάδα αυτή υλοποιεί την παραγωγή του LE_KEY ενός κλειδιού που απαιτείται όταν η λειτουργία του Bluetooth είναι σε λειτουργία χαμηλής κατανάλωσης, χρησιμοποιώντας αντίστοιχα πρωτόκολλα επικοινωνίας. Η μονάδα H76 έχει την ίδια λογική με την H2345, δηλαδή χρησιμοποιείται μία FSM η όποια ελέγχει το πότε τελειώνει και πότε αρχίζει η επεξεργασία δεδομένων στον αλγόριθμο CMAC-AES καθώς επίσης και τις τιμές των εισόδων. Οι συναρτήσεις H7 και H6 όπως αναφέρθηκε και στις περιγραφές των συναρτήσεων παραπάνω χρησιμοποιούν τον αλγόριθμο AES σε CMAC μορφή. Η είσοδος σε αυτή την μονάδα είναι μόνο το κλειδί σύνδεσης των 128 bits και τόσο το κλειδί όσο και το block που εισέρχονται στην υπό-μονάδα CMAC-AES συναρμολογούνται στην FSM ανάλογα με το ποια συνάρτηση θα χρησιμοποιηθεί. Μια επιπλέον μεταβλητή η οποία εισέρχεται από την υψηλότερη στην ιεραρχία μονάδα με το όνομα CT2 αναλαμβάνει να επιλέξει αν θα χρησιμοποιηθεί η H6 ή η H7 στο πρώτο επίπεδο υπολογισμού του κλειδιού. Στην εικόνα 23 και 24 φαίνεται η λειτουργία της FSM και το εσωτερικό της μονάδας H76 αντίστοιχα.



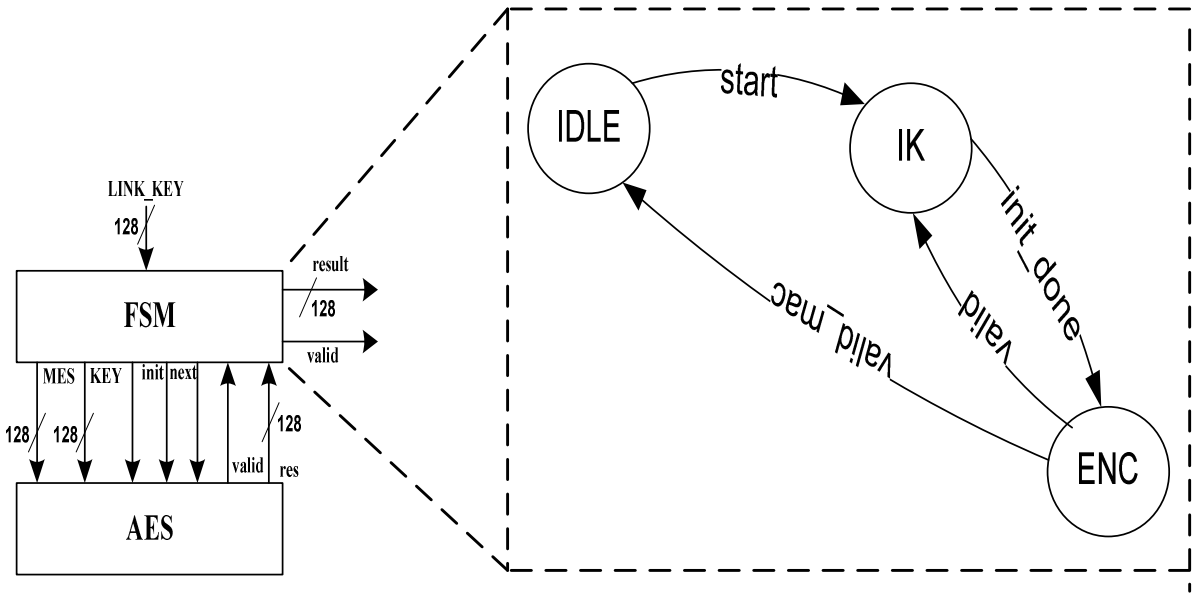
Εικόνα 23: Η λειτουργία της FSM στην H67



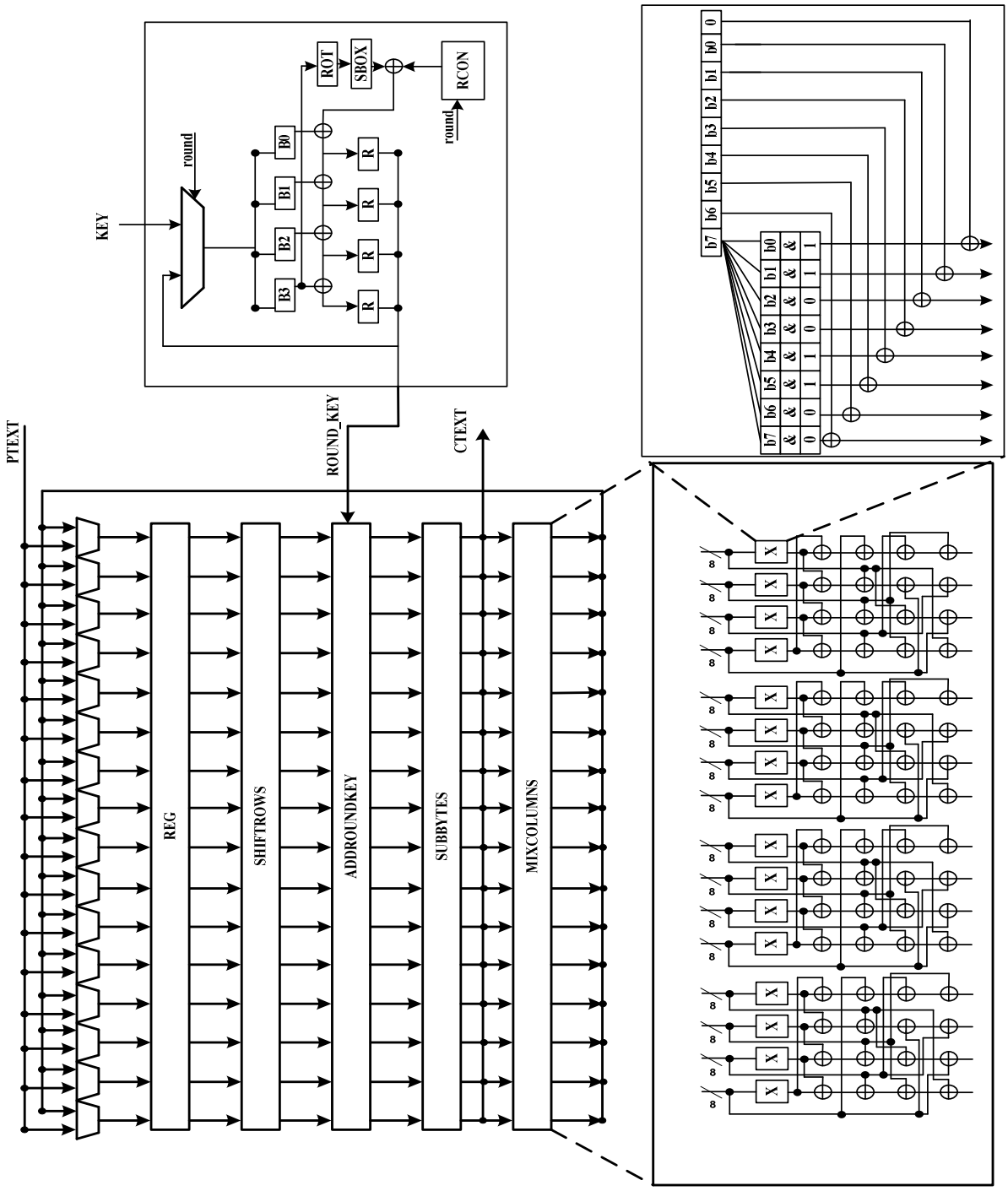
Εικόνα 24: Η μονάδα H67

7. Η ΜΟΝΑΔΑ CMAC-AES

Η μονάδα αυτή αποτελείται από τον αλγόριθμο AES ο οποίος χρησιμοποιείται για συγκεκριμένο αριθμό block καθώς οι συναρτήσεις του Bluetooth H6 και H7 έχουν συγκεκριμένο μήκος εισόδου. Η υλοποίηση στη συγκεκριμένη μονάδα έχει παρόμοια τεχνική με την μονάδα HMAC-SHA256, δηλαδή μία FSM αποφασίζει για το πόσες φορές τα δεδομένα θα επεξεργαστούν στην υπό-μονάδα AES βγάζοντας στην έξοδο το αποτέλεσμα μετά από δύο επεξεργασίες σύμφωνα με τις προδιαγραφές του CMAC που είδαμε στην εικόνα 10. Η αρχιτεκτονική της μονάδας και η περιγραφή της FSM φαίνεται στην εικόνα 25, ενώ η αρχιτεκτονική του επαναλαμβανόμενου γύρου του AES φαίνεται στην εικόνα 26.



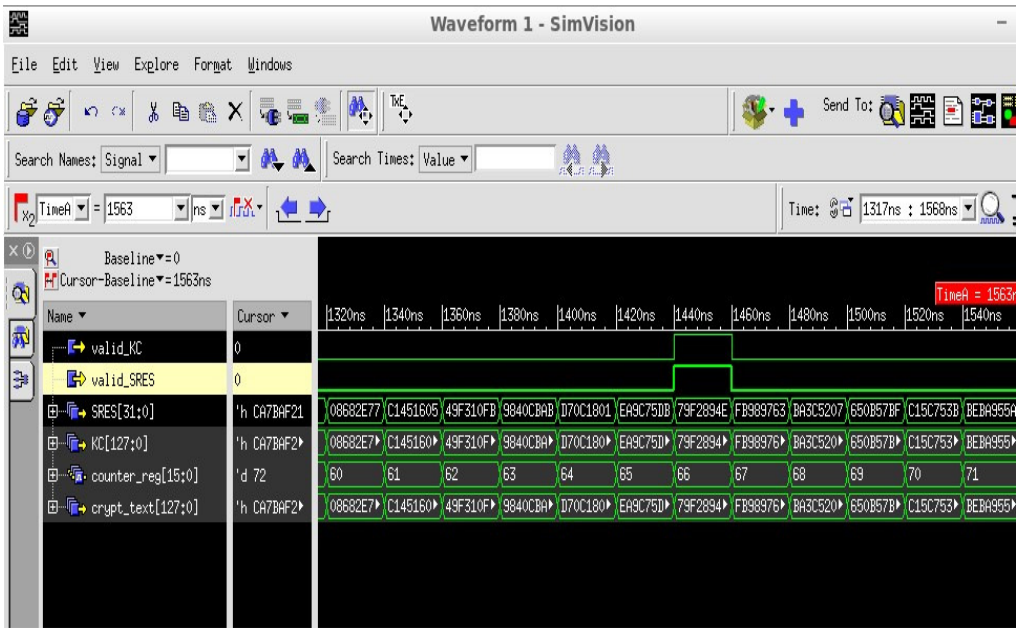
Εικόνα 25: Η μονάδα CMAC-AES



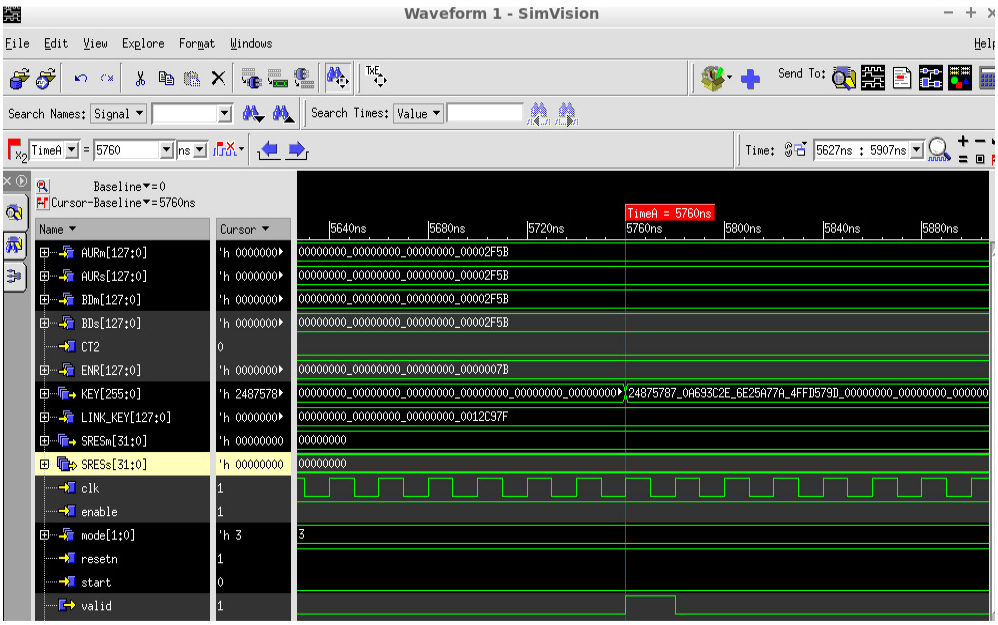
Εικόνα 26: Ο επαναλαμβανόμενος γύρος της μονάδας AES

3. ΑΠΟΤΕΛΕΣΜΑΤΑ ΠΡΟΣΟΜΟΙΟΣΗΣ ΤΗΣ ΜΟΝΑΔΑΣ KEYGEN

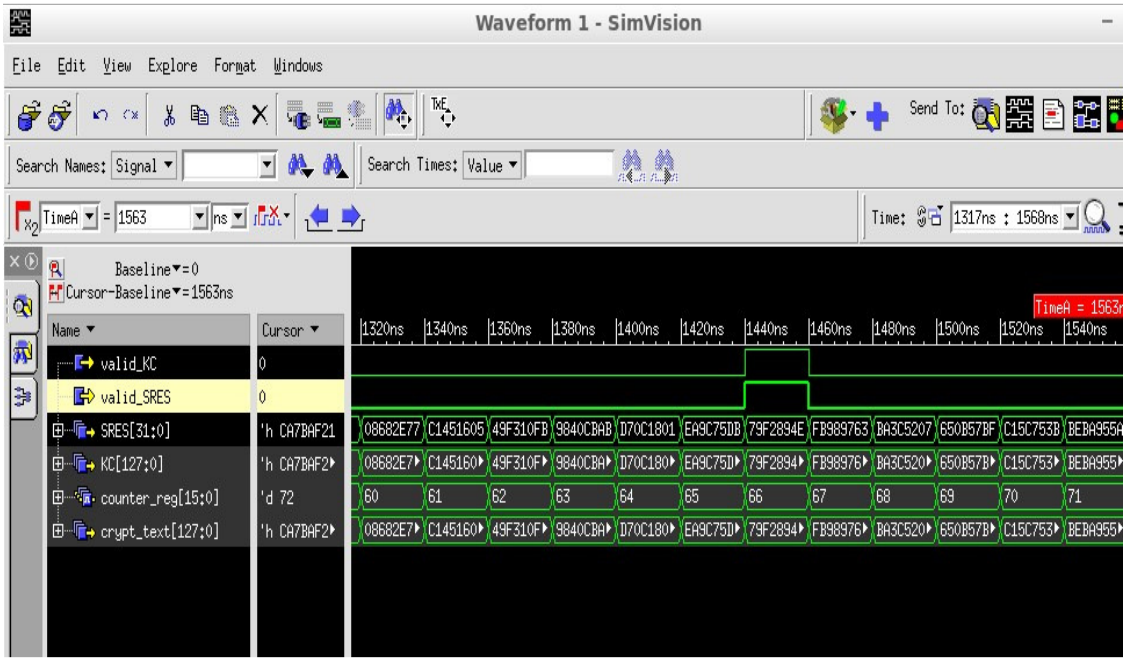
Τόσο ο συνθέσιμος κώδικας όσο και το testbench γράφτηκαν στη γλώσσα περιγραφής υλικού verilog ενώ η προσομοίωση έγινε με το εργαλείο ncverilog. Τα αποτελέσματα που φαίνονται στις εικόνες 27, 28 και 29 είναι από το εργαλείο simvision της Cadence.



Εικόνα 27: Προσομοίωση ασφαλής πιστοποίησης με τη μονάδα H2345



Εικόνα 28: Προσομοίωση παραγωγής κλειδιού LE_KEY



Εικόνα 29: Προσομοίωση παραγωγής κλειδιού για κρυπτογράφηση με stream_cipher.

4. ΑΠΟΤΕΛΕΣΜΑΤΑ ΣΥΝΘΕΣΗΣ

Η σύνθεση έγινε στο εργαλείο Vivado 2014.2 της Xilinx για την πλακέτα Zedboard της σειράς Zynq. Έγιναν δύο υλοποιήσεις, μία με την τεχνική επαναχρησιμοποίησης υλικού που περιγράψαμε παραπάνω και μία χωρίς. Χρησιμοποιήθηκαν όλες οι συναρτήσεις χωρίς καμία βελτιστοποίηση. Στους παρακάτω πίνακες φαίνεται το area που καταναλώνεται στο FPGA ανά μονάδα καθώς και ο αριθμός των κύκλων που απαιτούνται ανά διαδικασία.

	KEYGEN χωρίς την τεχνική επαναχρησιμοποίησης	KEYGEN με την τεχνική επαναχρησιμοποίησης
LUTs	23011	8042
FlipFlops	11240	6462

Πίνακας 1 : Area που καταναλώνεται στο FPGA με και χωρίς την τεχνική επαναχρησιμοποίησης.

	SAFER+	HMAC-SHA256	CMAC-AES
LUTs	1582	2141	2936
FlipFlops	278	2080	2989

Πίνακας 2 : Area που καταναλώνεται ανά κρυπτό-μονάδα.

ΔΙΑΔΙΚΑΣΙΑ	**ΚΥΚΛΟΙ**
ΠΙΣΤΟΠΟΙΗΣΗ ΓΙΑ STREAM CIPHER	34 clock-cycles
ΠΑΡΑΓΩΓΗ ΚΛΕΙΔΙΟΥ ΓΙΑ STREAM CIPHER	66 clock-cycles
ΠΙΣΤΟΠΟΙΗΣΗ ΓΙΑ ΚΡΥΠΤΟΓΡΑΦΗΣΗ ΜΕ AES-CCM	538 clock-cycles
ΠΑΡΑΓΩΓΗ ΚΛΕΙΔΙΟΥ ΓΙΑ ΚΡΥΠΤΟΓΡΑΦΗΣΗ ΜΕ AES-CCM	804 clock-cycles
ΠΑΡΑΓΩΓΗ ΚΛΕΙΔΙΟΥ AMP_KEY	538 clock-cycles
ΠΑΡΑΓΩΓΗ ΚΛΕΙΔΙΟΥ LE_KEY	284 clock-cycles

Πίνακας 3 : Latency που απαιτείται ανά διαδικασία.

5. ΣΥΜΠΕΡΑΣΜΑΤΑ

Στις περισσότερες συσκευές το κομμάτι της ασφάλειας του Bluetooth λαμβάνει χώρα σε λογισμικό, δηλαδή, για να υπολογιστούν τα κλειδιά κρυπτογράφησης αλλά και να επιτευχθεί η κρυπτογράφηση καταναλώνονται πόροι από τον κύριο επεξεργαστή. Ένα καλό παράδειγμα είναι οι συσκευές κινητής τηλεφωνίας. Στην περίπτωση αυτή, ο επεξεργαστής είναι σχεδόν πάντα πολύ ¨απασχολημένος¨ τρέχοντας εφαρμογές υψηλού επιπέδου, λειτουργικό σύστημα και εφαρμογές χαμηλού επιπέδου για την επικοινωνία περιφερικών με το χρήστη. Ένα πλεονέκτημα στην περίπτωση του λογισμικού που αφορά τον κατασκευαστή είναι το γεγονός ότι η υλοποίηση είναι πιο ευέλικτη για τυχόν διορθώσεις και μελλοντικές βελτιστοποιήσεις. Η συγκεκριμένη υλοποίηση σε υλικό μπορεί να αναλάβει το κομμάτι της πιστοποίησης και παραγωγής κλειδιών κρυπτογράφησης αποδεσμεύοντας τον κύριο επεξεργαστή από αυτή τη διαδικασία κάνοντας την σε υλικό.

Η υλοποίηση στη συγκεκριμένη συσκευή FPGA καταφέρνει να αγγίξει τα 160 MHz ταχύτητα ρολογιού πράγμα που καθιστά δυνατή την αντικατάσταση ενός ολοκληρωμένου ή λογισμικού που κάνει την συγκεκριμένη επεξεργασία με ένα FPGA. Η χρήση FPGA θα εξακολουθεί να δίνει την ευελιξία στον κατασκευαστή για τυχόν μελλοντικές αλλαγές καθώς και μεγαλύτερη ταχύτητα επεξεργασίας από υλοποιήσεις σε λογισμικό. Τέλος, η τεχνολογία των FPGAs θα μπορούσε να καταστήσει εφικτή την μεταφορά ολόκληρου του Bluetooth stack σε ένα FPGA αποδεσμεύοντας τον επεξεργαστή από οποιαδήποτε διαδικασία επικοινωνίας μέσω του Bluetooth στο άμεσο μέλλον.

3. ΑΝΑΦΟΡΕΣ

Authentication process in Bluetooth http://www.infotooth.com/knowbase/security/66.htm

Authentication in Bluetooth http://www.infotooth.com/knowbase/security/80.htm

Knowledge Base for Bluetooth information http://www.infotooth.com/

W. A. Arbaugh, "Wireless Research", http://www.cs.umd.edu/~waa/wireless.html

J. Walker, "Unsafe at any key size; An analysis of the WEP encapsulation", http://grouper.IEEE.org/groups/802/11/Documents/DocumentsHolder/0-362.zip Βιβλιογραφία 62 62

N. Borisov, I. Goldberg, and D. Wagner, "Intercepting Mobile Communications: The Insecurity of 802.11." http://wvvw.issac.sc.berkley.edu/issac/wep-faq.html.

W. Arbaugh, N. Shankar, Y.CJ. Wan, "Your 802.11 Wireless Network has No Clothes" http://www.cs.umd.edu/~waa/wireless.pdf

M. Jakobsson and S. Wetzel, "Security Weaknesses in Bluetooth" http://www.rsasecurity.com/rsalabs/staff/bios/mjakobsson/bluetooth/bluetooth.pdf

S.Fluhrer and S.Lucks,"Analysis of the E0 Encryption System" http://th.informatik.uni-mannheim.de/People/Lucks/papers/e0.ps.gz.

"B.Miller,"IEEE 802.11 and Bluetooth wireless technology", http://www -106.ibm.com/developerworks/wireless/library/wi-phone

GeneraI information on bluetooth http://www.mobi1einfo.com/bluetooth/

Annikka Aalto , Bluetooth http://www.tml.hut.f1/Studies/Tikl1 O.300/1999/Essays/bluetooth.htmI

Oraskari, Jyrki, Bluetooth 2000 http://www.hut.f1/~joraskur/bluetooth.html

How Stuff Works, information on BT http://www.howstuffworks.com/bluetooth3.htm