ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ
& ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΕΡΓΑΣΤΗΡΙΟ ΜΙΚΡΟΕΠΕΞΕΡΓΑΣΤΩΝ ΚΑΙ ΥΛΙΚΟΥ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΣΧΕΔΙΑΣΗ ΚΑΙ ΥΛΟΠΟΙΗΣΗ ΜΕ ΔΥΝΑΜΙΚΑ ΑΝΑΔΙΑΤΑΣΣΟΜΕΝΗ ΛΟΓΙΚΗ ΣΥΣΤΗΜΑΤΟΣ ΓΙΑ ΥΛΟΠΟΙΗΣΗ ΠΟΛΛΑΠΛΩΝ ΚΡΥΠΤΟΓΡΑΦΙΚΩΝ ΑΛΓΟΡΙΘΜΩΝ

Νικολουδάκης Γεώργιος

Επιβλέπων : Καθηγητής Απόστολος Δόλλας

Εξεταστική Επιτροπή :

Απόστολος Δόλλας	Καθηγητής Π.Κ.
Πνευματικάτος Διονύσιος	Αναπληρωτής Καθηγητής Π.Κ.
Παπαευσταθίου Ιωάννης	Επίκουρος Καθηγητής Π.Κ

Χανιά, 31 Ιουλίου 2009

Ευχαριστίες

Καταρχήν θα ήθελα να ευχαριστήσω τον καθηγητή μου κ. Απόστολο Δόλλα, για την υποστήριξη και την πολύτιμη συνεισφορά του στην υλοποίηση αυτής της διπλωματικής εργασίας καθώς και στο γεγονός ότι μέσα από την διδασκαλία των μαθημάτων του καθόλη την διάρκεια της φοίτησης μου, μου κίνησε το ενδιαφέρον ώστε τελικά να ασχοληθώ με τον τομέα Μικροεπεξεργαστών και Υλικού του Πολυτεχνείου Κρήτης.

Επίσης, θα ήθελα να ευχαριστήσω τον αναπληρωτή καθηγητή κ. Διονύσιο Πνευματικάτο και τον επίκουρο καθηγητή κ. Ιωάννη Παπαευσταθίου, οι οποίοι δέχτηκαν να αξιολογήσουν την διπλωματική μου διατριβή.

Ευχαριστώ τον Υποψήφιο Διδάκτορα Κυπριανό Παπαδημητρίου για την συνεχή καθοδήγηση καθώς και το αμέριστο ενδιαφέρον που έδειξε για την ολοκλήρωση της διατριβής αυτής.

Στην συνέχεια θα ήθελα να ευχαριστήσω :

Τον κ. Μάρκο Κιμιωνή, υπεύθυνο του εργαστηρίου Μικροεπεξεργαστών και Υλικού, μου παρείχε τόσο τον τεχνολογικό εξοπλισμό όσο και την υποστήριξη του σε διάφορα τεχνικά ζητήματα.

Όλους τους προπτυχιακούς μεταπτυχιακούς και διδακτορικούς φοιτητές του εργαστηρίου Μικροεπεξεργαστών και Υλικού για την βοήθεια τους και την στήριξη τους σε δύσκολα σημεία της διατριβής αυτής.

Ακόμη, όλους τους φίλους μου για όλες τις καλές στιγμές που περάσαμε κατά την διάρκεια των σπουδών μας.

Τέλος και περισσότερο από όλους, ευχαριστώ τους γονείς μου για την βοήθεια και την συμπαράσταση τους καθόλη την διάρκεια των σπουδών μου. Χωρίς την βοήθεια τους και τις προσωπικές θυσίες που έκαναν για μένα, θα ήταν αδύνατη τόσο η ολοκλήρωση αυτής της εργασίας όσο και η επιτυχής περάτωση των σπουδών μου.

Στήν οικογένεια μου

Περίληψη

Την τελευταία δεκαετία η εισβολή των ενσωματωμένων συστημάτων στην καθημερινότητα έχει γίνει ευρέως αντιληπτή. Οι FPGA αποτελούν συνήθως ένα μέρος των συστημάτων αυτών. Τα ενσωματωμένα συστήματα αυτά συνήθως χρησιμοποιούν ανασφαλή δίκτυα για την επικοινωνία τους γεγονός που κάνει σε πολλές περιπτώσεις απαραίτητη την κρυπτογράφηση των δεδομένων.

Στην παρούσα εργασία προχωράμε στην ανάπτυξη ενός ενσωματωμένου συστήματος κρυπτογραφίας με δυνατότητα υποστήριξης πολλαπλών κρυπτογραφικών αλγορίθμων. Η χρήση της Δυναμικής Αναδιάταξης επιτρέπει στο σύστημα μας την εναλλαγή αλγορίθμων κρυπτογραφίας σε πραγματικό χρόνο.

Το σύστημα υλοποιείται σε μία πλατφόρμα XUP χρησιμοποιώντας μια Virtex II Pro FPGA. Περιγράφεται η διαδικασία ανάπτυξης ενός τέτοιου συστήματος, αναλύονται τα πειραματικά αποτελέσματα και παρουσιάζονται τόσο τα οφέλη όσο και οι περιορισμοί από την χρήση της τεχνολογίας αυτής. Τέλος εξετάζεται η δυνατότητα χρήσης της τεχνολογίας αυτής σε συστήματα πραγματικού χρόνου και προτείνονται μέθοδοι πού μπορούν να αυξήσουν την απόδοση της τεχνολογίας κάνοντας πιο εφικτή την χρησιμοποίηση της.

Περιεχόμενα

Συντμήσεις

FPGA	Field Programmable Gate Array
GRM	General Routing Matrix
RPR	Run-Time Partial Reconfiguration
PR	Partial Reconfiguration
PRR	Partial Reconfigurable Module
RECOPS	Reconfigurable Programmable Devices for Military Hardware
SDR	Software Defined Radio
DMA	Direct Memory Acess
ASIC	Application Specific Integrated Circuit
EARP	Early Access Partial Reconfiguration
IOB	Input Output Buffer
DRC	Design Rule Check
NSA	National Security Agency

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

1 Εισαγωγή.

Στο κεφάλαιο αυτό παρουσιάζονται βασικές έννοιες της δυναμικής αναδιάταξης υλικού. Αναλύεται ο στόχος αυτής της διατριβής καθώς και η επιστημονική της συνεισφορά, ενώ στο τέλος παρουσιάζεται και η δομή της διατριβής.

1.1 Βασικές Έννοιες

Στις μέρες μας ολοένα και περισσότερα συστήματα υλοποιούνται με την χρήση FPGA οι οποίες κάνουν εύκολο τον επαναπρογραμματισμό του υλικού. Αυτή η δυνατότητα τους αποτελεί το κύριο πλεονέκτημα τους έναντι ολοκληρωμένων κυκλωμάτων ειδικού σκοπού (ASIC), ενώ σε σχέση με υλοποιήσεις συστημάτων σε επεξεργαστές γενικού σκοπού οι FPGA πλεονεκτούν κυρίως λόγω της αυξημένης απόδοσης τους. Παρόλα αυτά, λόγω του γεγονότος ότι οι σχεδιάσεις που υλοποιούνται σε FPGA είναι στατικές, υστερούν σε ευελιξία και πολλές φορές δεν αποφέρουν τα προσδοκώμενα οφέλη, σε εφαρμογές που είναι σύνθετες ή αυξημένου υπολογιστικού φόρτου.

Η μερική αναδιάταξη έρχεται σαν μια τεχνολογία η οποία κατά πρώτον θα επεκτείνει τα οφέλη από την υλοποίηση συστημάτων σε αναδιατασσόμενο υλικό και κατά δεύτερον θα επεκτείνει την γκάμα των συστημάτων που ωφελούνται από την υλοποίηση τους σε αναδιατασσόμενο υλικό. Η δυναμική αναδιάταξη δίνει την δυνατότητα δημιουργίας ενός εικονικού επίπεδου εφαρμογής, πάνω από την στατική σχεδίαση, το οποίο μπορεί να προσαρμόζει η να αλλάζει την συμπεριφορά του δυναμικά. Αυτό μας δίνει την δυνατότητα να χειριζόμαστε ολόκληρα τμήματα υλικού σαν να ήταν εφαρμογές λογισμικού. Λόγω αυτής της δυνατότητας η χρησιμοποίηση της δυναμικής αναδιάταξης ενδείκνυται σε εφαρμογές κρυπτογραφίας, επεξεργασίας σήματος, τηλεπικοινωνιών, ενώ από την εταιρία Xilinx η οποία πρώτη ξεκίνησε να παρέχει την τεχνολογία αυτή, κυρία εφαρμογή χρήσης δυναμικής αναδιάταξης θεωρείται το SDR.

Η δυναμική αναδιάταξη έχει κεντρίσει τόσο το ακαδημαϊκό ενδιαφέρον όσο και το ενδιαφέρον εταιριών που κατασκευάζουν ηλεκτρονικά κυκλώματα την τελευταία δεκαετία. Προς το παρόν όμως η χρήση της περιορίζεται σε συστήματα μεγάλου κόστους, τα οποία παράγονται σε μικρές ποσότητες και έχουν διάρκεια ζωής μερικές δεκαετίες. Τα συστήματα αυτά χρησιμοποιούν την δυναμική αναδιάταξη για να έχουν την δυνατότητα να ανταποκριθούν σε μελλοντικές απαιτήσεις που δεν είναι δυνατόν να προβλεφθούν κατά την διάρκεια της παραγωγής του συστήματος.

Ήδη σε στρατιωτικές εφαρμογές όπως συστήματα ραντάρ και ασφαλών επικοινωνιών γίνεται χρήση δυναμικής αναδιάταξης ενώ γίνεται όλο και περισσότερο προσπάθεια να ενσωματωθεί η τεχνολογία αυτή σε συσκευές καθημερινής χρήσης.

1.2 Στόχος και Επιστημονική Συνεισφορά

Η παρούσα εργασία μελετά την υλοποίηση συστημάτων τα οποία κάνουν χρήση της μερικής αναδιάταξης που υποστηρίζει το αναδιατασσόμενο υλικό της εταιρίας Xilinx. Μέσω της υλοποίησης αλγορίθμων κρυπτογραφίας γίνεται προσπάθεια παρουσίασης των πλεονεκτημάτων αλλά και μειονεκτημάτων που αποφέρει η χρήση της δυναμικής αναδιάταξης. Οι αλγόριθμοι κρυπτογραφίας ευνοούνται από την υλοποίηση τους σε υλικό λόγο του γεγονότος ότι εκτελούν συνήθως πολύπλοκες λειτουργίες κατά την εκτέλεση τους. Η πολυπλοκότητα τους αυτή μας δίνει την δυνατότητα να παρουσιάσουμε λεπτομερέστερα τα προβλήματα και τους περιορισμούς που εισάγουν τόσο τα σχεδιαστικά εργαλεία που προσφέρονται από την Xilinx, όσο και αυτές καθαυτές οι FPGA. Επισημαίνεται ότι ο σκοπός της εργασίας αυτής δεν επικεντρώνεται στην υλοποίηση συγκεκριμένων αλγορίθμων κρυπτογραφίας, αλλά στην δημιουργία μίας πλατφόρμας που θα έχει την δυνατότητα να ενσωματώσει μελλοντικά και άλλους αλγόριθμους υλοποιημένους σε υλικό, χωρίς να απαιτούνται ιδιαίτερες τροποποιήσεις σε αυτούς.Κύρια στάδια της εργασίας αυτής είναι :

1. Τροποποίηση συγκεκριμένων ήδη δημοσιευμένων αλγορίθμων ώστε να μπορούν να αποτυπωθούν αρχικά σε στατικά συστήματα πάνω στην πλατφόρμα που χρησιμοποιούμε και έπειτα στο Δυναμικά Αναδιατασσόμενο σύστημα μας.

2. Χωροταξικός Σχεδιασμός και ανάπτυξη Δυναμικά αναδιατασσόμενου Συστήματος που να μπορεί να ενσωματώσει τους παραπάνω αλγορίθμους κρυπτογραφίας.

3. Ανάπτυξη συστήματος Επεξεργαστή - Περιφερειακών καθώς και του απαραίτητου λογισμικού για την υποστήριξη του Δυναμικά Αναδιατασσόμενου Συστήματος και της λειτουργίας των κρυπτογραφικών αλγορίθμων.

4. Επαναληπτικός αριθμός βημάτων επιβεβαίωσης λειτουργίας του συστήματος, εξαγωγής πειραματικών αποτελεσμάτων και βελτίωσης του συστήματος βάση των πειραματικών αποτελεσμάτων και των αποτελεσμάτων υλοποίησης.

Η εργασία αυτή μελέτα επίσης το κατά πόσο είναι δυνατή η υλοποίηση δυναμικά αναδιατασσόμενων περιοχών που καταλαμβάνουν μεγάλα τμήματα του διαθέσιμου υλικού καθώς επίσης και τρόπους εναλλαγής σχεδιάσεων που απαιτούν διαφορετικά ρολόγια λειτουργίας η κάθε μία και υλοποιούνται στην ίδια δυναμικά αναδιατασσόμενη περιοχή. Το τελικό αποτέλεσμα είναι ένα πλήρως αυτόνομο αναδιατασσόμενο σύστημα το οποίο μπορεί επαναπρογραμματιστεί μερικώς χωρίς την απαίτηση σύνδεσης με εξωτερικό Υπολογιστή αλλά και χωρίς την διακοπή της λειτουργίας της υπόλοιπης στατικής σχεδίασης.

1.3 Δομή της Διατριβής

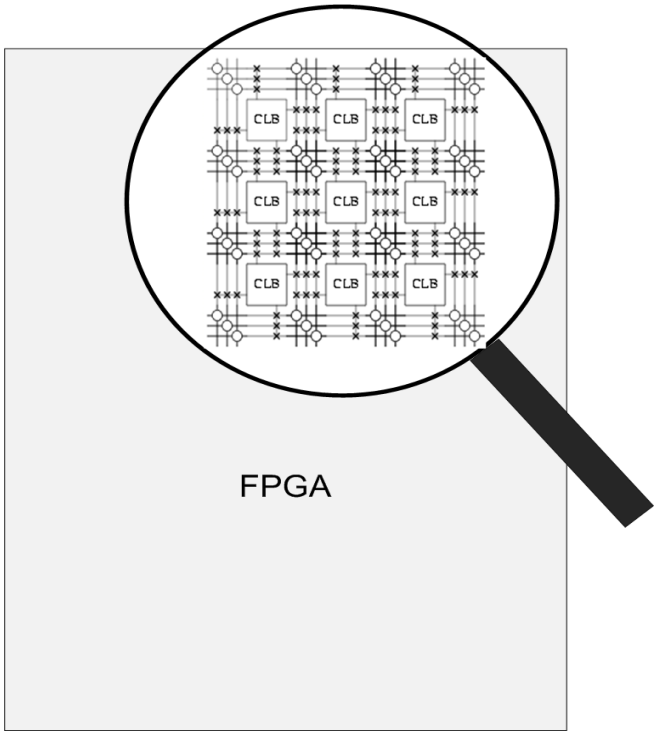
Στο Κεφάλαιο 2 γίνεται μια γενική εισαγωγή στην αναδιατασσόμενη λογική και μια περιγραφή της λειτουργιάς και των ιδιοτήτων των δυναμικά αναδιατασσόμενων συστημάτων. Στο Κεφάλαιο 3 παρουσιάζονται οι αλγόριθμοι κρυπτογραφίας που θα χρησιμοποιηθούν για την αποτίμηση της λειτουργίας της δυναμικής αναδιάταξης. Στο Κεφάλαιο 4 παρουσιάζονται οι σχετικές εργασίες και η επικρατούσα κατάσταση σε υλοποίηση δυναμικά αναδιατασσόμενων συστημάτων τόσο σε συστήματα κρυπτογραφίας όσο και σε άλλες εφαρμογές στις οποίες η χρήση της δυναμικής αναδιάταξης αποφέρει οφέλη. Στο Κεφάλαιο 5 παρουσιάζεται η Αρχιτεκτονική του συστήματος. Το Κεφάλαιο 6 χωρίζεται σε δύο μέρη. Το πρώτο παρουσιάζει τον τρόπο υλοποίησης του συστήματος ενώ το δεύτερο τον τρόπο επιβεβαίωσης της λειτουργίας του. Στο Κεφάλαιο 7 παρουσιάζονται τα πειραματικά αποτελέσματα από την λειτουργία του συστήματος. Κλείνοντας με το Κεφάλαιο 8 γίνεται μια αποτίμηση του όλου συστήματος και προτείνονται μελλοντικές βελτιώσεις.

2 Αναδιατασσόμενη Λογική.

Η Αναδιατασσόμενη Λογική λόγω της ικανότητας της να μεταβάλει σε μικρό χρόνο την λειτουργικότητα που παρέχει ένα ολοκληρωμένο κύκλωμα συνδυάζει την ευελιξία των εφαρμογών λογισμικού με την απόδοση των εφαρμογών που υλοποιούνται σε υλικό, γι'αυτό και χρησιμοποιείται σε μια μεγάλη γκάμα εφαρμογών. Η παρούσα εργασία ασχολείται με FPGAs και συγκεκριμένα FPGAs που έχουν την δυνατότητα να αναδιατάσσονται μερικά. Η μερική αναδιάταξη δίνει την δυνατότητα επαναπρογραμματισμού ενός τμήματος της FPGA ενώ το υπόλοιπο παραμένει σε λειτουργία. Πλέον η αναβάθμιση, ο επαναπρογραμματισμός και ακόμα και η διόρθωση σφαλμάτων των παραπάνω περιοχών είναι εύκολη και χωρίς να υπάρχει χρόνος που το σύστημα μένει εκτός ενεργείας, ιδιότητα ιδιαίτερα χρήσιμη σε αρκετές εφαρμογές. Οι παρακάτω παράγραφοι περιέχουν γενικές έννοιες και πληροφορίες για τα παραπάνω πριν αναφερθούμε σε λεπτομέρειες υλοποίησης στα επόμενα κεφάλαια.

2.1 FPGA

Η FPGA είναι μια συστοιχία ημιαγωγών που περιέχει προγραμματιζόμενη λογική και προγραμματιζόμενες διασυνδέσεις. Τα λογικά τμήματα μπορούν να προγραμματιστούν και να συνδεθούν με τέτοιο τρόπο ώστε να εκτελούν από απλές λειτουργίες βασικών πυλών (AND, OR, XOR) έως πιο πολύπλοκες συναρτήσεις ή μπορούν να σχηματίζουν στοιχεία μνήμης. Η δομή των λογικών μονάδων που περιβάλλονται από τις εσωτερικές διασυνδέσεις παρουσιάζεται στο σχήμα 1. Οι FPGAs κατασκευάσθηκαν αρχικά από την εταιρία Xilinx στα μέσα του 1985. Από τότε πολλές εταιρίες δραστηριοποιούνται στην παραγωγή FPGA. Οι FPGAs υπερτερούν των ASICs σε ευελιξία, δεδομένου ότι τα ASICs δεν έχουν την δυνατότητα να επαναπρογραμματιστούν, έχουν μειωμένο Non Recurring Engineering κόστος, ενώ απαιτείται λιγότερος χρόνος για την ολοκλήρωση της σχεδίασης. Παρόλα αυτά υστερούν σε απόδοση, έχουν μεγαλύτερη κατανάλωση ενέργειας και μεγαλύτερο κόστος ανά μονάδα. Πολλές σχεδιάσεις υλοποιούνται αρχικά σε FPGA ώστε να εξεταστεί η λειτουργία τους πριν παραχθούν σε ASIC. Ωστόσο τα μειονεκτήματα των FPGAs ελαττώνονται όσο αναπτύσσονται καλύτερες τεχνολογίες ημιαγωγών.



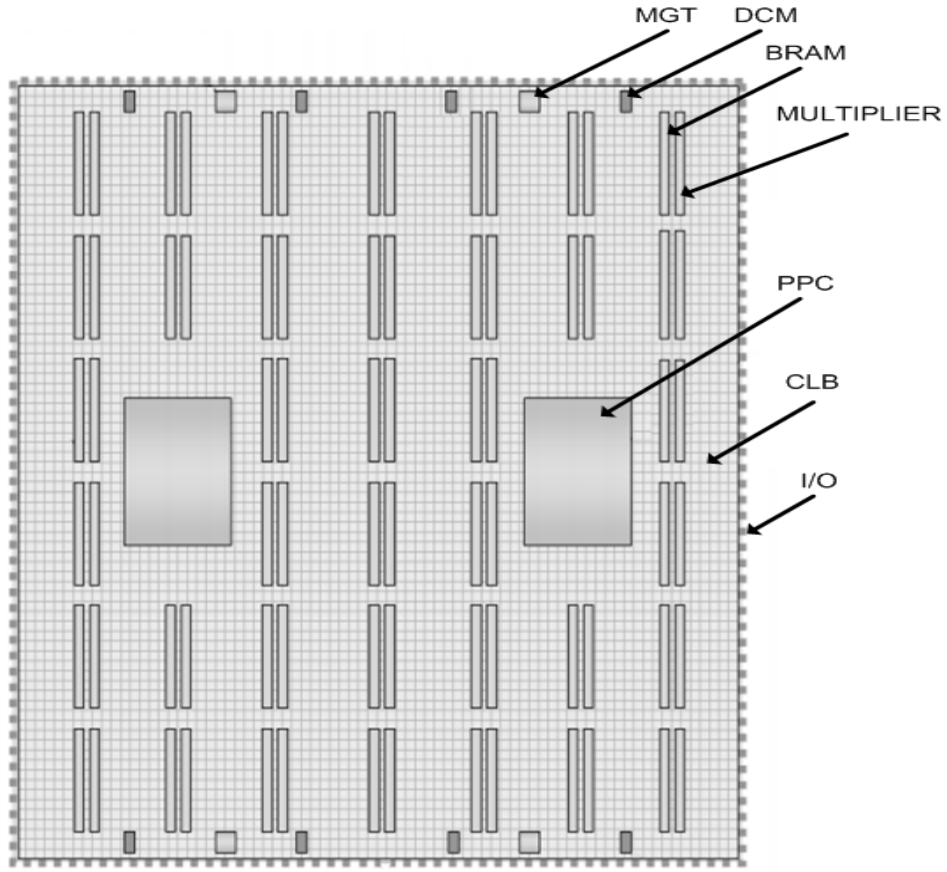
Σχήμα 1: Εσωτερική δομή της FPGA

2.2 Xilinx Virtex-II Pro XC2VP30

Η FPGA με την οποία θα ασχοληθούμε στην εργασία μας είναι η Virtex-II Pro XCV30. Ανήκει στη οικογένεια Virtex των FPGAs της εταιρίας Xilinx η οποία ανακοινώθηκε το 1998. Μέλη της ίδιας οικογένειας αποτελούν και οι Virtex-II, Virtex-E, Virtex-4, Virtex-5, καθώς και οι πρόσφατα ανακοινωμένες Virtex-6. Η συγκεκριμένη FPGA ανήκει στις μεσαίας κλίμακας FPGAs και παρουσιάζει πολύ καλό λόγο κόστους απόδοσης. Έχει χρησιμοποιηθεί εκτενώς τόσο για ακαδημαϊκές μελέτες, όσο και για πραγματικές εφαρμογές. Οπώς φαίνεται και στο σχήμα 2 η Fpga οργανώνεται από κατακόρυφα διανύσματα λογικών βαθμίδων τα επονομαζόμενα CLB τα οποία αποτελούν και το κύριο επαναπρογραμματίσιμο συστατικό των FPGAs. Εκτός από τα CLB η FPGA περιέχει ειδικά κυκλώματα όπως βαθμίδες μνήμης (BRAM), τοποθετημένους πολλαπλασιαστές, βαθμίδες εισόδου εξόδου (IOB), καθώς και διαχειριστές ρολογιού (DCM). Όλα τα παραπάνω συνδέονται μια ιεραρχική γενική βαθμίδα ελέγχου των καλωδιώσεων (GRM) η οποία με την βοήθεια ενσωματωμένων προγραμματίσιμων διακόπτων ελέγχει τις διασυνδέσεις τις κάθε βαθμίδας. Όλες οι παραπάνω βαθμίδες περιγράφονται από το όνομά τους και δύο συντεταγμένες X και Y. Για την αρίθμηση αυτήν θεωρούμε ένα καρτεσιανό επίπεδο οπού ο οριζόντιος άξονας είναι ο X ο κατακόρυφος άξονας είναι ο Y και το σημείο (0,0) αντι-

στοιχεί στην κάτω αριστερή γωνία της FPGA.

Η διαμόρφωση της Fpga εξαρτάται από μια μνήμη τυχαίας προσπέλασης η οποία καλείται μνήμη διαμόρφωσης (Configuration Memory). Η μνήμη αυτή πρέπει να φορτωθεί πριν αρχίσει την λειτουργία της η FPGA. Η μνήμη αυτή χωρίζεται σε κατακόρυφα τμήματα (frames) πλάτους ενός bit και μήκους που κυμαίνεται από 1472 έως 9792 bit. Το frame είναι η μικρότερη μονάδα που μπορούμε να επέμβουμε ώστε να κάνουμε αλλαγές στον προγραμματισμό της Fpga. Ο αριθμός των frame ανα δομικό στοιχείο φαίνεται στο Παράρτημα Α. Η συγκεκριμένη FPGA έχει 1756 frames με μεγέθους 6592 bit το κάθε ένα.



Σχήμα 2: Διάταξη επιμέρους στοιχείων μίας Virtex-II Pro

2.3 Δυναμική Αναδιάταξη

Δυναμική αναδιάταξη είναι η δυνατότητα ενός μέρους της FPGA να επαναπρογραμματίζεται όσο το υπόλοιπο σύστημα παραμένει αμετάβλητο και σε λειτουργία. Η δυναμική αναδιάταξη διαφέρει από την στατική μερική αναδιάταξη στην οποία επαναπρογραμματίζεται μεν ένα μόνο

μέρος της σχεδίασης αλλά το υπόλοιπο παραμένει σε αδράνεια. Η δυνατότητα δημιουργίας ενός εικονικού επιπέδου εφαρμογής πάνω στο υλικό το οποίο μάλιστα θα μπορεί να διαχειρίζεται από λογισμικό κάνει την χρήση της δυναμικής αναδιάταξης πολύ ελκυστική ιδίως σε συστήματα πραγματικού χρόνου. Τα πλεονεκτήματα που μας προσφέρει η τεχνολογία αυτή εστιάζονται στην αύξηση της ταχύτητας στην μείωση της κατανάλωσης και στην παροχή ευελιξίας του υλικού. Πιο συγκεκριμένα τα πλεονεκτήματα είναι :

- Αύξηση της Ταχύτητας: Με την χρήση της δυναμικής αναδιάταξης είναι δυνατή η υποστήριξη περισσοτέρων συναρτήσεων που έχουν την δυνατότητα να αποτυπωθούν στο υλικό κάποια χρονική στιγμή και να αποτελέσουν μέρος του συστήματος. Συνεπώς η υλοποίηση βέλτιστων τμημάτων ανά περίσταση μας οδηγεί σε επιτάχυνση του χρόνου εκτέλεσης.

- Μείωση της κατανάλωσης Ισχύος: Μπορεί να επιτευχθεί μείωσης της στατικής κατανάλωσης ισχύος μειώνοντας τον αριθμό των βαθμίδων που έχουν αποτυπωθεί στο υλικό μια δεδομένη στιγμή και χρησιμοποιώντας στην θέση τους blank bitstreams, ενώ μείωση της δυναμικής κατανάλωσης χρησιμοποιώντας την βέλτιστη σχεδίαση ανά περίσταση.

- Προσαρμοστικότητα: Χάρη στην δυναμική αναδιάταξη τα συστήματα είναι δυνατόν να προσαρμόζονται τόσο στο περιβάλλον που ενεργούν, σε διαφοροποιήσεις των δεδομένων που επεξεργάζονται αλλά και σε μετέπειτα αλλαγές της αποστολής τους.

- Αυτοέλεγχος, Βιωσιμότητα του συστήματος και ανάνηψη από σφάλματα: Η δυναμική αναδιάταξη επιτρέπει την δυναμική αποτύπωση κυκλωμάτων αυτοελέγχου του συστήματος έτσι ώστε να μπορεί να αποτιμηθεί η ορθότητα της λειτουργίας του συστήματος κατά την διάρκεια της εκτέλεσης. Εάν διαπιστωθεί κάποιο σφάλμα τότε είναι δυνατή η αναδιάταξη της εσφαλμένης βαθμίδας ώστε ολόκληρο το σύστημα να συνεχίσει την ορθή λειτουργία του.

- Υλοποίηση Προσαρμοστικών Αλγορίθμων: Η δυναμική αναδιάταξη δίνει την δυνατότητα υλοποίησης αλγορίθμων των οποίων η έκδοση θα εξαρτάται τόσο από τις ανάγκες της διεργασίας όσο και από τα δεδομένα προς επεξεργασία. Η ικανότητα αυτή του συστήματος το κάνει πιο αποδοτικό σε σχέση με την υλοποίηση ενός γενικού αλγορίθμου που δεν θα είναι βέλτιστος ανά περίπτωση. Όλα αυτά μπορούν να γίνουν σε συνδυασμό με την χρήση προσαρμοστικών τεχνικών όπως νευρωνικά δίκτυα για την υλοποίηση δυναμικά βέλτιστων συστημάτων.

Η εταιρία Xilinx υποστηρίζει πλέον την μερική αναδιάταξη για όλες τις FPGAs που παράγει από Virtex 5 έως τις χαμηλού κόστους Spartan.

Υπάρχουν δύο σχεδιαστικές ροές για την υλοποίηση ενός δυναμικά αναδιατασσόμενου συστήματος και για την παραγωγή των partial bitstreams οι οποίες παρουσιάζονται παρακάτω:

2.3.1 Module Based

Η ροή αυτή επιτρέπει την μερική αναδιάταξη περιοχών που έχουν οριστεί εξ'αρχής ως μερικά αναδιατασσόμενες (PPR). Σε κάθε μερικά αναδιατασσόμενη περιοχή μπορούν να αντιστοιχούν πολλές σχεδιάσεις. Για κάθε σχεδίαση θα παραχθεί ένα Partial Bitsteam και επίσης για κάθε περιοχή θα παραχθεί ένα blank bitstream το οποίο στην ουσία αναιρεί οποιοδήποτε προγραμματισμό έχει γίνει μέσα σε αυτήν την περιοχή και την επαναφέρει στην αρχική της κατάσταση. Κατά τον προγραμματισμό ενός Partial Bitsteam προγραμματίζεται όλη η περιοχή εξ αρχής άσχετα εάν η νέα σχεδίαση διαφέρει σε μερικά μόνο σημεία από αυτήν που έχει ήδη αποτυπωθεί στο υλικό. Κατά την παραγωγή του συνολικού bitstream το οποίο θα προγραμματίζει ολόκληρη την FPGA αρχικά, ο σχεδιαστής καλείται να επιλέξει μια προεπιλεγμένη σχεδίαση που θα αποτυπώνεται σε κάθε PRR. Η σχεδιαστική ροή αυτή εισάγει πολλούς περιορισμούς. Οι κυριότεροι από αυτούς είναι :

- Η επικοινωνία των PRR πρέπει να γίνεται με προκαθορισμένους και πλήρως στατικούς διαύλους που ονομάζονται Bus Macros. Τα Bus Macros διασφαλίζουν ότι μετά την εφαρμογή της μερικής αναδιάταξης οι βαθμίδες θα έχουν την δυνατότητα επικοινωνίας με την υπόλοιπη σχεδίαση.

- Το μέγεθος της αναδιατασσόμενης περιοχής εξαρτάται από το μέγεθος της μεγαλύτερης σχεδίασης που θα έχει την δυνατότητα να αποτυπωθεί μέσα σε αυτήν. Συνεπώς όταν αποτυπώνονται σχεδιάσεις μικρότερου μεγέθους υπάρχει δεσμευμένος χώρος που δεν μπορεί να χρησιμοποιηθεί από το υπόλοιπο σύστημα.

- Το ύψος της αναδιατασσόμενης περιοχής καταλαμβάνει ολόκληρη την συσκευή συνεπώς είναι αδύνατη η τοποθέτηση δύο PRR εντός της ίδιας στήλης.

- Η δρομολόγηση των σημάτων ρολογιού και των IOB είναι ξεχωριστή από την αναδιατασσόμενη περιοχή μιας και προγραμματίζονται από ξεχωριστά frames του Partial Bitsream.

2.3.2 Difference Based

Η σχεδιαστική ροή αυτή χρησιμοποιείται μόνο για μικρές αλλαγές στην σχεδίαση που έχει ήδη αποτυπωθεί σε μια FPGA. Για την υλοποίηση της απαιτείται η τροποποίηση της υπάρχουσας σχεδίασης με λογισμικό χαμηλού επιπέδου (FPGA Editor). Στην συνέχεια ακολουθεί παραγωγή ενός partial bitstream το οποίο περιέχει μόνο τις αλλαγές που έχουν γίνει στην υπάρχουσα σχεδίαση. Η ροή αυτή επιτρέπει γρηγορότερο επαναπρογραμματισμό της συσκευής δεδομένου ότι μόνο οι αλλαγές πρέπει να επαναπρογραμματιστούν. Η Difference Based προσέγγιση παρουσιάζει δύο βασικούς περιορισμούς :

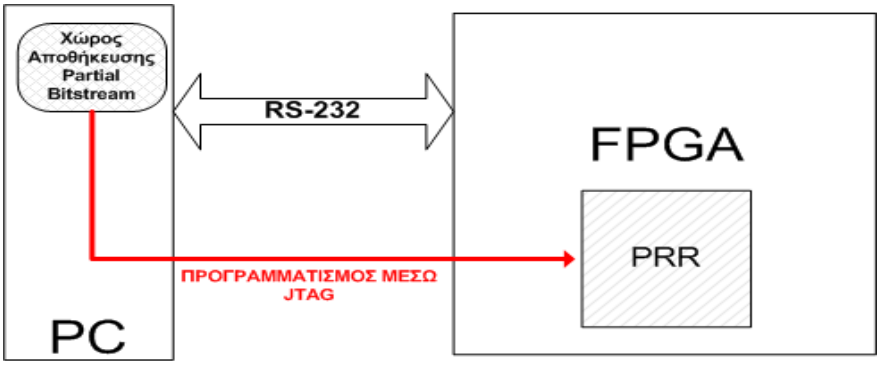
- Δεν υπάρχει δυνατότητα αλλαγής των συνδέσεων (Routing) που έχουν ήδη αποτυπωθεί στην FPGA.
- Εφαρμόζεται σε πολύ περιορισμένο εύρος εφαρμογών δεδομένου ότι οι σχεδιάσεις που εναλλάσσονται πρέπει να είναι πανομοιότυπες μεταξύ τους και σχετικά απλές ώστε να είναι δυνατή η τροποποίηση τους και η μεταλλαγή τους από τον χρήστη.

Στην παρούσα εργασία εφαρμόζεται η πρώτη μέθοδος δεδομένης της ποικιλομορφίας που παρουσιάζουν οι αλγόριθμοι που εξετάζουμε.

2.3.3 Στρατηγικές Υλοποίησης ενός Αναδιατασσόμενου Συστήματος.

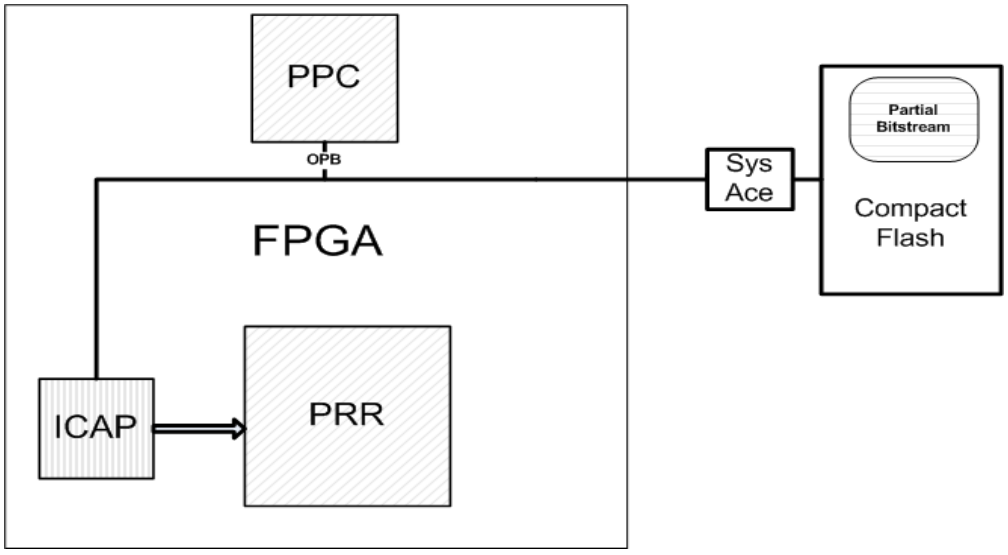
Υπάρχουν διάφοροι τρόποι με τους οποίους μπορεί να υλοποιηθεί ένα δυναμικά αναδιατασσόμενο σύστημα και διαφοροποιούνται στην τοποθεσία που αποθηκεύονται τα partial bitstream και στο ποιό σύστημα παίρνει δυναμικά την απόφαση για μερική αναδιάταξη:

- Εξωτερική Διαμόρφωση: Στην περίπτωση αυτή η διαμόρφωση γίνεται εξωτερικά της FPGA από ένα PC το οποίο είναι συνδεμένο μέσω JTAG. Τα partial bitstream αποθηκεύονται στο εξωτερικό PC το οποίο παίρνει την απόφαση για την αναδιάταξη μιας περιοχής και προγραμματίζει ανάλογα την FPGA. Ο τροπός υλοποίησης ενός τέτοιου συστήματος παρουσιάζεται στο σχήμα 3.



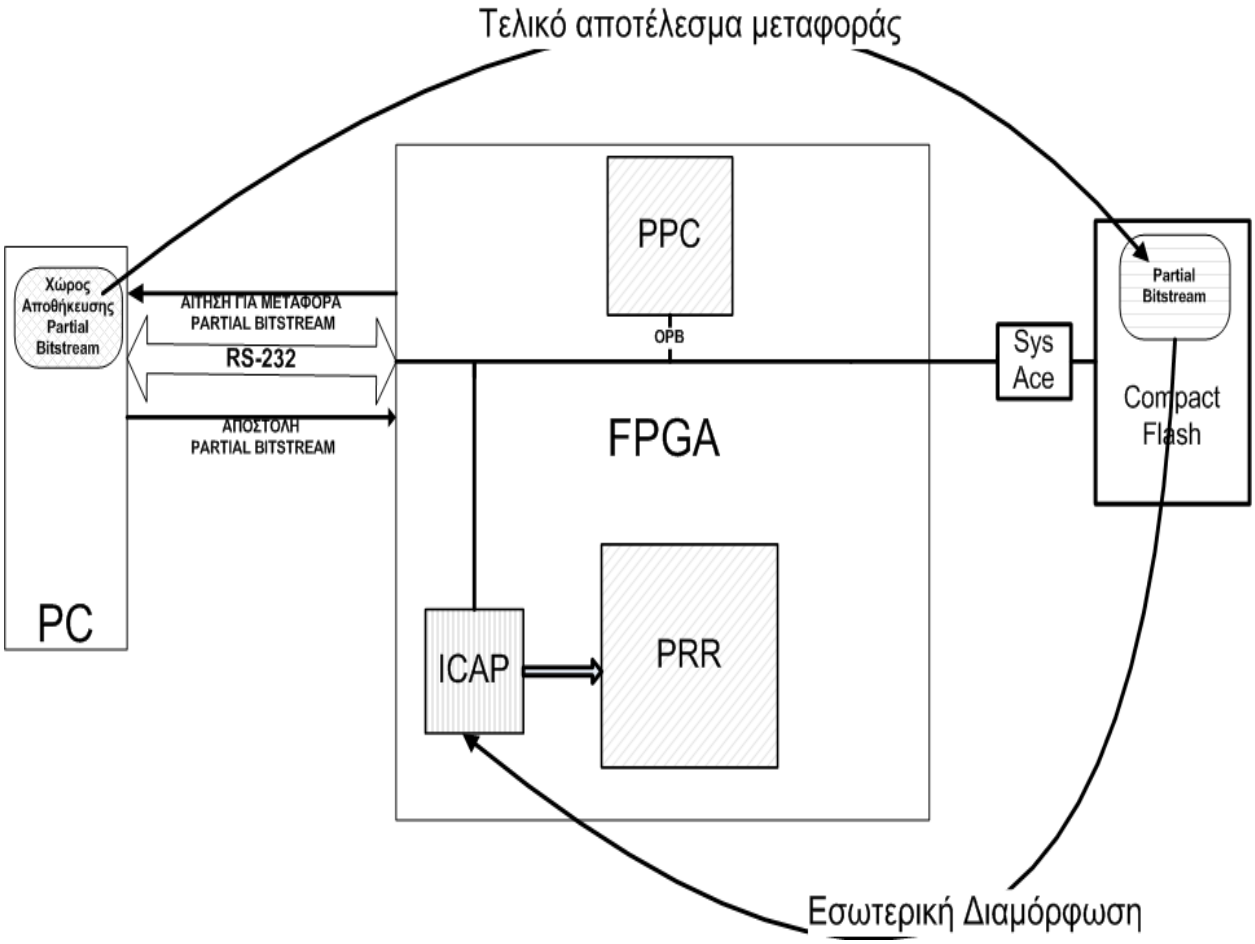
Σχήμα 3: Δυναμική Αναδιάταξη με χρήση εξωτερικού PC

- Εσωτερική Διαμόρφωση: Στην περίπτωση αυτή η διαμόρφωση γίνεται εσωτερικά του συστήματος και το σύστημα είναι πλήρως αυτόνομο και χωρίς την ανάγκη ύπαρξης εξωτερικού PC. Τα partial bitstream αποθηκεύονται είτε σε μια Compact Flash είτε σε εξωτερική μνήμη και με την βοήθεια ενός εσωτερικού επεξεργαστή (PowerPC) μεταφέρονται στον ICAP [49] ο οποίος αναλαμβάνει τον προγραμματισμό των PPR. Ο ICAP ακολουθώντας τον μηχανισμό ανάγνωση-επεξεργασία-εγγραφή εκτελεί τον επαναπρογραμματισμό των PRR τροποποιώντας την Configuration Memory της FPGA. Ο ρόλος του ενσωματωμένου επεξεργαστή είναι συν τις άλλοις η μεταφορά των partial bitstreams από την εξωτερική του μνήμη προς τον ICAP και η λήψη της απόφασης για το πότε πρέπει να γίνει δυναμική αναδιάταξη. Η δομή ενός τέτοιου συστήματος παρουσιάζεται στο σχήμα 4.



Σχήμα 4: Αυτόνομο Δυναμικά Αναδιατασσόμενο Σύστημα

- Υβριδικό Σύστημα: Η διαμόρφωση αυτή συνδυάζει τις δύο προηγούμενες μεθόδους και ουσιαστικά αναφέρεται σε ένα σύστημα το οποίο τοπικά έχει αποθηκευμένες κάποιες σχεδιάσεις για τα PRR και έχει επίσης την δυνατότητα αναβάθμισης της βιβλιοθήκης από τα partial bitstreams τα οποία αποθηκεύονται εντός του συστήματος. Συνεπώς αν απαιτηθεί κάποια σχεδίαση της οποίας το partial bitstream δεν υπάρχει στο χώρο αποθήκευσης των partial bitstreams, τότε το σύστημα αιτείται σε ένα εξωτερικό εξυπηρετητή για μεταφορά του partial bitstream τοπικά και στην συνέχεια εκτελεί κανονικά εσωτερική διαμόρφωση. Ο τρόπος υλοποιήσης ενός τέτοιου συστήματος παρουσιάζεται στο σχήμα 5.



Σχήμα 5: Υβριδική Υλοποίηση ενός Δυναμικά Αναδιατασσόμενου Συστήματος

3 Κρυπτογραφικοί Αλγόριθμοι.

Για να μελετήσουμε τη δυναμική αναδιάταξη σε βάθος θα έπρεπε να μελετήσουμε την επίδραση της εφαρμογής της πάνω σε συγκεκριμένες εφαρμογές. Η ψηφιακή εποχή στην οποία ζούμε κάνει απαραίτητη την διασφάλιση της προστασίας των δεδομένων από κακόβουλους χρήστες τόσο κατά την μετάδοση όσο και την αποθήκευση των δεδομένων. Η διακίνηση των δεδομένων μέσα από ανασφαλή δίκτυα κάνει ακόμα πιο επιτακτική την ανάγκη για κρυπτογράφηση δεδομένων. Για τις ανάγκες της εργασίας επιλέχθηκαν τρείς από τους πιο διαδεδομένους αλγόριθμους συμμετρικής κρυπτογραφίας ο DES, ο Triple-DES και ο AES με σκοπό να μελετήσουμε την επίδραση της εφαρμογής της μερικής αναδιάταξης στην υλοποίηση ενός συστήματος που υποστηρίζει και του τρείς παραπάνω αλγόριθμους. Οι αλγόριθμοι αυτοί επιλέχθηκαν για δύο βασικούς λόγους:

1. Και οι τρείς παραπάνω αλγόριθμοι είναι ευρέως χρησιμοποιούμενοι όπου απαιτείται κρυπτογράφηση δεδομένων.

2. Η υλοποίηση τους σε μία ενιαία πλατφόρμα κρυπτογραφίας θα μας βοηθήσει να εξετάσουμε την συμπεριφορά της μερικής αναδιάταξης σε

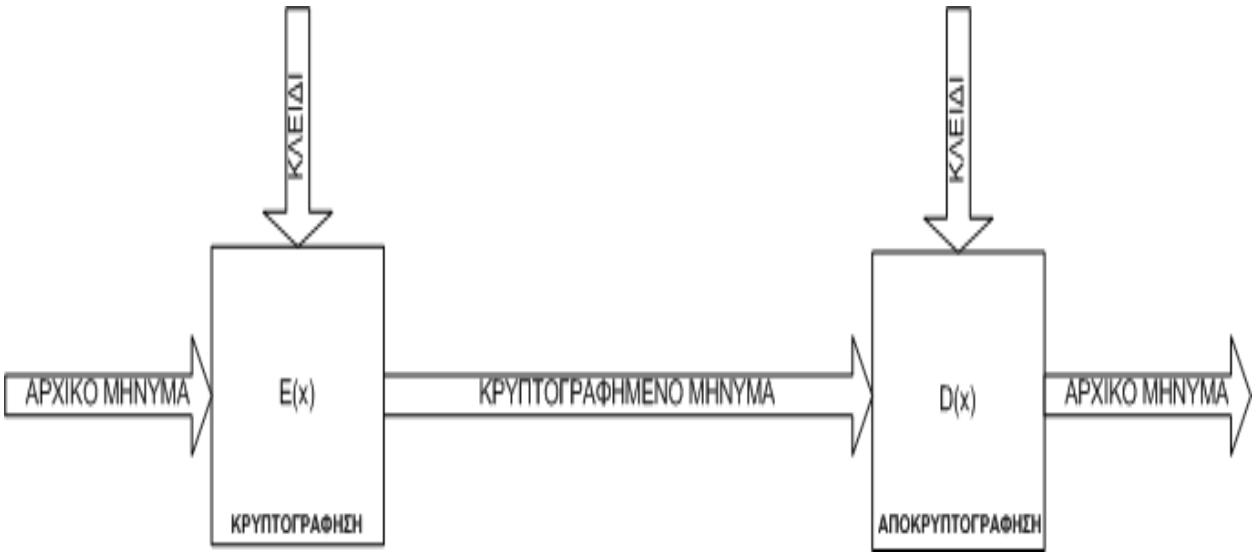
   i Υλοποίηση αλγορίθμων ιδιαίτερου υπολογιστικού φόρτου και πολυπλοκότητας.

   ii Δυναμική εναλλαγή αλγορίθμων που διαφέρουν αρκετά στις απαιτήσεις πόρων που απαιτείται για να υλοποιηθούν όσο και στην συχνότητα ρολογιού που απαιτείται για να λειτουργήσουν.

   iii Αλγορίθμους και σχεδιάσεις που η ταυτόχρονη λειτουργία τους σε μια στατική σχεδίαση θα ήταν αδύνατη λόγω μη ύπαρξης των απαιτούμενων διαθέσιμων πόρων.

3.1 Συμμετρική Κρυπτογραφία

Οι αλγόριθμοι κρυπτογραφίας χωρίζονται σε δύο μεγάλες κατηγορίες· τους αλγόριθμους συμμετρικού κλειδιού και τους αλγόριθμους ασύμμετρου κλειδιού. Οι αλγόριθμοι που ασχολείται η παρούσα εργασία είναι συμμετρικού κλειδιού. Η λειτουργία των αλγορίθμων αυτών παρουσιάζεται στο σχήμα 6.



Σχήμα 6: Αλγόριθμοι Κρυπτογραφίας Συμμετρικού Κλειδιού

Η ασφάλεια των αλγορίθμων συμμετρικού κλειδιού εξαρτάται μόνο από το κλειδί το οποίο είναι γνωστό μόνο στον αποστολέα και στους αποδέκτες των κρυπτογραφημένων μηνυμάτων. Για τον λόγο αυτό οι αλγόριθμοι συμμετρικού κλειδιού αναφέρονται και σαν αλγόριθμοι ιδιωτικού κλειδιού.

Οι συμμετρικοί αλγόριθμοι αν και διαφέρουν μεταξύ τους στον τρόπο που επιδρούν στα δεδομένα χαρακτηρίζονται από τις εξής παραμέτρους:

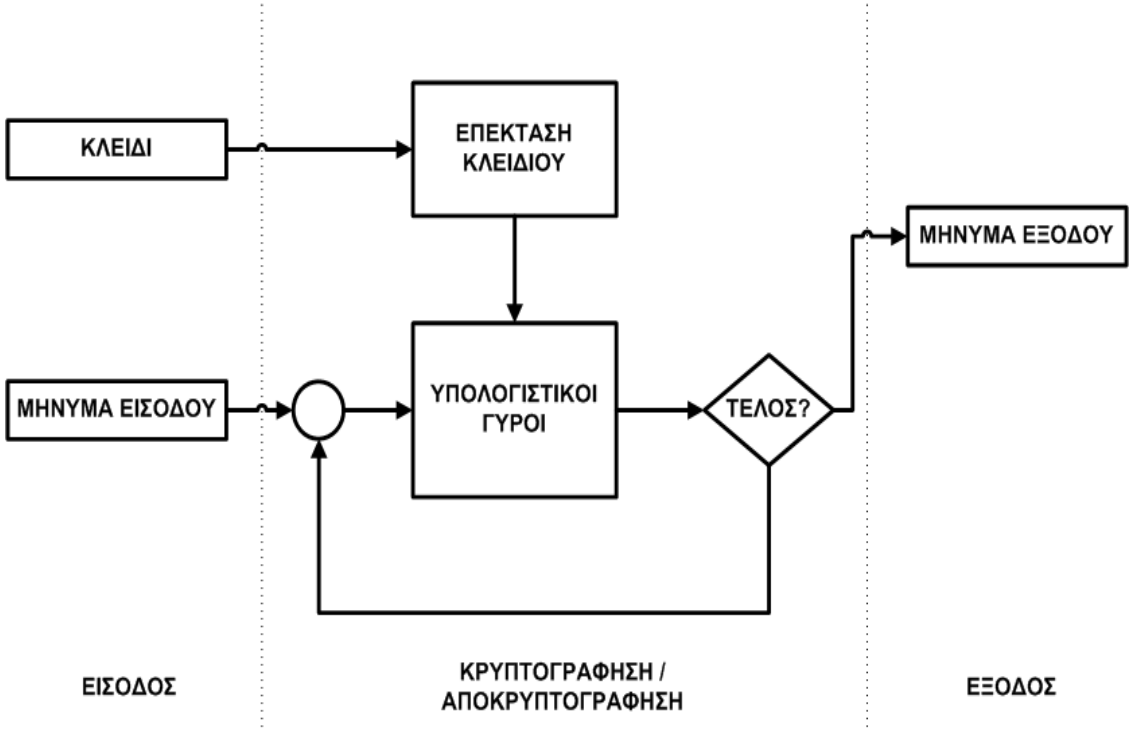
1. Το μέγεθος σε bit του μυστικού κλειδιού που χρησιμοποιείται κατά την διάρκεια της κρυπτογράφησης και της αποκρυπτογράφησης. Όσο μεγαλύτερο το μέγεθος του κλειδιού τόσο πιο ασφαλής και ανθεκτικός σε επιθέσης ωμής βίας (brute force) θεωρείται ο αλγόριθμος. Ταυτόχρονα όμως αύξηση του μεγέθους του κλειδιού συνεπάγεται και αυξήση των απαιτούμενων πόρων για την λειτουργία του.

2. Το μέγεθος σε bit του μπλοκ δεδομένων που κρυπτογραφείται και αποκρυπτογραφείται κάθε φορά. Μεγάλα μπλόκ δεδομένων εχούν αυξήμενες απαιτήσεις σε μνήμη.

3. Τον αριθμό των επαναληπτικών υπολογιστικών κύκλων που πραγματοποιούνται κατά την διάρκεια της κρυπτογράφησης και της αποκρυπτογράφησης. Ο αριθμός των επαναληπτικών κύκλων είναι σημαντικός για το επίπεδο ασφαλείας που προσφέρει ο αλγόριθμος. Όσο περισσότεροι υπολογιστικοί κύκλοι, τόσο λιγότερη συσχέτιση υπάρχει μεταξύ του αρχικού και του κρυπτογραφημένου μυνήματος αλλα και τόσο περιορίζεται η ταχύτητα

επεξεργασίας δεδομένων του αλγορίθμου.

Ο τρόπος με τον οποίο λειτουργούν παρουσιάζεται στο σχήμα 7. Βάση του κλειδιού παράγονται νέα υπο-κλειδιά τα οποία χρησιμοποιούνται σε κάθε υπολογιστικό γύρο του αλγορίθμου. Τα δεδομένα προς κρυπτογράφηση - αποκρυπτογράφηση υπόκεινται σε επαναλαμβανόμενη επεξεργασία η οποία καλείται γύρος. Όταν ο προκαθορισμένος αριθμός των γύρων επιτευχτεί η διαδικασία έχει ολοκληρωθεί. Υπάρχουν διάφοροι τρόποι με τους οποίους μπορεί να επιδράσει ένας αλγόριθμος κρυπτογραφίας πάνω σε ένα μήνυμα. Οι πιο διαδεδομένοι είναι ο ECB (Electronic Code Book) όπου κάθε μπλοκ κρυπτογραφείται ξεχωριστά, αλλά και μέθοδοι αλυσίδας όπως CBC (Chipher Block Chaining), CFB (Chipher Feedback), OFB (Output FeedBack) οπού κάθε μπλοκ συνδυάζεται είτε με προγενέστερα block είτε με προγενέστερες εξόδους. Ο ECB τρόπος προσφέρει μεγαλύτερη ταχύτητα, ενώ οι υπόλοιποι τρόποι μεγαλύτερη ασφάλεια και προστασία από επιθέσεις επανάληψης [8],[9], [10].

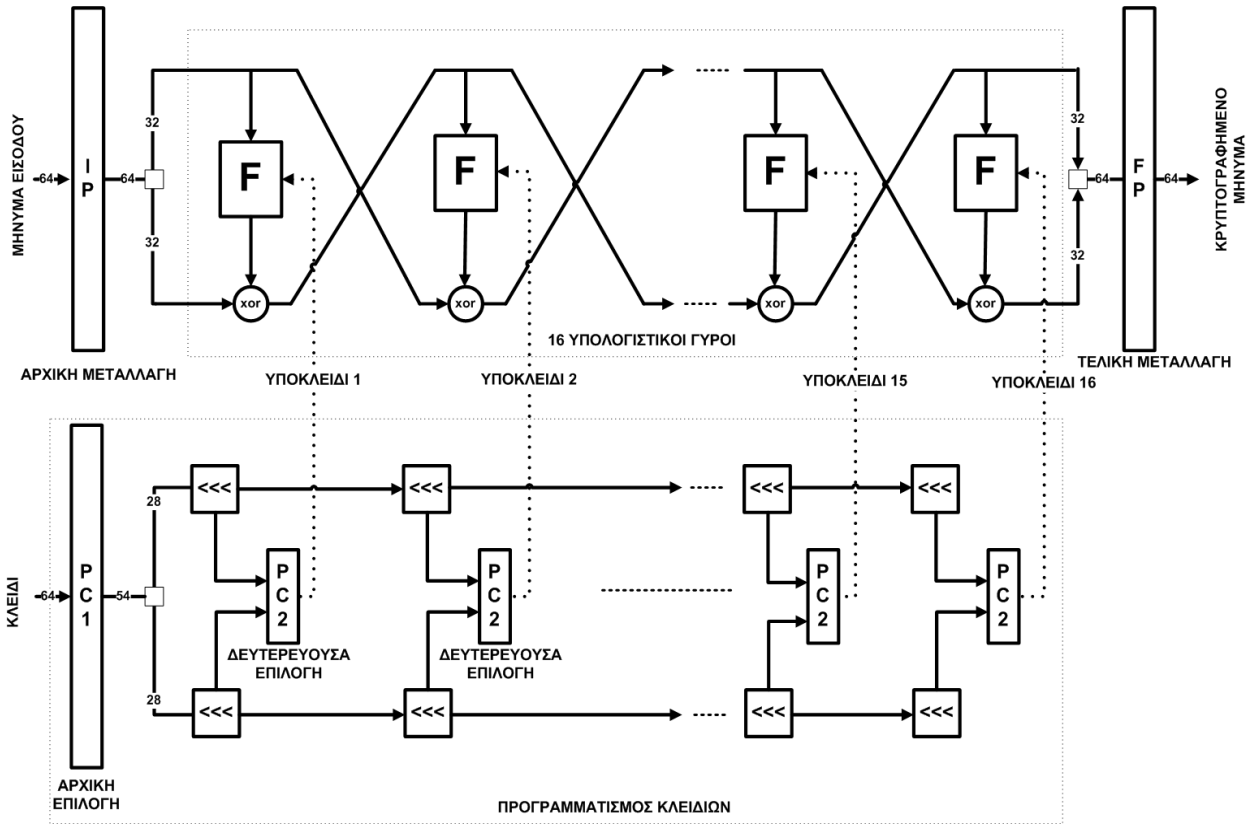


Σχήμα 7: Διάγραμμα Λειτουργίας Συμμετρικών Αλγορίθμων Κρυπτογραφίας

3.2 Data Encryption Standard

Ο DES είναι ένας συμμετρικού κλειδιού αλγόριθμος κρυπτογραφίας μπλοκ δεδομένων. Αναπτύχθηκε τα έτη 1974 − 1975 από την εταιρία IBM βασισμένος στον προϋπάρχοντα αλγόριθμο DTD-1 (Lucifer), για κάλυψη απαίτησης ενός προτύπου κρυπτογραφίας για ευαίσθητες μη διαβαθμισμένες πληροφορίες της Αμερικάνικης κυβέρνησης. Καθιερώθηκε ως επίσημο Ομοσπονδιακό Πρότυπο Επεξεργασίας Πληροφοριών (FIPS) το 1976 από το σώμα προτύπων των Η.Π.Α και επιβεβαιώθηκε σαν πρότυπο άλλες τέσσερις φορές έως ότου το Μάιο του 2002 αντικαταστάθηκε από τον πιο προηγμένο AES. Παρόλο που έχει συμπληρώσει τριάντα χρόνια ζωής και η ασφάλεια που προσφέρει είναι πλέον αμφισβητούμενη ο DES χρησιμοποιείται ευρέως. Ο DES κρυπτογραφεί μπλόκ δεδομένων 64 bit την φορά με την χρήση κλειδιού μήκους 64 bit. Από τα 64 bit του κλειδιού μόνο τα 56 χρησιμοποιούνται για την κρυπτογράφηση ενώ τα υπόλοιπα για έλεγχο ισοτιμίας. Ο αλγόριθμος κρυπτογραφεί τα δεδομένα βάση μιας επαναλαμβανόμενης διαδικασίας η οποία καλείται γύρος. Σε κάθε γύρο χρησιμοποιείται ένα υποκλειδί το οποίο έχει παραχθεί από το αρχικό κλειδί με βάση μια διαδικασία η οποία καλείται προγραμματισμός κλειδιών. Πριν και μετά του υπολογιστικούς γύρους τα δεδομένα υπόκεινται σε δύο συναρτήσεις μεταλλαγής, που είναι αντίστροφες μεταξύ τους και αλληλοαναιρούνται, οι οποίες δεν έχουν καμία κρυπτογραφική αξία απλώς εισήχθησαν στον αλγόριθμο για να επιβραδύνουν την εκτέλεση του σε λογισμικό.

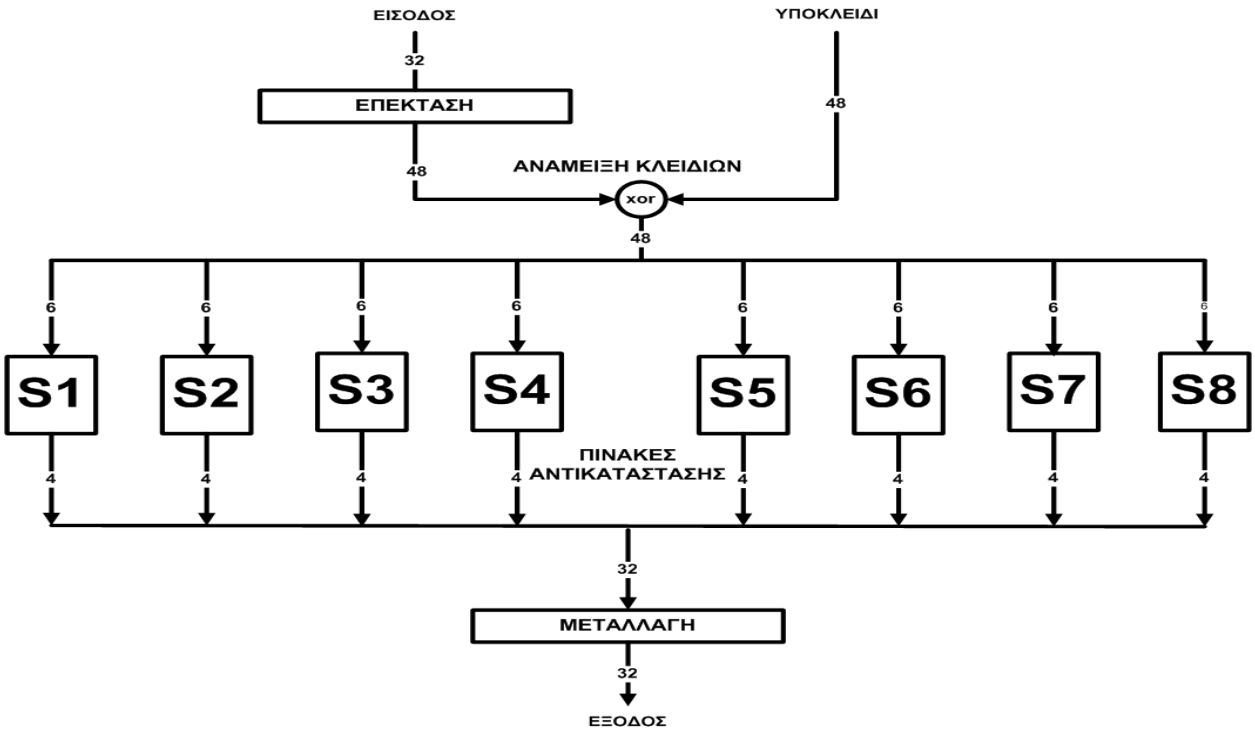
Στο σχήμα 8 και στο σχήμα 9 παρουσιάζεται η λειτουργία του αλγορίθμου .Τα δεδομένα εισόδου αρχικά υφίστανται μια αρχική μετάλλαξη και στην συνέχεια χωρίζονται σε δύο 32 bit ποσότητες. Σε κάθε γύρο μία από τις δύο 32 bit ποσότητες μετατρέπεται σύμφωνα μίας συνάρτησης F που περιγράφεται παρακάτω. Στο τέλος του γύρου γίνεται μια πράξη xor μεταξύ της μίας 32 bit ποσότητας εισόδου και της 32 bit ποσότητας εξόδου της συνάρτησης F. Οι δύο 32 bit ποσότητες επεξεργάζονται εναλλάξ σε κάθε γύρο και η διαδικασία αυτή επαναλαμβάνεται για δεκαέξι γύρους. Μετά την ολοκλήρωση των υπολογιστικών γύρων τα δεδομένα υφίστανται μια τελική μετάλλαξη και το κρυπτογραφημένο μήνυμα είναι πλέον διαθέσιμο. Για την παραγωγή των απαιτούμενων υποκλειδιών που χρησιμοποιούνται σε κάθε γύρο του αλγορίθμου ακολουθείται μια συγκεκριμένη διαδικασία η οποία ονομάζεται προγραμματισμός κλειδιών. Κατά τον προγραμματισμό κλειδιών αρχικά γίνεται επιλογή των 56 bit που θα χρησιμοποιηθούν στον αλγόριθμο. Στην συνέχεια το κλειδί διασπάται σε δύο 28 bit ποσότητες οι οποίες ολισθαίνουν κατά μία ή δύο θέσεις ανάλογα τον γύρο που βρίσκεται ο αλγόριθμος, ενώ μια δευτερεύουσα



Σχήμα 8: Γενική Περιγραφή Λειτουργίας του DES.

επιλογή επιλέγει τα 48 bit που τελικά θα χρησιμοποιηθούν, σαν υποκλειδί του εκάστοτε γύρου, από την συνάρτηση F. Η συνάρτηση F επιδρά πάνω σε μια 32 bit ποσότητα και αρχικά την επεκτείνει σε 48 bit διπλασιάζοντας κάποια bit της αρχικής ποσότητας. Στην συνέχεια εκτελείται μια πράξη xor μεταξύ της επεκταμένης ποσότητας και του υποκλειδίου. Το αποτέλεσμα της xor χωρίζεται σε οκτώ ομάδες μεγέθους 6 bit που καθεμία διευθυνσιοδοτεί ένα πίνακα αντικατάστασης (S-Boxes) του οποίου η έξοδος είναι 4 bit. Τελικά η 32 bit έξοδος των πινάκων αντικατάστασης αναδιατάσσεται σύμφωνα με ένα πίνακα μεταλλαγής.

Η διαδικασία αποκρυπτογράφησης είναι η ίδια με την διαφορά ότι τα υποκλειδία εφαρμόζονται με την αντίστροφη σειρά.



Σχήμα 9: Περιγραφή της συνάρτησης F του αλγόριθμου DES.

3.3 Triple - Advanced Encryption Standard

Δεδομένου ότι ο DES θεωρείτο ανασφαλής μια παραλλαγή του προτάθηκε σαν λύση για την υλοποίηση ενός πιο ισχυρού και πιο ασφαλή αλγορίθμου κρυπτογραφίας. Ο αλγόριθμος αυτός ονομάζεται Triple DES ή αλλιώς TDEA (Triple Des Encryption Algorithm) και έχει γίνει ήδη αποδεκτός για κρυπτογράφηση ευαίσθητων πληροφοριών έως το έτος 2030. Ο αλγόριθμος αποτελείται από τρεις διαδοχικούς DES αλγορίθμους. Η λειτουργία του φαίνεται στο σχήμα 10. Συνήθως ο ενδιάμεσος αλγόριθμος εκτελεί αποκρυπτογράφηση ενώ οι άλλοι δύο εκτελούν κρυπτογράφηση. Ένα μήνυμα κρυπτογραφημένο με τον Triple DES θα περιγράφεται ως εξής

$$T = E_{K3}(D_{K2}(E_{K1}(M))) \tag{1}$$

όπου M θα είναι το αρχικό μήνυμα, T θα είναι το τελικό κρυπτογραφημένο μήνυμα, E η διαδικασία κρυπτογράφησης D η διαδικασία αποκρυπτογράφησης και K τα κλειδιά [16][13]. Στην περιγραφή του αλγορίθμου υπάρχουν αρκέτες επιλογές για την επιλογή των κλειδιών οι οποίες φαίνονται στον πίνακα 1 [17][18].

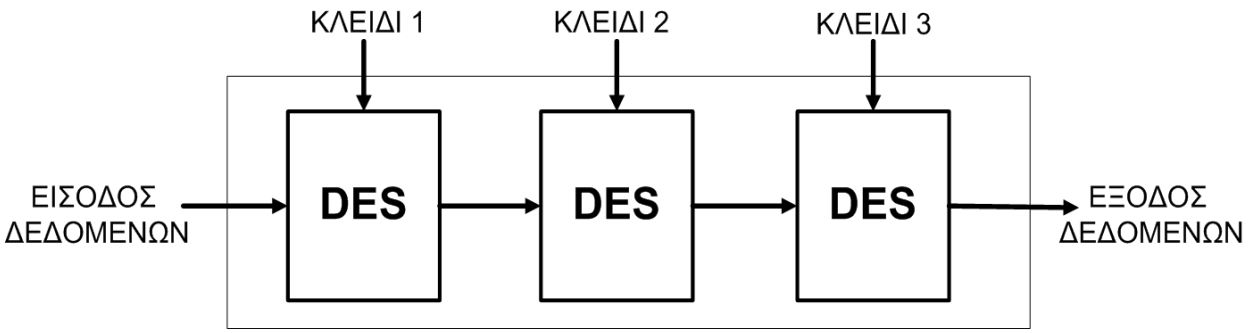
Α/Α	ΕΠΙΛΟΓΗ	ΜΗΚΟΣ ΚΛΕΙΔΙΟΥ (bit)
1	$K1 \neq K2 \neq K3$	168
2	$(K1 = K3) \neq K2$	112
3	$K1 = K2 = K3$	56

Πίνακας 1: Δυνατές επιλογές Κλειδιών για τον Αλγόριθμο Triple DES.

Είναι αξιοσημείωτο το γεγονός ότι αν επιλέξουμε όλα τα κλειδιά του αλγορίθμου να είναι όμοια (επιλογή 3 τού πίνακα 1) τότε ο αλγόριθμος μας είναι ακριβώς όμοιος με ένα αλγόριθμο DES με κλειδί Κ δεδομένου ότι τα δύο πρώτα στάδια αναιρούνται μεταξύ τους όπως φαίνεται και στο σχήμα 10. Η αποκρυπτογράφηση ακολουθεί την αντίστροφη διαδικασία δηλαδή

$$M = D_{K3}(E_{K2}(D_{K1}(T))) \tag{2}$$

όπου Μ θα είναι το αρχικό μήνυμα, Τ θα είναι το τελικό κρυπτογραφημένο μήνυμα, Ε η διαδικασία κρυπτογράφησης D η διαδικασία αποκρυπτογράφησης και Κ τα κλειδιά [13] [16].



Σχήμα 10: Περιγραφή της Λειτουργίας του Αλγόριθμου Triple Data Encryption Algorithm.

3.4 Advanced Encryption Standard

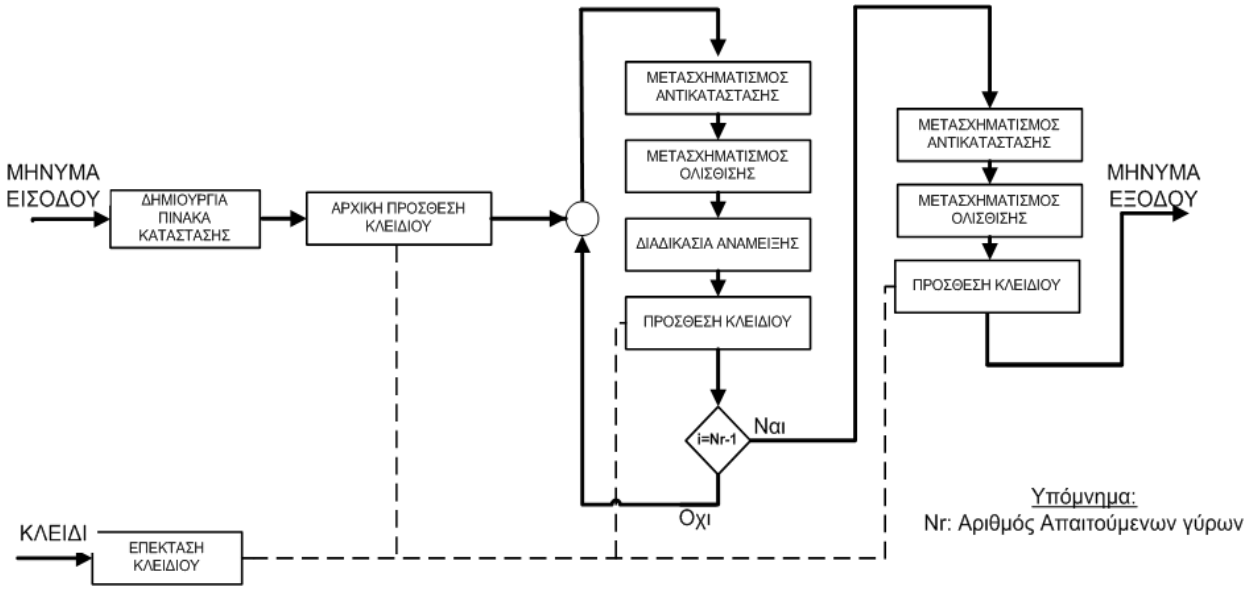
Ο AES είναι ένας συμμετρικού κλειδιού αλγόριθμος κρυπτογραφίας μπλοκ δεδομένων. Επιλέχθηκε από το Διεθνή Οργανισμό Προτύπων και Τεχνολογίας (NIST) ως ο αντικαταστάτης του DES το 2001. Από τους δεκαπέντε υποψήφιους αλγόριθμους για το πρότυπο AES τελικά επιλέχθηκε ο αλγόριθμος Rijndael λόγω της υψηλής ασφάλειας που προσφέρει και της σχετικά εύκολης υλοποίησης του τόσο σε υλικό όσο και λογισμικό. Είναι γεγονός ότι ο AES είναι ο πρώτος ανοιχτός αλγόριθμος στο κοινό ο οποίος πιστοποιήθηκε από την Υπηρεσία Εθνικής Ασφάλειας των Η.Π.Α. (NSA) για χρήση σε άκρως απόρρητες πληροφορίες [20].

Ο αλγόριθμος κρυπτογραφεί μπλοκ δεδομένων μεγέθους 64 bit την φορά. Συμφώνα με το πρότυπο δύναται η χρησιμοποίηση κλειδιών μήκους 128, 192 και 256 bit. Ο αλγόριθμος κρυπτογραφεί τα δεδομένα βάση μιας επαναληπτικής διαδικασίας η οποία καλείται γύρος. Ο αλγόριθμος χωρίζεται σε τρία βασικά στάδια· στον αρχικό γύρο που έχει σαν είσοδο τα δεδομένα και το αρχικό κλειδί, στους ενδιαμέσους γύρους που έχουν σαν είσοδο την έξοδο του προηγούμενου γύρου και υποκλειδιά που έχουν παραχθεί βάση του αρχικού κλειδιού και στον τελικό γύρο η έξοδος του οποίου είναι το κρυπτογραφημένο μήνυμα. Ο αριθμός των γύρων ποικίλει ανάλογα με το μήκος του κλειδιού και φαίνεται στον πίνακα 2 [20].

Μήκος κλειδιού	Αριθμός Γύρων
128	10
192	12
256	14

Πίνακας 2: Δυνατές επιλογές Μήκους Κλειδιού για τον Αλγόριθμο AES.

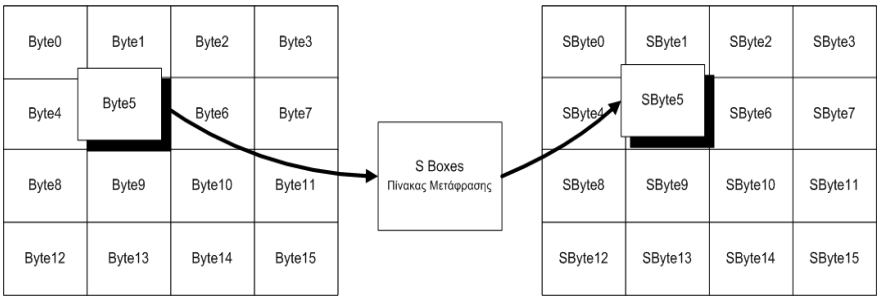
Η λειτουργία του αλγορίθμου φαίνεται στο σχήμα 11. Τα δεδομένα εισόδου αναδιατάσσονται σε ένα πινάκα 4 X 4 bytes ο οποίος καλείται κατάσταση. Όλη η λειτουργία του αλγορίθμου περιγράφεται σαν η επεξεργασία που γίνεται σε αυτόν τον πίνακα κατάστασης.



Σχήμα 11: Περιγραφή της Λειτουργίας Κρυπτογράφησης του Αλγόριθμου AES.

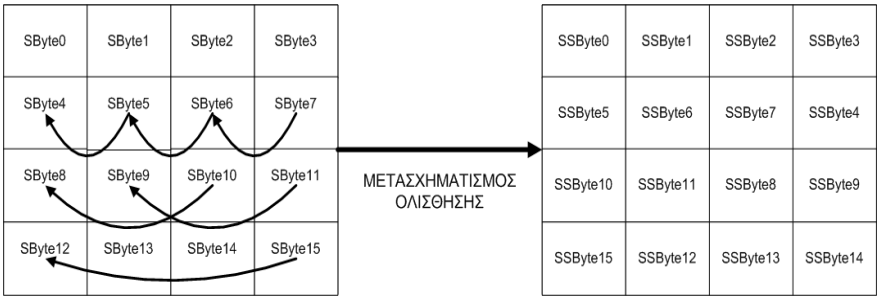
Κάθε γύρος αποτελείται από

- Ένα Μετασχηματισμό αντικατάστασης οπού κάθε byte του πίνακα κατάστασης αντικαθιστάται σύμφωνα με ένα πίνακα μετάφρασης όπως φαίνεται στο σχήμα 12 .

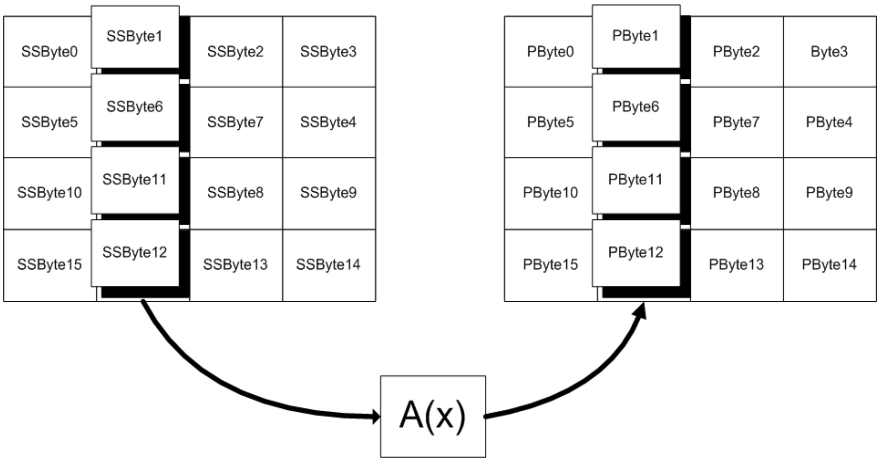


Σχήμα 12: Μετασχηματισμός Αντικατάστασης Byte.

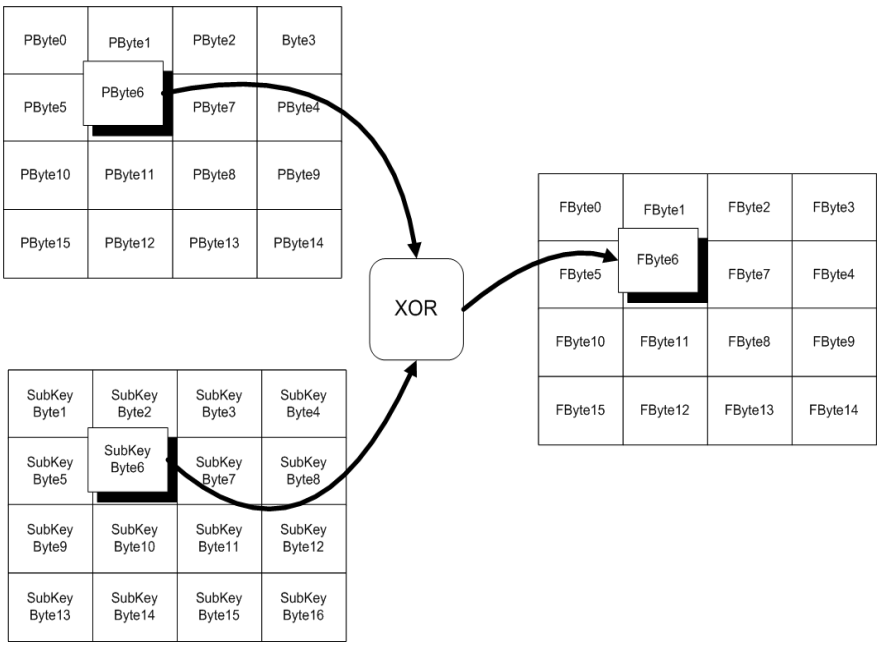
- Ένα μετασχηματισμό ολίσθησης όπου κάθε byte του πίνακα κατάστασης ολισθαίνει κυκλικά κατά αριθμό ανάλογο της σειράς που βρίσκεται όπως φαίνεται στο σχήμα 13 .
- Μια διαδικασία ανάμειξης οπού κάθε στήλη του πίνακα κατάστασης πολλαπλασιάζεται με ένα σταθερό πολυώνυμο όπως φαίνεται στο σχήμα 14.
- Μία διαδικασία πρόσθεσης κλειδιού όπου πραγματοποιείται μια πράξη xor μεταξύ καθενός byte του πίνακα κατάστασης με ένα byte του υποκλειδιού του αντίστοιχου γύρου όπως φαίνεται στο σχήμα 15 .



Σχήμα 13: Μετασχηματισμός Ολίσθησης Byte.

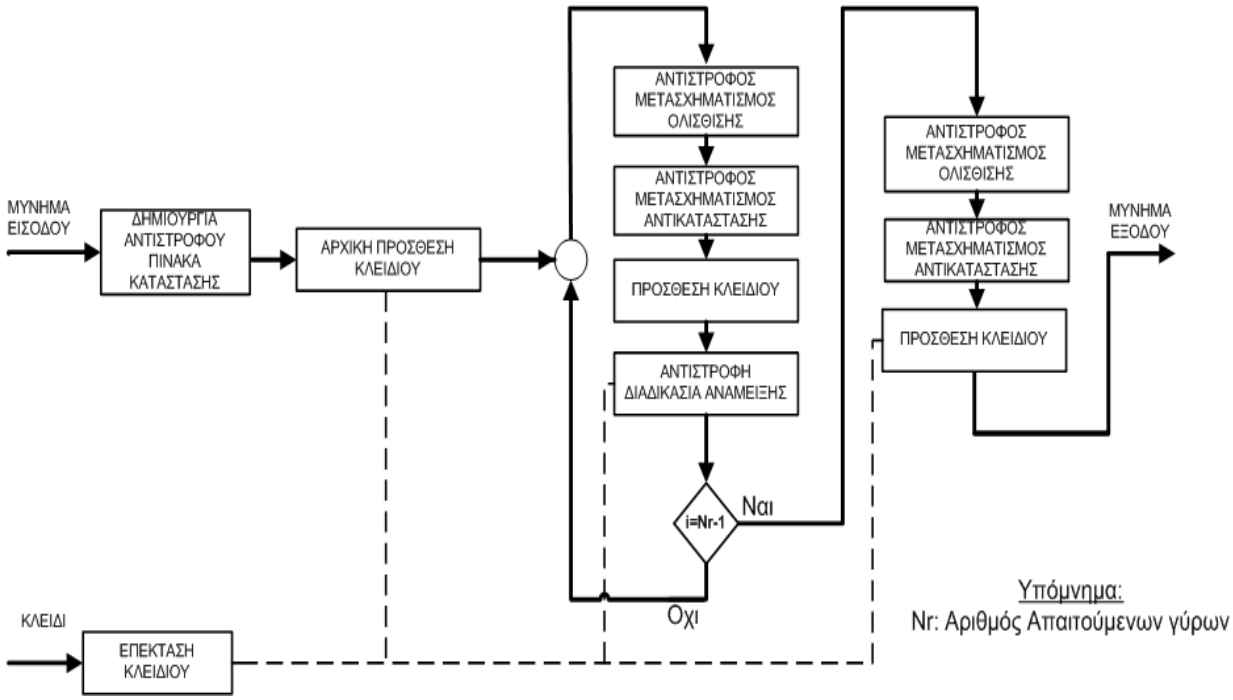


Σχήμα 14: Μετασχηματισμός Ανάμειξης Byte.



Σχήμα 15: Διαδικασία Πρόσθεσης Κλειδιού.

Ο πρώτος γύρος αποτελείται μόνο από την διαδικασία πρόσθεσης κλειδιού ενώ ο τελευταίος γύρος δεν περιλαμβάνει την διαδικασία ανάμειξης. Τα υποκλειδιά που απαιτούνται για κάθε γύρο του αλγορίθμου παράγονται από την διαδικασία επέκτασης κλειδιού. Η διαδικασία της αποκρυπτογράφησης είναι ακριβώς ίδια μόνο που οι διαδικασίες κάθε γύρου είναι τροποποιημένες ώστε να κάνουν την αντίστροφη διαδικασία όπως φαίνεται στο σχήμα 16 [8][9][10].



Σχήμα 16: Περιγραφή της Λειτουργίας Αποκρυπτογράφησης του Αλγόριθμου AES.

4 Σχετική Έρευνα.

Το κεφάλαιο αυτό παρουσιάζει την έρευνα που έχει γίνει και είναι σχετική με το θέμα αυτής της διατριβής. Το κεφάλαιο χωρίζεται σε δύο μέρη. Στο πρώτο μέρος περιγράφονται εργασίες που αφορούν γενικά την μερική αναδιάταξη υλικού, ενώ το δεύτερο μέρος επικεντρώνεται σε εργασίες σχετικά με συγκεκριμένους αλγόριθμους κρυπτογραφίας που έχουν υλοποιηθεί είτε σε δυναμικά είτε σε στατικά αναδιατασσόμενο υλικό.

4.1 Δυναμική Αναδιάταξη Υλικού

Ο Ανυφαντής [24] προχώρησε σε μια πειραματική ανάλυση των καθυστερήσεων που δημιουργούνται κατά την εφαρμογή της δυναμικής αναδιάταξης σε μια Virtex- II Pro FPGA. Ακολουθώντας Different Based ροή κατάφερε να δώσει πραγματικά αποτελέσματα για τους χρόνους που χρειάζονται για αναδιάταξη της μικρότερης δυνατής βαθμίδας της FPGA δηλαδή των frames. Με την εργασία του επαλήθευσε και επέκτεινε τα θεωρητικά στοιχειά που δίνονται από τον κατασκευαστή για την τεχνολογία αυτή.
Ο Παπαδημητρίου [29] πρότεινε και εν συνέχεια προχώρησε στην αποτίμηση ενός μοντέλου για προφόρτωση (prefetching) διάφορων αναδιατασσόμενων περιοχών που δεν χρησιμοποιούνται μια δεδομένη στιγμή, με σχεδιάσεις που επίκειται να χρησιμοποιηθούν στο άμεσο μέλλον. Με τον τρόπο αυτό κατάφερε να επικαλύψει χρόνο που χρειάζεται για δυναμική αναδιάταξη με αποτέλεσμα την μείωση του συνολικού χρόνου εκτέλεσης της διαδικασίας από 6% έως 86%. Στο [30] προχώρησε σε διάσπαση του συνολικού χρόνου αναδιάταξης σε επιμέρους τμήματα και παρουσίασε ποσοτικά αποτελέσματα για το κάθε τμήμα ξεχωριστά για μία Virtex-II Pro FPGA.
Ο Ευφραιμίδης [21] προχώρησε στην υλοποίηση ενός αυτόνομου μερικά αναδιατασσόμενου συστήματος γενετικού αλγορίθμου το οποίο υποστηρίζει την αλλαγή συνάρτησης προσαρμογής (Fitness Function)σε χρόνο εκτέλεσης με την χρήση της δυναμικής αναδιάταξης. Με τον τρόπο αυτό κατάφερε να υλοποιήσει ένα συστήμα γενετικού αλγοριθμού που έχει δυνατότητα υλοποίησης θεωρητικά άπειρων συναρτήσεων προσαρμογής αλλά και την δυνατότητα να αλλάζει την συνάρτηση προσαρμογής σε πραγματικό χρόνο χωρίς να επηρεάζεται το υπόλοιπο σύστημα.
Ο Sedcole [32] σε συνεργασία με μηχανικούς της Xilinx αφού προχώρησε σε εκτενή μελέτη της ροής Module Based της κατασκευάστριας εταιρίας Xilinx, πρότεινε μια νέα σχεδιαστική ροή όπου η αλλαγή των σχεδιάσεων σε μια δυναμικά αναδιατασσόμενη περιοχή (PRR) θα γίνεται

μόνο στα frames που απαιτείται. Θα γίνεται δηλαδή ανάγνωση της υπάρχουσας σχεδίασης, εύρεση των frames που περιέχουν αλλαγές και εγγραφή μόνο των frames που περιέχουν αλλαγές στην μνήμη διαμόρφωσης. Με την ροή αυτή κατάφερε να πετύχει αύξηση του χρόνου αναδιάταξης κατά περίπου 50%.

Ο Claus [31] προχώρησε στην υλοποίηση ενός ελεγκτή PLB-ICAP και χρησιμοποιώντας τεχνικές άμεσης πρόσβασης μνήμης (DMA) κατάφερε να πετύχει ρυθμό διαμεταγωγής 20 φορές πιο γρήγορο από εκείνο που παρέχεται από την Xilinx .

Ο Custodio [23] προχώρησε στην υλοποίηση δυναμικά αναδιατασσόμενου συστήματος το οποίο έχει την δυνατότητα εντοπισμού λαθών και επιδιόρθωσης σε πραγματικό χρόνο ενώ το υπόλοιπο σύστημα παραμένει ανεπηρέαστο.

Παρόλο που την τελευταία δεκαετία η χρήση της δυναμικής αναδιάταξης υλικού μελετήθηκε ευρέως σε ερευνητικό επίπεδο ακόμα και σήμερα υπάρχουν πολύ λίγες πραγματικές εφαρμογές που την χρησιμοποιούν.

Το Recops [27] είναι έως σήμερα το μεγαλύτερο εγχείρημα για την μελέτη της δυναμικής αναδιάταξης. Αναπτύχθηκε από την Ευρωπαϊκή Επιτροπή Άμυνας (EDA) και χρηματοδοτήθηκε επίσης από το Γαλλικό, το Ιταλικό και το Βελγικό Υπουργείο Αμύνας. Πάνω από δέκα διαφορετικές εταιρίες και πανεπιστήμια συμμετέχουν στην ολοκλήρωση του προγράμματος. Τα πειράματα αφορούν μια ευρεία γκάμα εφαρμογών όπως:

- Ψηφιακή επεξεργασία και μετάδοση εικόνας.
- Υλοποίηση μόντεμ για επικοινωνίες δεδομένων.
- Software Define Radio (Ολοκληρωμένο σύστημα Πομπού Δέκτη).
- Συσκευές Ηλεκτρονικού Πολέμου.

και είχαν ως κύριο σκοπό να καλύψουν το χάσμα μεταξύ της ακαδημαϊκής έρευνας και της βιομηχανικής αντίληψης περί την χρήση της δυναμικής αναδιάταξης. Στα πρώτα τους δημοσιευμένα αποτελέσματα [28] συγκρίνουν τα θεωρητικά πλεονεκτήματα της τεχνολογίας με τα αποτελέσματα των πειραματικών μετρήσεων τους. Χαρακτηρίζουν την τεχνολογίας ως πολλά υποσχόμενη αλλά επισημαίνουν ότι ναι μεν τα οφέλη από την εξοικονόμηση χώρου και την εικονοποίηση του υλικού είναι δεδομένα, αλλά θέματα όπως η μείωση της καταναλώσεως ισχύος και η αύξηση της ταχύτητας εξαρτώνται όχι μόνο από την τεχνολογία αλλά και από την εφαρμογή. Αξιοσημείωτο

είναι το γεγονός ότι κατάφεραν να κατασκευάσουν ενα ολοκληρώμενο σύστημα για πρόσβαση και επεξεργασία της μνήμης διαμόρφωσης μιας Virtex-II Pro FPGA , το οποίο εκτέλει ακριβώς τις ίδιες διαδικασίες με το OPB-ICAP που παρέχει η Xilinx, αλλα είναι 84 φορές πιο γρήγορο.

4.2 Υλοποιήση Αλγορίθμων Κρυπτογραφίας σε Δυναμικά Αναδιατασσόμενο Υλικό

Ο Lagger [22] προχώρησε στην ανάπτυξη μιας δυναμικά αναδιατασσόμενης πλατφόρμας κρυπτογραφίας. Οι αλγόριθμοι που υποστήριζε ήταν ο DES, ο Triple DES και ο RC4. Προχώρησε στην χρησιμοποίηση λειτουργικών συστημάτων Xilkernel και uClinux σε υλοποιημένο Microblaze επεξεργαστή μιας Virtex-II 1000 FPGA. Τα αποτελέσματα της εργασίας του αφορούσαν την σύγκριση του χρόνου εκτέλεσης των αλγορίθμων αποκλειστικά στον επεξεργαστή ή σε υλοποιημένο συνεπεξεργαστή. Η δυναμική αναδιάταξη χρησιμοποιήθηκε για εναλλαγή των συνεπεξεργαστών χωρίς να επηρεάζεται το υπόλοιπο σύστημα. Τα αποτελέσματα της μελέτης του παρουσιάζονται στον πίνακα 3.

Αλγόριθμος Κρυπτογραφίας	Αριθμός Slices	Throughput (Mbps)				Μέγεθος Bitstream (Bytes)	Χρόνος Αναδιάταξης (ms)
		Xilkernel		uClinux			
		S/W	H/W	S/W	H/W		
Des	499	0.24	3.52	0.24	1.12	62896	318
Triple-Des	815	0.08	2.09	0.08	1.09	65116	323
RC4	614	0.95	3.22	1.18	2.10	70832	347

Πίνακας 3: Αποτίμηση Λειτουργίας Δυναμικά Αναδιατασσόμενης Πλατφόρμας Κρυπτογραφίας

Ο Gonzalez [33] προχώρησε στην χρησιμοποίηση της δυναμικής αναδιάταξης για την βελτίωση της λειτουργίας του κρυπτογραφικού αλγορίθμου IDEA σε μία Virtex 600 FPGA. Η τεχνική που ακολούθησε ήταν να αφαιρέσει τα τμήματα του αλγορίθμου που ήταν υπεύθυνα για την επεξεργασία των κλειδιών και την παραγωγή των υποκλειδιών για τους υπολογιστικούς γύρους του αλγορίθμου. Στην θέση τους χρησιμοποίησε ένα ενσωματωμένο επεξεργαστή ο οποίος υπολόγιζε τα απαιτούμενα υποκλειδιά για την λειτουργία του αλγορίθμου με ένα συγκεκριμένο κλειδί και εν συνεχεία χρησιμοποιώντας δυναμική αναδιάταξη προγραμμάτιζε τις βαθμίδες που

απαιτούσαν τα υποκλειδιά με τις ήδη υπολογισμένες τιμές. Η υλοποίηση του αυτή τον οδήγησε σε 18% με 36% βελτίωση της απαίτησης σε χώρο του αλγορίθμου, ενώ όσο αναφορά την διεκπεραιωτική ικανότητα του αλγορίθμου αυτή βελτιώθηκε κατά 5 φορές σε σχέση με προηγούμενες υλοποιήσεις.

Ο Gonzalez [34] προχώρησε στην υλοποίηση συνεπεξεργαστών κρυπτογραφίας για διάφορους διαδεδομένους αλγορίθμους συμπεριλαμβανομένου του AES, DES, Triple DES. Στην συνέχεια προχώρησε σε σύγκριση της εκτέλεσης των παραπάνω αλγορίθμων στους συνεπεξεργαστές που υλοποίησε σε σχέση με την εκτέλεση του σε υλοποιημένους μικροεπεξεργαστές Microblaze και LEON2. Έπειτα προχώρησε στην υλοποίηση του πρωτοκόλλου SSH μόνο που την διαδικασία της κρυπτογράφησης και αποκρυπτογράφησης με AES και DES αναλάμβαναν οι δικοί του συνεπεξεργαστές και όχι ο επεξεργαστής που έτρεχε όλο το πρωτόκολλο. Η υλοποίηση του συστήματος έγινε σε μια Spartan 3 2000 FPGA και χρησιμοποιήθηκε δυναμική αναδιάταξη για την εναλλαγή των συνεπεξεργαστών ώστε να χωρέσουν μέσα στην FPGA. Αναλυτικά αποτελέσματα της υλοποίησης παρουσιάζονται στο πίνακα 4.

Αλγόριθμος Κρυπτογραφίας	Μήκος Κλειδιού (bit)	Αριθμός Slices	Throughput (Mbps)			
			Microblaze		LEON2	
			S/W	H/W	S/W	H/W
Des	64	3920	1.82	24.0	1.7	18.0
Triple-Des	192	11768	0.2	13.0	0.3	8.5
RC4	128	4492	0.9	32.25	0.92	37.5

Πίνακας 4: Αποτίμηση Λειτουργίας Δυναμικά Αναδιατασσόμενης Πλατφόρμας Κρυπτογραφίας

Ο Granado[35] προχώρησε στην υλοποίηση των κρυπτογραφικών αλγορίθμων IDEA και AES χρησιμοποιώντας δυναμική αναδιάταξη. Για την επιτάχυνση των αλγορίθμων χρησιμοποιήθηκαν τεχνικές όπως η χρήση πολλαπλών Datapath και ομοχειρίας πολλών σταδίων. Η δυναμική αναδιάταξη χρησιμοποιήθηκε για να διοχετεύει στις βαθμίδες τα υποκλειδιά μιας και είχαν αφαιρεθεί οι βαθμίδες επεξεργασίας του κλειδιού και παραγωγής των υποκλειδιών από τον αλγόριθμο. Ο αλγόριθμος υποστήριζε μονό κλειδιά για τα οποία υπήρχαν αποθηκευμένα όλα τα παραγόμενα υποκλειδιά. Το όλο σύστημα αναπτύχθηκε σε μια Virtex- II 6000 FPGA. Τα

αποτελέσματα της εργασίας του παρουσιάζονται στον πίνακα 5. Η τελευταία στήλη αναφέρεται στον χρόνο αναδιάταξης ολόκληρου του πυρήνα κρυπτογραφίας χωρίς όμως να υπολογίζεται ο χρόνος για την μεταφορά του partial bitstream στον επεξεργαστή.

Αλγόριθος Κρυπτογραφίας	Αριθμός Slices	Troughput (Gbps)	Χρόνος Αναδιάταξης (ms)
AES	3.720	24.9	159.77
IDEA	15.016	27.4	159.77

Πίνακας 5: Αποτίμιση Λειτουργίας Δυναμικά Αναδιατασσόμενης Πλατφόρμας Κρυπτογραφίας

4.3 Υλοποιήση Αλγορίθμων Κρυπτογραφίας Υλικό

Στον παρακάτω πίνακα 6 παρουσιάζονται μερικές από τις πιο αποδοτικές στατικές υλοποιήσεις που έχουν καταγραφεί έως σήμερα για την υλοποίηση σε FPGA των αλγορίθμων DES και AES. Η αποδοτικότητα αφορά τόσο την χωρική κατανάλωση όσο και την διεκπεραιωτική ικανότητα των αλγορίθμων.

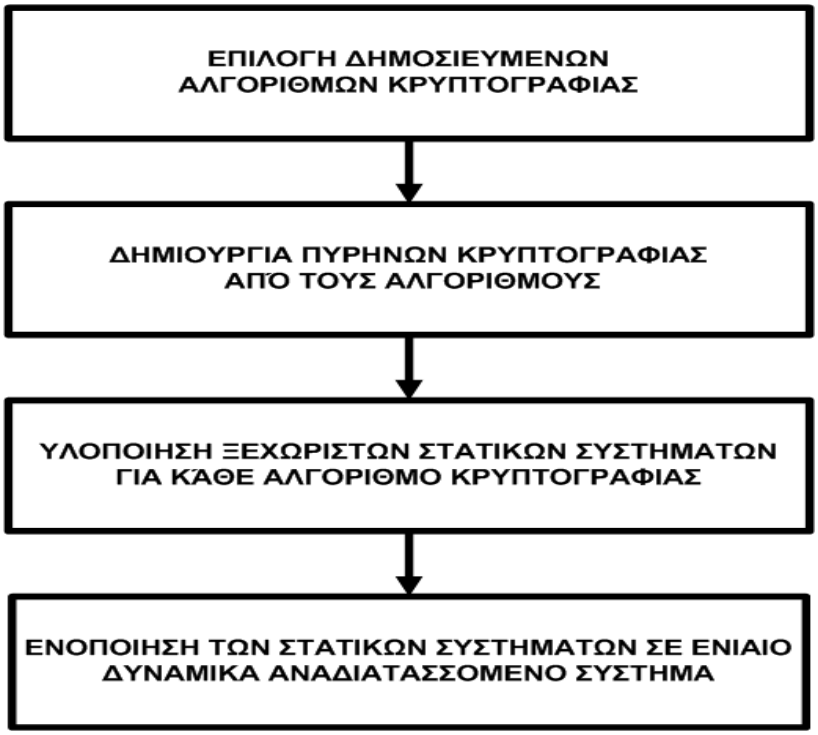
Αλγόριθμος AES				Αλγόριθμος DES			
FPGA	Αριθμός Slices	Troughput (Gbps)	Αναφορά	FPGA	Αριθμός Slices	Troughput (Mbps)	Αναφορά
XCV4000	6842	23.57	[37]	XCV150	1584	10753	[40]
XCV1000	11022	21.56	[38]	XCV1000	6446	3808	[41]
XCV812	12600	12.20	[39]	XCV400	117	274	[42]
XCV812	9406	11.97	[38]	XCV402	741	402.7	[43]
XCV3200	2874	11.78	[36]	XCV402	438	26.7	[44]

Πίνακας 6: Αποδοτικές Υλοποιήσεις Αλγορίθμου AES - AES

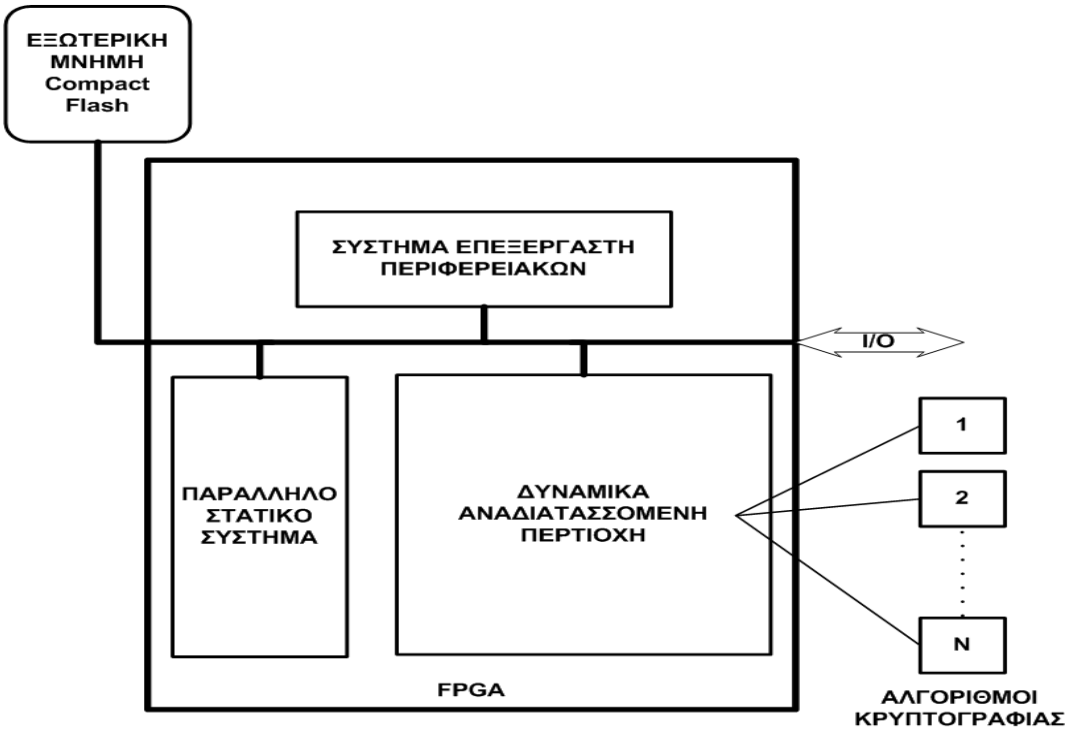
5 Σχεδίαση και Αρχιτεκτονική Συστήματος.

Όπως έχει αναφερθεί στο Κεφάλαιο 1 από τα αρχικά βήματα της σχεδίασης του συστήματος στόχος ήταν η δημιουργία ενός πλήρως αυτόνομου δυναμικά αναδιατασσόμενου συστήματος για την υλοποίηση πολλαπλών κρυπτογραφικών αλγορίθμων. Το σύστημα αυτό θα έπρεπε να έχει την δυνατότητα εύκολης επεκτασιμότητας ώστε να υποστηρίξει μελλοντικά επιπλέον κρυπτογραφικούς αλγόριθμους.

Η ανάπτυξη του συστήματος περιλάμβανε δύο μέρη. Αρχικά την ανάπτυξη στατικού συστήματος κρυπτογραφίας το οποίο θα υποστήριζε έναν μόνο αλγόριθμο κρυπτογραφίας. Συνεπώς υλοποιήθηκαν τρία ξεχωριστά συστήματα για τους τρεις ξεχωριστούς αλγόριθμους που χρησιμοποιήσαμε. Το δεύτερο μέρος της υλοποίησης περιλάμβανε την χρήση δυναμικής αναδιάταξης για την ενοποίηση αυτών των τριών ξεχωριστών συστημάτων ένα ενιαίο σύστημα κρυπτογραφίας υποστήριξης πολλαπλών αλγορίθμων μέσω δυναμικής αναδιάταξης. Η ανάπτυξη του συστήματος έγινε από μηδενική βάση. Σαν δεδομένο υλικό, χρησιμοποιήθηκαν μόνο δημοσιευμένες υλοποιήσεις των αλγορίθμων που χρησιμοποιήσαμε και προήλθαν από το [48]. Οι αλγόριθμοι αυτοί τροποποιήθηκαν όπως περιγράφεται παρακάτω ώστε να είναι δυνατή τόσο η εφαρμογή τους στο Δυναμικά Αναδιατασσόμενο Σύστημα όσο και η αύξηση της απόδοσης όπου αυτό ήταν εφικτό. Το παρακάτω διάγραμμα ροής 17 περιγραφεί συνοπτικά τα βήματα που έγιναν για την ολοκλήρωση της σχεδίασης και της αρχιτεκτονικής του συστήματος. Μια γενική επισκόπηση του συστήματος φαίνεται στο παρακάτω σχήμα 18. Όλο το σύστημα υλοποιείται εντός της FGPA εκτός της μνήμης που αποθηκεύονται τα Partial Bitstream. Εντός της Δυναμικά Αναδιατασσόμενης Περιοχής δύναται να εναλλάσσονται οι πυρήνες κρυπτογραφίας που έχουμε υλοποιήσει. Η συγκεκριμένη ροή ακολουθήθηκε διότι ο σχεδιασμός ενός δυναμικά αναδιατασσόμενου συστήματος απαιτεί εκ των προτέρων την γνώση των απαιτήσεων τόσο σε πόρους όσο και σε συχνότητες ρολογιού των σχεδιάσεων που πρόκειται να αποτυπωθούν. Επίσης λόγω της έλλειψης δυνατότητας ελέγχου πριν την υλοποίηση των εργαλείων που χρησιμοποιήσαμε για τα δυναμικά αναδιατασσόμενα τμήματα η προσέγγιση αυτή προσφέρει ευκολότερο εντοπισμό της πηγής των προβλημάτων εφόσον έχουμε εξασφαλίσει την ορθή λειτουργία των στατικών συστημάτων.



Σχήμα 17: Διαδικασίες που Ακολουθήθηκαν για την Ολοκλήρωση της Σχεδίασης και Αρχιτεκτονικής του Συστήματος



Σχήμα 18: Γενική Περιγραφή του Συστήματος

5.1 Σχεδιαστικά Ζητήματα

Η επιλογή όχι μόνο των εργαλείων που θα χρησιμοποιούσαμε αλλά και της έκδοσης αυτών ήταν κρίσιμη διότι αφενός η Δυναμική Αναδιάταξη δεν υποστηρίζεται από όλες τις εκδόσεις των εργαλείων της Xilinx, αφετέρου δεν υπάρχει επαρκής τεκμηρίωση για την χρήση των εργαλείων στην ανάπτυξη δυναμικά αναδιατασσόμενων συστημάτων. Η παρούσα εργασία κινήθηκε στα πλαίσια της σχεδιαστικής ροής που προτείνει η Xilinx στο EARP [1] τμήμα της ιστοσελίδας της. Η πρόσβαση στον τομέα αυτό δεν είναι ελεύθερη και απαιτείται έγκριση για την είσοδο στο διαθέσιμο υλικό.

Το ISE 9.1.02 [45] για την σχεδίαση, την σύνθεση και την υλοποίηση των τμημάτων του συστήματος των οποίων η λειτουργία περιγράφηκε με γλώσσα περιγραφής υλικού VHDL. Χρησιμοποιήθηκε επίσης για την τελική σύνδεση όλων των επιμέρους συστημάτων και την δημιουργία του τελικού συστήματος.

Το EDK 9.1.02 [46] χρησιμοποιήθηκε για την προσθήκη ενσωματωμένου επεξεργαστή Power PC στο σύστημα μας, για την ενσωμάτωση των απαραίτητων περιφερειακών στον επεξεργαστή, για την υλοποίηση Εισόδου Εξόδου του συστήματος καθώς και για την ανάπτυξη του λογισμικού που θα εκτελεί ο επεξεργαστής μας.

Το PlanAhead 10.1 [47] χάρη στο φιλικό προς τον χρήστη περιβάλλον και την υποστήριξη σχεδιαστικής ροής για δυναμική αναδιάταξη χρησιμοποιήθηκε από την χωρική τακτοποίηση (floorplaning) του συστήματος έως και την παράγωγη των bitstream. Εκτός της υποστήριξης με γραφικό περιβάλλον όλης της σχεδιαστικής ροής για Δυναμική Αναδιάταξη προσφέρει και επιπλέον δυνατότητες όπως ο έλεγχος για DRC σφάλματα αλλά και η αυτόματη τοποθέτηση Bus Macros και η εξαγωγή στατιστικών στοιχείων για τα Δυναμικά Αναδιατασσόμενα τμήματα της σχεδίασης. Η υποστήριξη της αυξανόμενης σχεδίασης (Incremental Design) επιτρέπει την μερική τροποποίηση της σχεδίασης χωρίς να είναι απαραίτητη όλη η ανάπτυξη του συστήματος εξ αρχής παρά μόνο των τμημάτων που επηρεάστηκαν από την αλλαγή. Η δυνατότητα αυτή μας επιτρέπει να πετύχουμε την επιθυμητή απόδοση και λειτουργικότητα του συστήματος με το ελάχιστο χρονικό κόστος.

5.2 Σύστημα Κρυπτογραφίας Χωρίς την Χρήση Δυναμικής Αναδιάταξης

Όπως αναφέρθηκε στα εισαγωγικά αυτού του κεφαλαίου ήταν απαραίτητη η αρχική υλοποίηση στατικών συστημάτων κρυπτογραφίας. Η διαδικασία ξεκίνησε με την υλοποίηση των πυρήνων κρυπτογραφίας.

5.2.1 Υλοποίηση Πυρήνων Κρυπτογραφίας

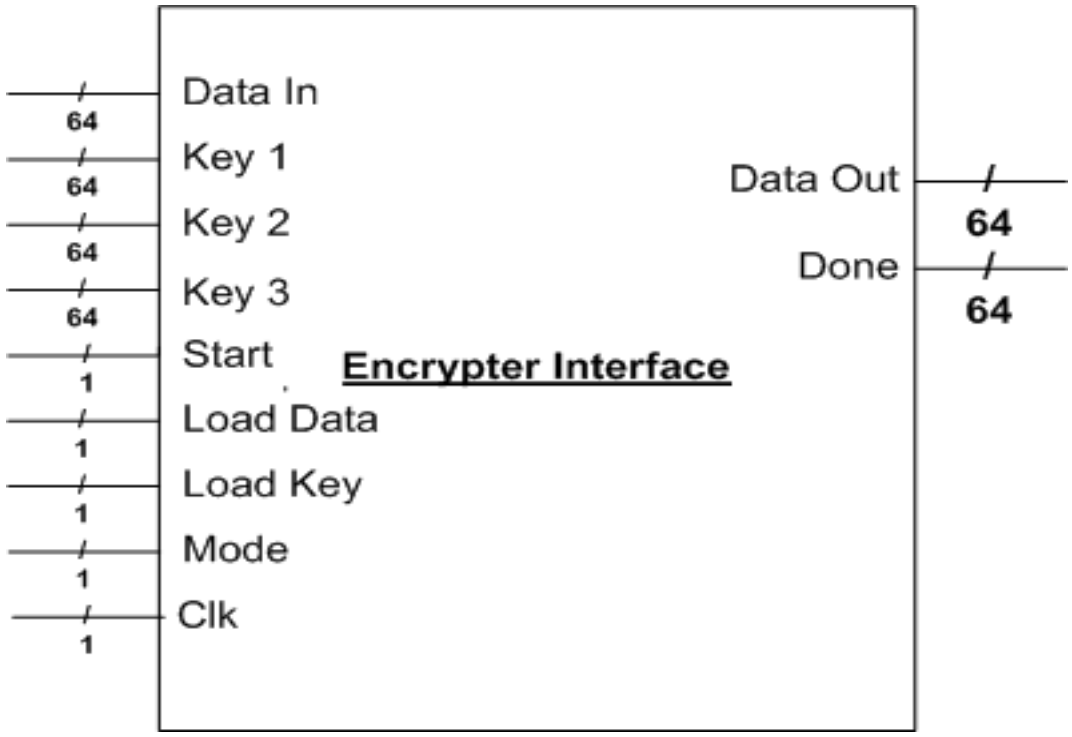
Αρχικοί κώδικες για τους πυρήνες κρυπτογραφίας AES DES και Triple - DES βρέθηκαν στο [48]. Οι κώδικες είχαν δοκιμαστεί για σύνθεση σε άλλες πλατφόρμες, ενώ μόνο ο αλγόριθμος AES είχε αποτυπωθεί σε FPGA και είχε επιβεβαιωθεί η λειτουργιά του πάνω σε αυτή. Καταρχήν έγιναν οι απαραίτητες αλλαγές ώστε οι κώδικες να είναι συνθέσιμοι για την FPGA που χρησιμοποιήσαμε. Η ιεραρχία των αρχείων Vhdl τροποποιήθηκε ώστε να έχουν όλοι οι πυρήνες δενδρική μορφή με μια μόνο ρίζα, δηλαδή ένα μόνο αρχείο στο ρόλο της κορυφαίας οντότητας στο οποίο περιγράφονται όλοι οι είσοδοι και έξοδοι του κάθε πυρήνα κρυπτογραφίας. Αντικαταστάθηκε κώδικας περιγραφής συμπεριφοράς (behavioural) με κώδικα δομικής περιγραφής (structural). Κρίθηκε σκόπιμη η υλοποίηση συγκεκριμένης διεπαφής που θα υλοποιούν οι πυρήνες, όπως ορίζεται από την σχεδιαστική ροή που ακολουθούμε αλλά και επιβάλλεται από την προοπτική ενσωμάτωσης νέων αλγορίθμων στο μέλλον. Η διεπαφή καθορίζεται από τα στοιχεία του πίνακα 7.

Αλγόριθμος Κρυπτογραφίας	Μέγεθος Μηνύματος Εισόδου (bit)	Μέγεθος Κλειδιού (bit)	Μέγεθος Κρυπτογραφημένου Μηνύματος (bit)
DES	64	64	64
Triple Des	64	192	64
AES	128	128	128

Πίνακας 7: Μέγεθος Εισόδου Εξόδου Κρυπτογραφικών Λλγορίθμων

Από την μελέτη των μεγεθών του πίνακα 7 και άλλων αλγορίθμων συμμετρικού κλειδιού αποφασίστηκε η διεπαφή που υλοποιούν οι πυρήνες κρυπτογραφίας να έχουν την μορφή του σχήματος 19. Με την μορφή αυτή υπάρχει μια ισορροπία στον συνολικό αριθμό σημάτων εισόδου εξόδου, που καθορίζουν και τον αριθμό των Bus Macros που θα χρησιμοποιήσουμε, με τον

αριθμό των κύκλων ρολογιού για την εισαγωγή και εξαγωγή δεδομένων.



Σχήμα 19: Διεπαφή Αλγορίθμων Κρυπτογραφίας

Τα δεδομένα εισέρχονται στην είσοδο Data In και το κρυπτογραφημένο μήνυμα λαμβάνεται από την έξοδο Data Out όταν το σήμα Done υποδείξει ότι έχει τελειώσει όλη η διαδικασία. Το σήμα mode καθορίζει εάν ο πυρήνας εκτελεί κρυπτογράφηση ή αποκρυπτογράφηση. Ανάλογα με τον αριθμό των κλειδιών που χρησιμοποιεί ο κάθε αλγόριθμος μπορούν να χρησιμοποιηθούν οι είσοδοι Key1, Key2, Key3. Τα σήματα Load Data, Load Key και Start χρησιμοποιούνται σε περίπτωση που απαιτούνται πέρα του ενός κύκλου για την φόρτωση των δεδομένων.

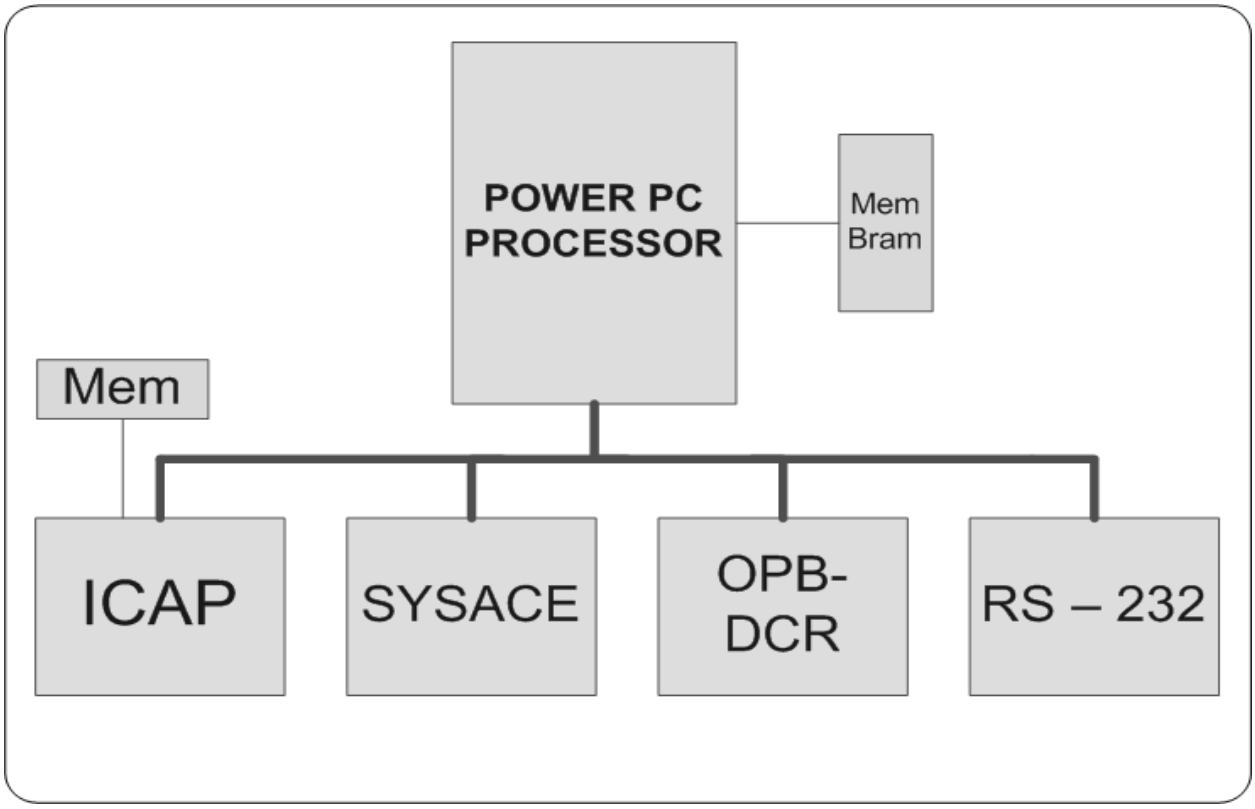
5.2.2 Υλοποίηση Συστήματος Επεξεργαστή Περιφερειακών

Η δημιουργία ενός αυτόνομου συστήματος προϋποθέτει την ύπαρξη ενός επεξεργαστή ο οποίος :

1. Διαχειρίζεται θέματα Εισόδου − Εξόδου.
2. Φροντίζει για την αρχικοποίηση του Συστήματος.
3. Διαχειρίζεται την διαδικασία της Δυναμικής Αναδιάταξης.
4. Υλοποιεί την διεπαφή Χρήστη − Συστήματος και Εξωτερικού PC − Συστήματος.

5. Εκτελεί το αντίστοιχο λογισμικό για τον κάθε Κρυπτογραφικό Αλγόριθμο.

Δεδομένου του γεγονότος ότι η FPGA που επιλέχθηκε διαθέτει 2 ενσωματωμένους επεξεργαστές επιλέξαμε την χρήση ενός εκ των δύο για να εκτελεί της παραπάνω εργασίες. Η διαμόρφωση του συστήματος μαζί με τα περιφερειακά φαίνεται στο σχήμα 20.



Σχήμα 20: Σύστημα Επεξεργαστή και Περιφερειακών

Το σύστημα αποτελείται από :

1. Επεξεργαστή Power PC μαζί με την μνήμη του μεγέθους 64 KByte (Bram).

2. Περιφερειακό RS232 για την υποστήριξη επικοινωνίας της πλατφόρμας με εξωτερικό PC . Η ταχύτητα που επιλέχθηκε ήταν τα 115200 bps. Στο σύστημά μας η επικοινωνία μέσω σειριακής θύρας χρησιμοποιείται για :

   i Μεταφορά δεδομένων προς Κρυπτογράφηση/Αποκρυπτογράφηση στην Πλατφόρμα.

   ii Εξαγωγή των Κρυπτογραφημένων/Αποκρυπτογραφημένων δεδομένων από την Πλατφόρμα.

iii Μεταφορά εντολών του χρήστη ή άλλου εξωτερικού συστήματος για εφαρμογή της δυναμικής αναδιάταξης και αλλαγή της διαμόρφωσης του συστήματος.

iv Μεταφορά από την Πλατφόρμα προς εξωτερικό σύστημα πληροφοριών σχετικά με την τρέχουσα κατάσταση και την διαμόρφωση του συστήματος.

3. Ελεγκτή SysAce για την υποστήριξη εξωτερικής μνήμης Compact Flash . Στην εξωτερική αυτή μνήμη θα αποθηκεύονται :

   i Το συνολικό Bitstream διαμόρφωσης του συστήματος το οποίο θα αρχικοποιεί το σύστημα κατά την εκκίνηση.

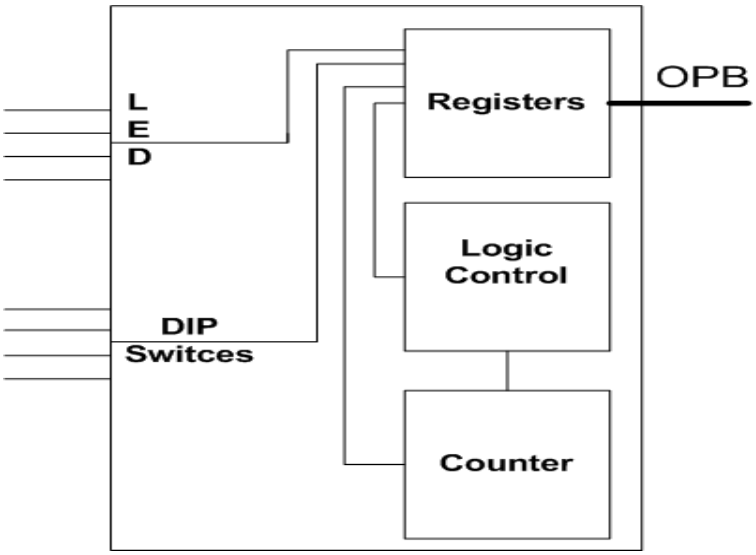
   ii Τα partial Bitstream με τα οποία θα εναλλάσσουμε με την εφαρμογή της Δυναμικής Αναδιάταξης του πυρήνες κρυπτογραφίας.

4. Περιφερειακό Υποδοχής OPB−DCR η οποία χρησιμοποιείται για να μεταφέρουμε μέσω των registers του DCR τον δίαυλο OPB σε ένα εξωτερικό του συστήματος περιφερειακό. Συνεπώς το περιφερειακό που θα εναλλάσσονται οι κρυπτογραφικοί αλγόριθμοι θα είναι συνδεμένο σε μια προέκταση του διαύλου OPB ενώ η διεπαφή σύνδεσης θα είναι ακριβώς όμοια με ένα κοινό OPB περιφερειακό.

5. Περιφερειακό HWICAP με δικιά του αποκλειστική μνήμη Bram το οποίο εξασφαλίζει την επικοινωνία του επεξεργαστή με την βαθμίδα ICAP που βρίσκεται ενσωματωμένη εντός της FPGA. Η μεταφορά των Bytes του Partial Bitstream και η τροποποίηση της μνήμης διαμόρφωσης της FPGA από τον ICAP θα αποτελέσει τον μηχανισμό εναλλαγής των κρυπτογραφικών αλγορίθμων του συστήματος.

5.2.3 Υλοποίηση Στατικού Παράλληλου Συστήματος

Όπως έχουμε αναφέρει και στο Κεφάλαιο 2 που παρουσιάστηκαν γενικά θέματα δυναμικής αναδιάταξης ένα από τα κύρια πλεονεκτήματα της τεχνολογίας αυτής είναι ότι επιτρέπει στο υπόλοιπο σύστημα να παραμένει λειτουργικό κατά την εκτέλεσή της. Για την παρουσίαση της δυνατότητας αυτής αναπτύξαμε ένα στατικό σύστημα το οποίο επικοινωνεί με τον επεξεργαστή μας και εκτελεί επικουρικές εργασίες. Στην θέση αυτού του συστήματος θα μπορούσε να υλοποιηθεί οποιαδήποτε σύστημα το οποίο θα επέκτεινε την λειτουργικότητα του συστήματος και κυρίως η λειτουργία του θα ήταν αδιάκοπτη, χαρακτηριστικό πολύ χρήσιμο σε εφαρμογές πραγματικού χρόνου. Το σύστημα που υλοποιήσαμε φαίνεται στο σχήμα 21.



Σχήμα 21: Δομικό Διάγραμμα Παράλληλου Συστήματος

Η επικοινωνία με τον επεξεργαστή γίνεται μέσω του διαύλου OPB. Το σύστημα μπορεί να χρησιμοποιηθεί για

1. Απεικόνιση πληροφοριών στον χρήστη μέσω τον οπτικών ενδείκτων LED που διαθέτει η πλατφόρμα. Ο χρήστης χρησιμοποιώντας τους διακόπτες DIP Switch μπορεί να παρατηρήσει την εμφάνιση στα LED πληροφοριών όπως :

   i Ποιος πυρήνας κρυπτογραφίας είναι υλοποιημένος μια δεδομένη χρονική στιγμή.

   ii Χρονική στιγμή έναρξης και χρονική στιγμή ολοκλήρωσης της διαδικασίας εναλλαγής των κρυπτογραφικών πυρήνων.

iii Παρακολούθηση μετρητών που συνεχίζουν αδιάλειπτα την λειτουργία τους όσο το υπόλοιπο σύστημα αναδιατάσσεται.

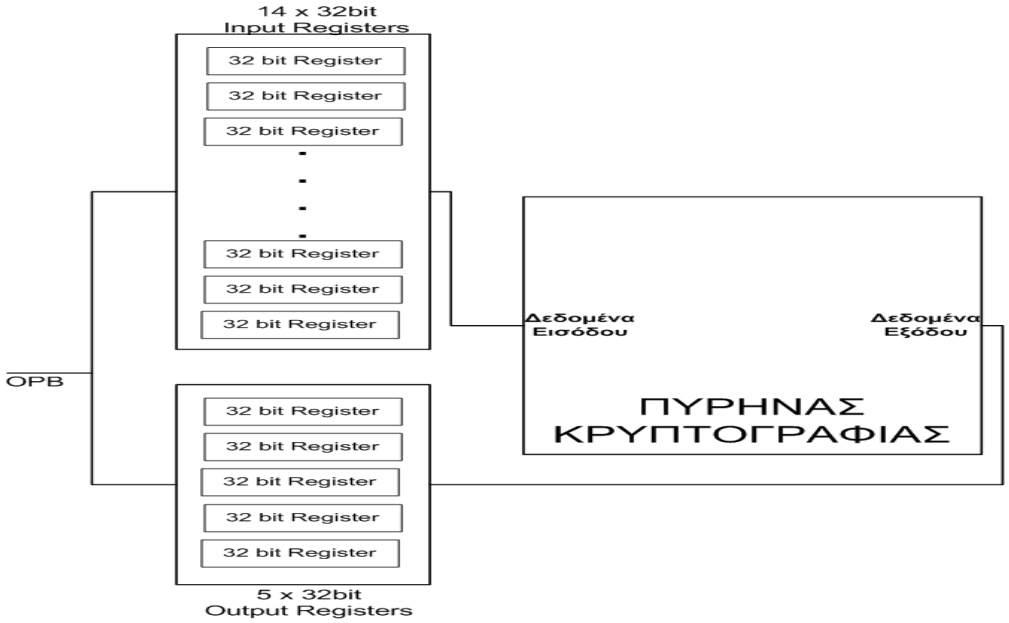
2. Μετρήσεις χρόνων για διάφορες διαδικασίες μέσω των ενσωματωμένων μετρητών που διαθέτει.

5.2.4 Ενοποίηση σε ένα Στατικό Σύστημα

Έπειτα από την ολοκλήρωση όλων των επιμέρους στατικών βαθμίδων απομένει η ένωση τους σε ένα ενιαίο σύστημα.

Μετατροπή Συστημάτων σε Περιφερειακά OPB

Όπως έχει ήδη αναφερθεί ο δίαυλος OPB μέσω της υποδοχής OPB−DCR επεκτείνεται και εκτός του συστήματος Επεξεργαστή Περιφερειακών. Οι πυρήνες κρυπτογραφίας με την βοήθεια του εργαλείου EDK μετατράπηκαν σε περιφερειακά OPB. Εντός των περιφερειακών προστέθηκαν και οι απαραίτητοι καταχωρητές για τα δεδομένα Εισόδου−Εξόδου. Η τελική δομή των OPB περιφερειακών κρυπτογραφίας φαίνεται στο σχήμα 22 ενώ στον πίνακα 8 φαίνεται η αντιστοιχία Εισόδων−Εξόδων και καταχωρητών.



Σχήμα 22: Δομή OPB Περιφεριακών Κρυπτογραφίας

Η ίδια τακτική ακολουθήθηκε και στην μετατροπή του παράλληλου συστήματος το οποίο έχει ήδη παρουσιαστεί στο σχήμα 21.

Καταχωρητής	Σήμα Πυρήνα Κρυπτογραφίας
♯1 ♯2	Data In
♯3 − ♯8	Key1, Key2, Key3
♯9	Mode, Load Data, Load Key, Start
♯10 − ♯15	Data Out
♯16	Ready

Πίνακας 8: Αντιστοιχία Καταχωρήτων και Εισόδων Εξόδων του Πυρήνα Κρυπτογραφίας

Διαμοιρασμός Ρολογιού

Η σύνδεση όλων των υποσυστημάτων μεταξύ τους απαιτούσε την εξασφάλιση ότι όλα τα υποσυστήματα θα τροφοδοτούνται με ρολόι που θα επέτρεπε κατά πρώτον την ορθή και κατά δεύτερον την αποδοτική λειτουργία τους.

Στον πίνακα 9 παρουσιάζονται οι μέγιστες δυνατές συχνότητες ρολογιού όπως αυτές εμφανίστηκαν στις αναφορές αποτελεσμάτων υλοποίησης των εργαλείων ISE και EDK. Η χρήση ενός μόνο ρολογιού για το σύστημα δεν είναι εφικτή διότι το σύστημα Επεξεργαστή Περιφερειακών απαιτεί υποχρεωτικά την χρήση ρολογιού 100 Mhz ώστε να λειτουργεί να παράγεται η ορθή συχνότητα λειτουργίας του ICAP που είναι τα 66 Mhz ενώ η μέγιστη συχνότητα που μπορεί να λειτουργήσει ο αλγόριθμος AES είναι τα 79,3 Mhz. Κάτι τέτοιο άλλωστε θα προκαλούσε και μείωση της απόδοσης των πυρήνων DES και Triple DES .

Σύστημα	Μέγιστη Συχνότητα Ρολογίου
Σύστημα Επεξεργαστή	100 (MHz)
Παράλληλο Στατικό Υποσύστημα	100 (MHz)
Πυρήνας AES	79,3 (MHz)
Πυρήνας DES	170 (MHz)
Πυρήνας Triple Des	170 (MHz)

Πίνακας 9: Μέγιστες δυνατές Συχνότητες Ρολογιού Επιμέρους Συστημάτων

Το πρόβλημα έγκειται κυρίως στο περιφερειακό που υλοποιεί τους πυρήνες κρυπτογραφίας διότι απαιτείται να εισέρχονται όλα τα προαναφερμένα ρολόγια. Τα ρολόγια των 79,3 MHz και των 170 MHz χρησιμοποιούνται από τους κρυπτογραφικούς αλγόριθμους ενώ το ρολόι των 100

MHz για την διεπαφή με τον δίαυλο OPB.

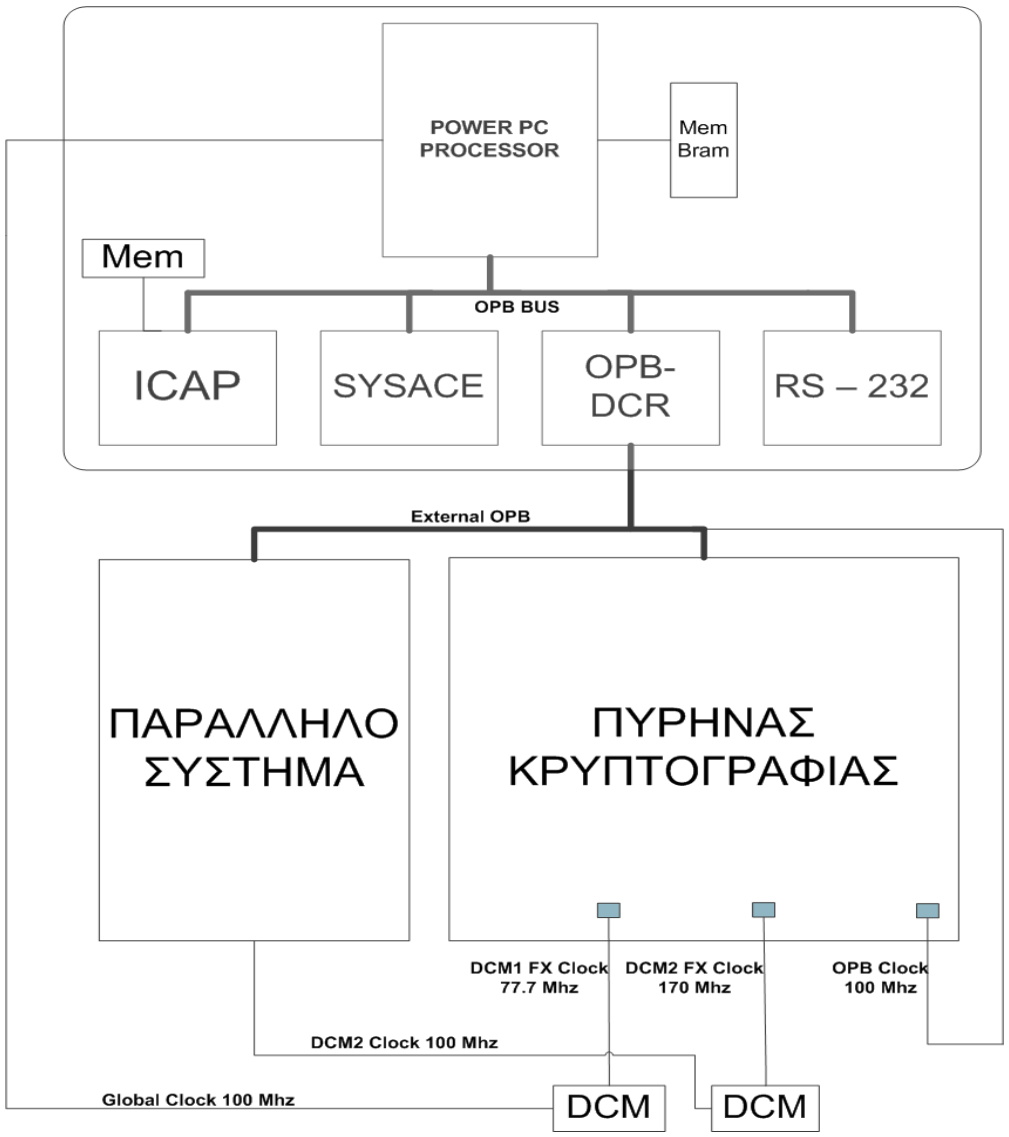
Για την ικανοποίηση των παραπάνω περιορισμών επιλέχτηκε σαν λύση η χρησιμοποίηση δύο από τα οκτώ διαθέσιμα DCM της FPGA. Για να το πετύχουμε αυτό αρχικά αφαιρέσαμε το DCM που προεπιλεγμένα χρησιμοποιεί το EDK για την ανάπτυξη οποιουδήποτε συστήματος επεξεργαστή. Το σύστημα επεξεργαστή τροφοδοτείται πλέον με ρόλοι από το πρώτο DCM που προσθέσαμε του οποίου η έξοδος ρυθμίστηκε στα 100 MHz. Το ρολόι αυτό καθορίστηκε και σαν γενικό ρολόι του συστήματος κάτι που κρίθηκε απαραίτητο για να λειτουργήσουν τα σχεδιαστικά εργαλεία. Η έξοδος του δεύτερου DCM ρυθμίστηκε στα 170 MHz. Η μεταβλητή έξοδο του πρώτου DCM ρυθμίστηκε στα 77,7 MHz. Η διαμόρφωση των DCM φαίνεται στον πίνακα 10 .

DCM	Ρολογιού	Κανονική Έξοδος		Μεταβλητή Έξοδος		
	Εισόδου	Συχνότητα	Σύνδεση	Συχνότητα	Σύνδεση	Πολλ/στής
DCM 1	100 (MHz)	100 (MHz)	Σύστημα Επεξεργαστή	77.7 (MHz)	Περιφερειακό Κρυπτογραφίας	14/18
DCM 2	100 (MHz)	100 (MHz)	Παράλληλο Σύστημα	170 (MHz)	Περιφερειακό Κρυπτογραφίας	17/10

Πίνακας 10: Διαμόρφωση DCM Σχεδίασης.

Επισκόπηση Στατικών Συστημάτων

Η σύνδεση όλων των παραπάνω σε ένα στατικό σύστημα θα οδηγήσει στην δημιουργία τριών πανομοιότυπων συστημάτων κρυπτογραφίας που υλοποιούν ένα μόνο κρυπτογραφικό αλγόριθμο το κάθε ένα. Η γενική μορφή των στατικών συστημάτων κρυπτογραφίας φαίνεται στο σχήμα 23 .



Σχήμα 23: Επισκόπηση Στατικών Συστημάτων.

5.3 Εφαρμογή Δυναμικής Αναδιάταξης για την δημιουργία Συστήματος Υποστήριξης Πολλαπλών Κρυπτογραφικών Αλγορίθμων

Στον πίνακα 11 φαίνονται οι απαιτήσεις πόρων του κάθε υποσυστήματος που υλοποιήσαμε στατικά. Η FPGA που χρησιμοποιούμε διαθέτει 13696 slices ενώ η υλοποίηση του συστήματος απαιτεί 14309 slices . Εύκολα κάποιος συμπεραίνει ότι η χρήση της δυναμικής αναδιάταξης γίνεται απαραίτητη ώστε τα συστήματα που έχουμε δημιουργήσει να μπορέσουν να υλοποιηθούν στους διαθέσιμους πόρους της FPGA αλλά και να μπορούν να επεκταθούν μελλοντικά.

Υποσύστημα	Αριθμός Slices	% Χρήση της FPGA
AES	9671	70,61
DES	815	5,95
Triple − DES	1837	13,41
Επεξεργάστη − Περιφερειακών	1377	10,05
Παράλληλο Σύστημα	600	4,38
Σύνολο	**14300**	**104,40**

Πίνακας 11: Κατανομή των Απαιτήσεων σε Πόρους των Επίμερους Υποσυστημάτων

5.3.1 Ορισμός Δυναμικά Αναδιατασσόμενης Περιοχής

Μελετώντας τον σκοπό της εργασίας και τα στοιχεία του πίνακα είναι προφανές ότι απαιτείται η δημιουργία μιας αναδιατασσόμενης περιοχής στην οποία θα εναλλάσσονται οι τρεις κρυπτογραφικοί αλγόριθμοι τους οποίους υλοποιεί το σύστημα μας. Το μέγεθος της δυναμικά αναδιατασσόμενης περιοχής καθορίζεται από το μέγεθος της μεγαλύτερης σχεδίασης. Αυτό είναι και το κύριο μειονέκτημα της σχεδιαστικής ροής Module Base που χρησιμοποιήσαμε διότι ενώ μεν η μεγαλύτερη σχεδίαση προσαρμόζεται σχεδόν αποδοτικά στην αναδιατασσόμενη περιοχή οι υπόλοιπες σχεδιάσεις δεν χρησιμοποιούν όλους τους διαθέσιμους πόρους. Το γεγονός αυτό προκαλεί

1. Ύπαρξη διαθέσιμων πόρων οι οποίοι δεν μπορούν να χρησιμοποιηθούν διότι δεσμεύονται χωρίς να χρησιμοποιούνται εντός της δυναμικά αναδιατασσόμενης περιοχής.

2. Σπατάλη χρόνου στο επαναπρογραμματισμό μέσω δυναμικής αναδιάταξης τμημάτων της FPGA που δεν υλοποιούν καμία λογική.

Τα ποσοτικά στοιχεία του παραπάνω προβλήματος παρουσιάζονται στον πίνακα 12. Επισημαίνεται ότι το μέγεθος της δυναμικά αναδιατασσόμενης περιοχής δεν μπορούσε να ήταν ακριβώς το μέγεθος που απαιτεί ο αλγόριθμος AES λόγω τον περιορισμών που υπάρχουν ως προς την δέσμευση της δυναμικά αναδιατασσόμενης περιοχής όπως αναφέρονται αναλυτικά στο Κεφάλαιο 2.

Αλγόριθμος	Αριθμός Slices	% Χρήση της Δυναμικά Αναδιατασσόμενης Περιοχής
AES	9671 από 9920	97,48 %
DES	815 από 9920	8,21 %
Triple − DES	1837 από 9920	18,51 %

Πίνακας 12: % Χρήση της Δυναμικά Αναδιατασσόμενης Περιοχής

5.3.2 Χωρική Τοποθέτηση Δυναμικά Αναδιατασσόμενης Περιοχής

Δεδομένης της απαίτησης μιας δυναμικά αναδιατασσόμενης περιοχής η οποία απαιτεί 9920 από τα 13696 Slices (72,42 %) η χωρική τοποθέτηση της δεν ήταν μια εύκολη διαδικασία. Η τοποθέτηση της περιοχής έπρεπε να συνάδει με τους αρκετούς περιορισμούς οι οποίοι είτε ήταν εξαρχής γνωστοί είτε ανακαλύφθηκαν κατά την διαδικασία υλοποίησης του συστήματος. Αρχικά εξωτερικά της πλευράς της δυναμικά αναδιατασσόμενης περιοχής θα έπρεπε να εξασφαλίσουμε ότι υπάρχουν διαθέσιμες δύο στήλες από slices ώστε να είναι δυνατή η μετέπειτα τοποθέτηση των Bus Macros. Επίσης οι πλευρές της δυναμικά αναδιατασσόμενης περιοχής που δεν περιείχαν Bus Macros δεν θα έπρεπε να εφάπτονται στα σύνορα της FPGA διότι θα υπήρχε πρόβλημα με την διασύνδεση των IOB που βρίσκονται περιφερειακά της FPGA. Από τις Bram και τους ενσωματωμένους πολλαπλασιαστές που περιέχονται μέσα στην δυναμικά αναδιατασσόμενη περιοχή αφαιρέθηκε η δυνατότητα να αναδιατάσσονται δυναμικά έτσι ώστε να είναι δυνατή η χρησιμοποίηση τους από το υπόλοιπο σύστημα.

Η παρουσία των δύο ενσωματωμένων επεξεργαστών εντός της δυναμικά αναδιατασσόμενης περιοχής ήταν αναπόφευκτη δεδομένου ότι οι επεξεργαστές βρίσκονται τοποθετημένοι συμμετρικά στο κέντρο της FPGA. Παρόλο που ο κύριος επεξεργαστής βρίσκονταν εντός των

συνόρων μιας δυναμικά αναδιατασσόμενης περιοχής, αποτελώντας όμως στατικό κομμάτι όλης της σχεδίασης αφού δεν υποστηρίζεται οποιαδήποτε μεταβολή σε αυτόν με την χρήση Δυναμικής Αναδιάταξης, η μνήμη του επιλέχθηκε να βρίσκεται σε μια Bram εκτός της Δυναμικά Αναδιατασσόμενης Περιοχής και συγκεκριμένα στην αριστερότερη στήλη Bram της FPGA για αποφυγή περαιτέρω προβλημάτων που είχαν παρουσιαστεί σε δοκιμαστικές σχεδιάσεις τους συστήματος.

Ένας δεύτερος περιορισμός που δεν αναφερόταν στην τεκμηρίωση της Xilinx ήταν σχετικά με τον ICAP. Παρόλο που ο ICAP βρίσκεται ήδη υλοποιημένος στην FPGA για την χρησιμοποίηση του απαιτείται η υλοποίηση πάνω στην αναδιατασσόμενη περιοχή του HWICAP που αποτελεί την διεπαφή του ICAP με την σχεδίαση του χρήστη. Ο ICAP βρίσκεται στο δεξιότερη κάτω περιοχή της FPGA και παρατηρήθηκε ότι η ύπαρξη δυναμικά αναδιατασσόμενης περιοχής που περιέχουν ίδιες στήλες Slices με την περιοχή του ICAP δημιουργούσε περιοδικά πρόβλημα και πάγωμα του ICAP κατά την διάρκεια της αναδιάταξης των οποίων η προέλευση και τα αίτια δεν ήταν δυνατό να εντοπιστούν.

Με βάση τα παραπάνω επιλέχθηκε ο ορισμός μια Δυναμικά Αναδιατασσόμενης Περιοχής η οποία ισαπέχει από όλες της πλευρές της FPGA. Η απόσταση από τον άνω και το κάτω άκρο είναι δύο γραμμές από Slices ενώ από το δεξιό και το αριστερό άκρο είναι πέντε Slices ώστε να μην αντιμετωπίζεται πρόβλημα με την σύνδεση των στατικών σχεδιάσεων περιφερειακά της Δυναμικά Αναδιατασσόμενης Περιοχής. Τα όρια της περιοχής σε αντιστοιχία με το καρτεσιανό επίπεδο συντεταγμένων είναι από X10 Y154 έως X80 Y5 .

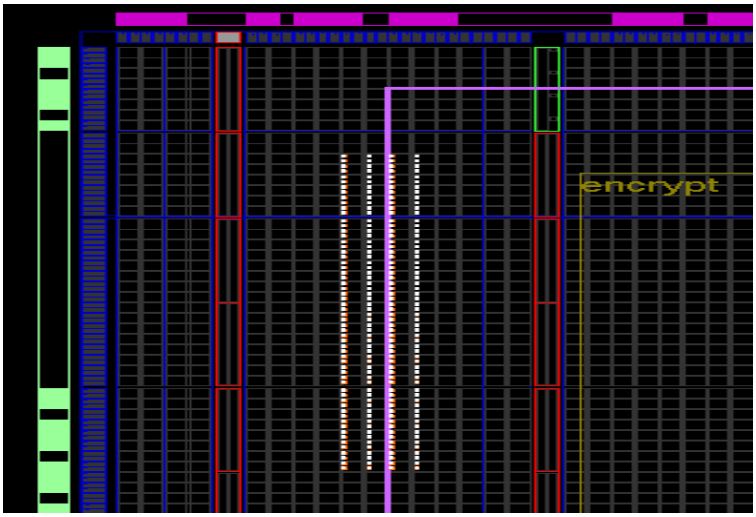
5.3.3 Τοποθέτηση των *Bus Macros*

Όλοι οι πυρήνες κρυπτογραφίας έχουν υλοποιηθεί σαν περιφερειακά OPB. Συνεπώς πρέπει να επιτρέψουμε στην προέκταση του διαύλου OPB που έχουμε κατασκευάσει να εισέλθει εντός της Δυναμικά Αναδιατασσόμενης Περιοχής ώστε να είναι εφικτή η επικοινωνία του εκάστοτε πυρήνα κρυπτογραφίας με το σύστημα επεξεργαστή. Τα σήματα του OPB BUS καθώς και ο αριθμός και ο τύπος των Bus Macros που χρησιμοποιούμε φαίνονται στον πίνακα 13. Είναι γνωστό ότι τα Bus Macros για την οικογένεια της FPGA που χρησιμοποιούμε έχουν συγκεκριμένη κατεύθυνση. Συνεπώς λόγω του γεγονότος ότι όλα τα Bus Macros τοποθετήθηκαν στην αριστερότερη πλευρά της Δυναμικά Αναδιατασσόμενης Περιοχής χρησιμοποιήσαμε Bus Macros LeftToRight για τα σήματα εισόδου και RightToLeft για τα σήματα εξόδου. Το σήμα

enable χρησιμοποιήθηκε για τα Bus Macros των σημάτων εξόδου διότι κατά την διάρκεια της Δυναμικής Αναδιάταξης τα σήματα δύναται να πάρουν διάφορες τιμές, γεγονός που θα δημιουργούσε πρόβλημα στα υπόλοιπα περιφερειακά που θα ήταν συνδεμένα στο ίδιο δίαυλο. Η χωρική τους τοποθέτηση μέσω του εργαλείου PlanAhead φαίνεται στο σχήμα 24.

Τύπος	Μέγεθος	Αριθμός Bus Macros	Τύπος Bus Macros
Δίαυλος Δεδομένων Εισόδου	32 bit	4	LeftToRight
Δίαυλος Δεδομένων Εξόδου	32 bit	4	RightToLeft−Enable
Δίαυλος Διευθύνσεων	32 bit	4	LeftToRight
Σήματα Ελέγχου Εισόδου	8 bit	2	LeftToRight
Σήματα Ελέγχου Εξόδου	4 bit	1	RightToLeft−Enable
Σύνολο	110 bit	15	

Πίνακας 13: Κατανομή Bus Macros για την επικοινωνία με Δ.Α.Π.

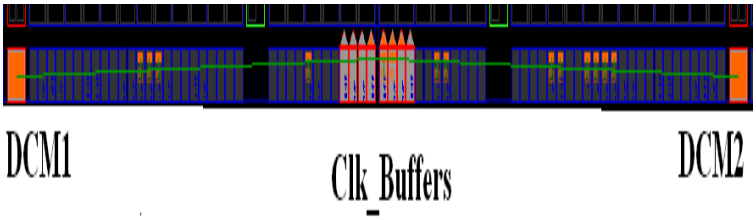


Σχήμα 24: Τοποθέτηση Bus Macros.

5.3.4 Τοποθέτηση των *DCM*

Τα DCM τοποθετήθηκαν στο κάτω άκρο της FPGA στις θέσεις DCM−X2Y0 και DCM−X3Y0 ώστε να είναι όσο το δυνατό πιο κοντά στην δυναμικά αναδιατασσόμενη περιοχή. Το εξωτερικό ρολόι πριν οδηγηθεί στην είσοδο των DCM πέρασε μέσα από BUFG και IBUFG τα οποία βρίσκονται ανάμεσα στα δύο DCM. Η τοποθέτηση των DCM χειροκίνητα κρίθηκε απαραίτητη

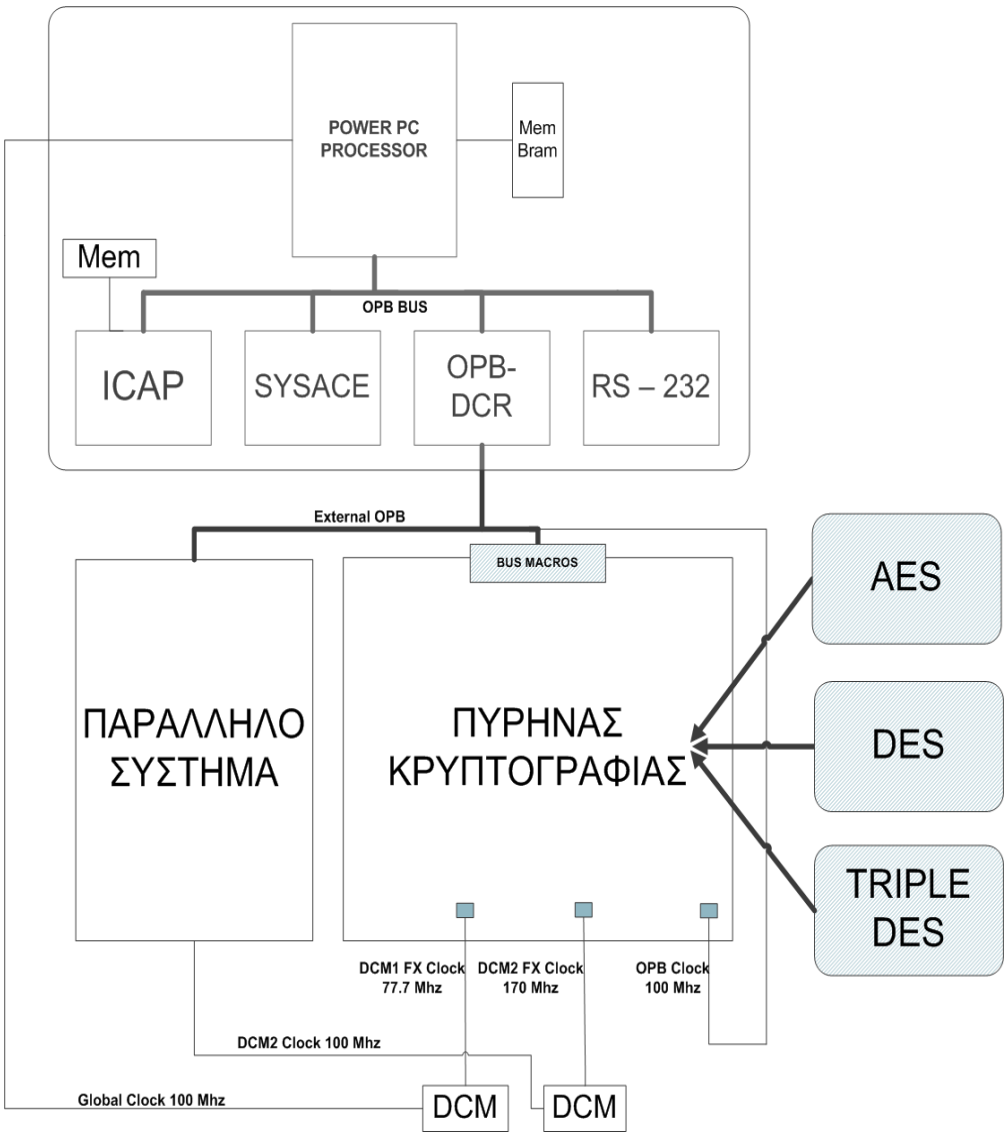
διότι διαφορετικά τα εργαλεία δεν αναγνώριζαν ότι οι έξοδοι των DCM ήταν σήματα ρολογιού και απαιτούσαν την διέλευση μέσω Bus Macros . Ένα μειονέκτημα της τεχνολογίας είναι ότι δεν υποστηρίζεται η δυναμική αλλαγή των παραμέτρων των DCM συνεπώς νέες σχεδιάσεις που μελλοντικά ενδέχεται να προστεθούν στο σύστημα μας πρέπει να χρησιμοποιούν ένα από τα ήδη υπάρχοντα διαθέσιμα ρολόγια. Η χωρική τους τοποθέτηση μέσω του εργαλείου PlanAhead φαίνεται στο σχήμα 25.



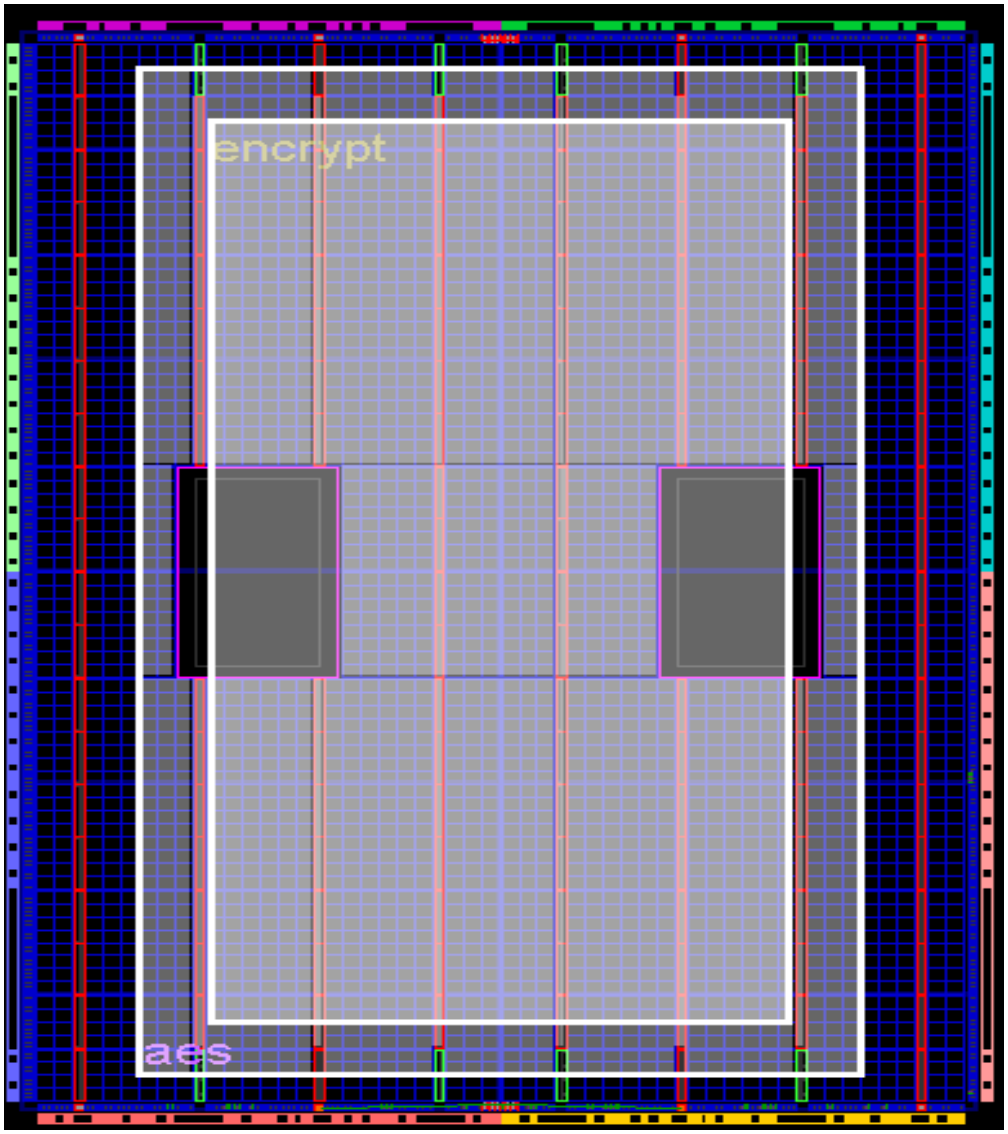
Σχήμα 25: Τοποθέτηση DCM και Clock Buffer .

5.3.5 Επισκόπηση Τελικού Συστήματος.

Στο σχήμα φαίνεται η διαμόρφωση του τελικού μας συστήματος. Οι γραμμοσκιασμένες περιοχές δείχνουν τα κύριες αλλαγές που εμφανιζόνται σε σχέση με τα στατικά συστήματα που είχαμε υλοποιήσει αρχικά. Στο σχήμα 27 φαίνεται μια γενική άποψη του συστήματος όπως αυτή προέκυψε μέσω του εργαλείου PlanAhead.



Σχήμα 26: Δυναμικά Αναδιατασσόμενο Σύστημα Κρυπτογραφίας.

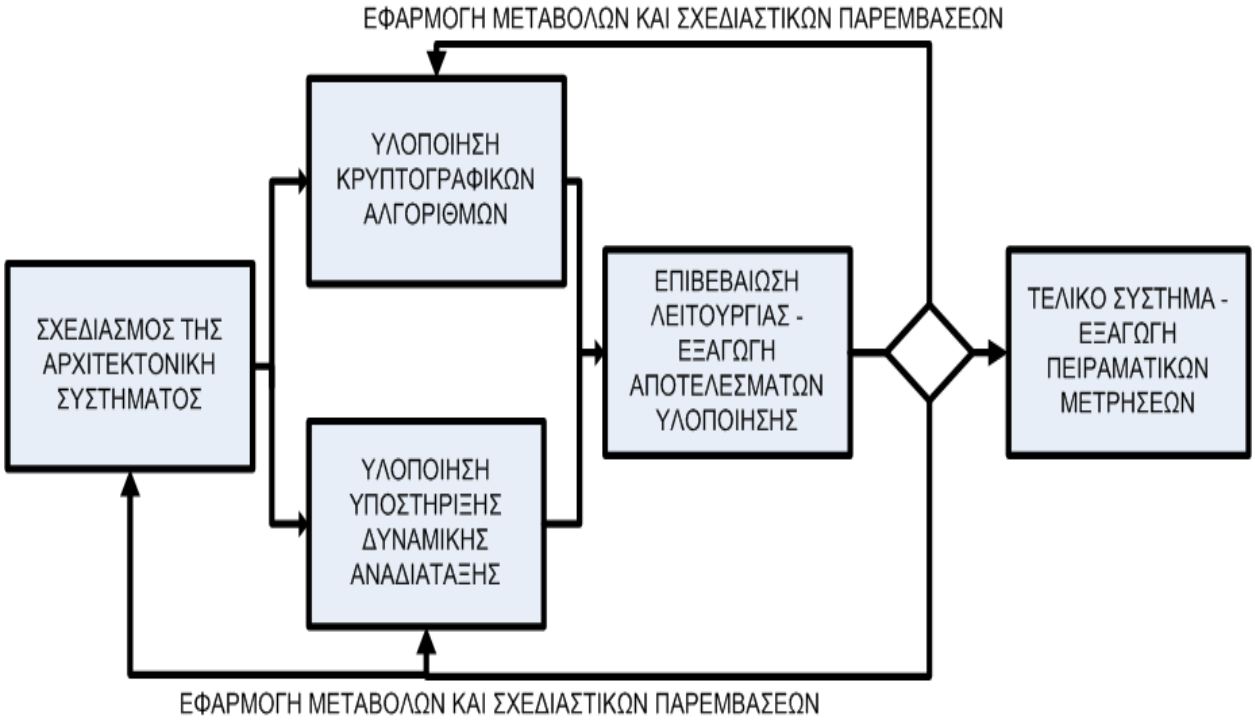


Σχήμα 27: Άποψη μέσω PlanAhead του Δυναμικά Αναδιατασσόμενου Συστήματος Κρυπτογραφίας.

.

6 Υλοποιήση Συστήματος , Επιβεβαίωση Λειτουργίας και Πειραματικά Αποτελέσματα.

Στο παρόν κεφάλαιο παρουσιάζεται ο τρόπος υλοποίησης των βασικών υποσυστημάτων που συνθέτουν την αρχιτεκτονική του συστήματος καθώς και τα αποτελέσματα υλοποίησης του συστήματος μας.Για την επιτυχής ολοκλήρωση του συστήματος απαιτούνταν η ταυτόχρονη ολοκλήρωση τόσο των κρυπτογραφικών αλγορίθμων όσο και του συστήματος που θα υποστήριζε την δυναμική αναδιάταξη. Η ταυτόχρονη υλοποίηση ήταν απαραίτητη γιατί σχεδιαστικές επιλογές του ενός τμήματος επηρέαζαν άμεσα και το άλλο τμήμα. Έπειτα από την ολοκλήρωση και την ενοποίηση σε ένα ενιαίο σύστημα ακολούθησε η επιβεβαίωση της λειτουργίας και η εξαγωγή των αποτελεσμάτων υλοποίησης. Βάση αυτών, γινόταν σχεδιαστικές παρεμβάσεις τόσο στην αρχική σχεδίαση του συστήματος όσο και στον τρόπο υλοποίησης των επιμέρους υποσυστημάτων έως ότου το σύστημα λειτουργήσει ορθά και με την επιθυμητή απόδοση. Μια απλοποιημένη σχεδιαστική ροή που ακολουθήθηκε παρουσιάζεται στο παρακάτω σχήμα 28.



Σχήμα 28: Σχεδιαστική ροή Υλοποίησης Συστήματος

6.1 Αλγόριθμοι Κρυπτογραφίας

Η λειτουργία του κάθε αλγόριθμου κρυπτογραφίας έχει αναλυτικά παρουσιαστεί στο Κεφάλαιο 3. Επίσης κάποια αποτελέσματα υλοποίησης που επηρέασαν άμεσα τις σχεδιαστικές μας επιλογές έχουν ήδη αναφερθεί στο Κεφάλαιο 5. Στις παρακάτω υποενότητες παρουσιάζονται ζητήματα που αφορούν την υλοποίηση των αλγορίθμων.

6.1.1 Χαρακτηριστικά Αλγορίθμων Κρυπτογραφίας

Ο πίνακας 14 παρουσιάζει τα χαρακτηριστικά του κάθε αλγορίθμου κρυπτογραφίας. Η διεκπεραιωτική ικανότητα των Πυρήνων Κρυπτογραφίας υπολογίστηκε βάση του παρακάτω τύπου:

$$P = \frac{S(bit) * F(Mhz)}{cc} \tag{3}$$

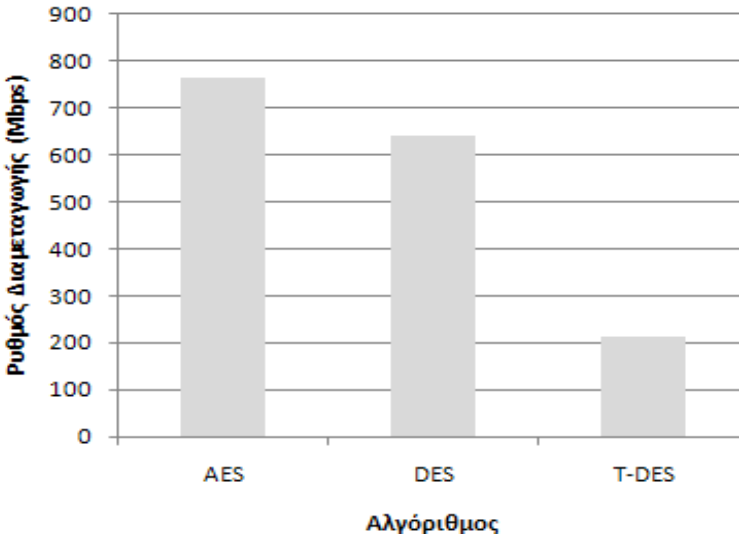
όπου P η διεκπεραιωτική ικανότητα, S το μέγεθος του μηνύματος εισόδου σε bit, F η συχνότητα λειτουργίας του πυρήνα σε MHz και cc ο αριθμός των κύκλων ρολογιού για την εξαγωγή του αποτελέσματος.

$$P_{AES} = \frac{128bit * 77,7Mhz}{13}, \quad P_{DES} = \frac{64bit*170Mhz}{17}, \quad P_{T-DES} = \frac{64bit * 170Mhz}{51}$$
$$= 765Mbps, \qquad = 640Mbps, \qquad = 213,3Mbps$$

Αλγόριθμος	AES	DES	Triple Des
Μέγεθος Μηνύματος Εισόδου (bit)	128	64	64
Μέγεθος Μηνύματος Εξόδου (bit)	128	64	64
Μέγεθος Κλειδιού (bit)	128	64	192
Μέγιστη Συχνότητα Λειτουργίας (MHz)	77,7	170	170
Αριθός Κύκλων για την Εξαγωγή Αποτελέσματος	13	17	51
Ρυθμός Διαμεταγωγής (Mbps)	765	640	213,3

Πίνακας 14: Χαρακτηριστικά Αλγορίθμων

Στο σχήμα 29 παρουσιάζεται γραφικά η σύγκριση της διεκπεραιωτικής ικανότητας των τριών αλγορίθμων.



Σχήμα 29: Σύγκριση Διεκπεραιωτικής Ικανότητας Αλγορίθμων Κρυπτογρφίας

6.1.2 Κατανάλωση Πόρων Εντός Δυναμικά Αναδιατασσόμενης Περιοχής

Όπως έχει ήδη αναφερθεί από το Κεφάλαιο 5 η δυναμικά αναδιατασσόμενη περιοχή περιλαμβάνει εσωτερικά και τους δύο ενσωματωμένους επεξεργαστές Power PC. Στο σχήμα 15 παρουσιάζονται τα χαρακτηριστικά της Δυναμικά Αναδιατασσόμενη περιοχής ενώ μπορούμε βάση αυτών των μεγεθών και να υπολογίσουμε και τον αριθμό των Slices που καταλαμβάνουν χωρικά οι δύο Power PC. Η αφαίρεση του αναμενόμενου αριθμού Slices από το πραγματικό αριθμό μας δείχνει ότι και οι δύο Power PC καταλαμβάνουν τόσο χώρο όσο θα καταλάμβαναν 1024 Slices εντός της FPGA .

Στήλες Slices	72
Γραμμές Slices	152
Αριθμός Slices	9920
Αναμενόμενος Αριθμός Slices	10944
PPC Slices	1024

Πίνακας 15: Χαρακτηριστικά Δυναμικά Αναδιατασσόμενης Περιοχής

Η κατανάλωση πόρων των τριών αλγορίθμων AES, DES, Triple Des εντός της δυναμικά αναδιατασσόμενης περιοχής φαίνεται στο σχήμα 16, σχήμα 17, σχήμα 18 αντίστοιχα.

Τύπος	Διαθέσιμα	Απαιτούμενα	% Χρησιμοποίηση
LUT	19840	15853	79.90
Flip Flop	19840	1678	8.46
Slices	9920	9671	97.49
Multiplier	92	0	0
BRam16	92	0	0

Πίνακας 16: Κατανάλωση Πόρων AES εντός Δυναμικά Αναδιατασσόμενης Περιοχής

Τύπος	Διαθέσιμα	Απαιτούμενα	% Χρησιμοποίηση
LUT	19840	1014	5,11
Flip Flop	19840	748	3,77
Slices	9920	815	8,21
Multiplier	92	0	0
BRam16	92	0	0

Πίνακας 17: Κατανάλωση Πόρων DES εντός Δυναμικά Αναδιατασσόμενης Περιοχής

Τύπος	Διαθέσιμα	Απαιτούμενα	% Χρησιμοποίηση
LUT	19840	3008	15.16
Flip Flop	19840	1700	8.50
Slices	9920	1837	18.51
Multiplier	92	0	0
BRam16	92	0	0

Πίνακας 18: Κατανάλωση Πόρων Triple Des εντός Δυναμικά Αναδιατασσόμενης Περιοχής

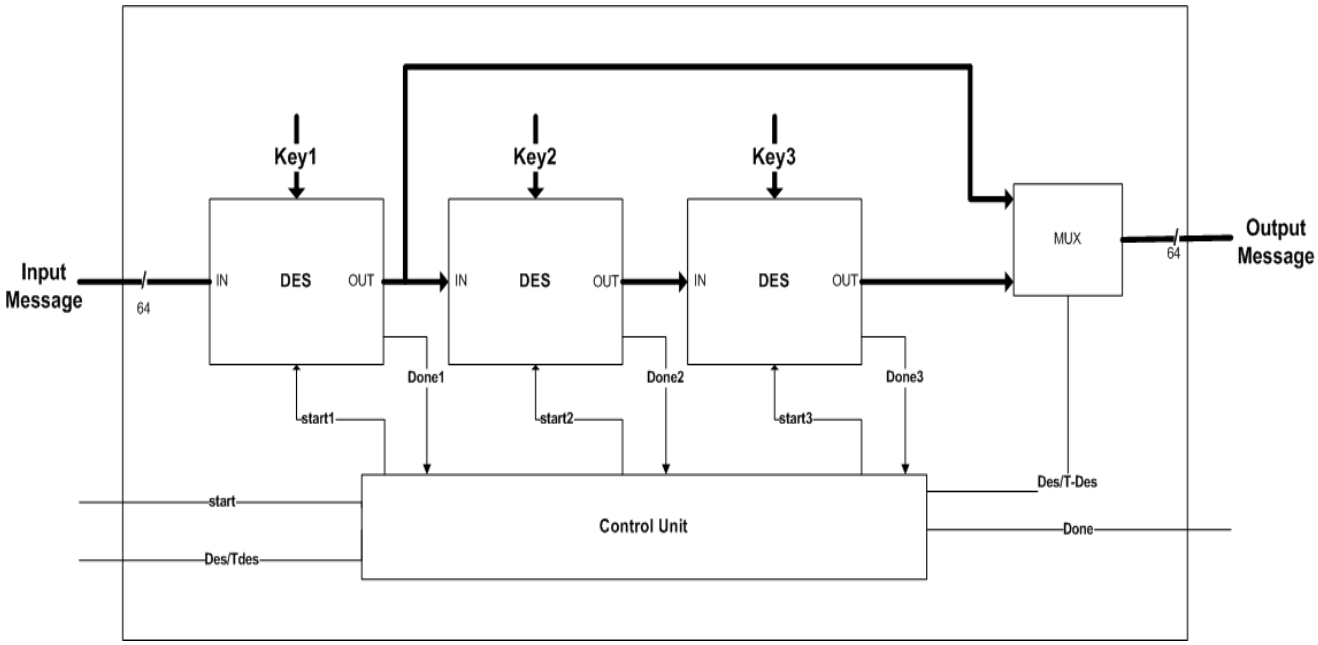
6.1.3 Μετατροπές βάση των Αποτελεσμάτων Υλοποίησης

Τα αποτελέσματα υλοποίησης των προηγούμενων παραγράφων μας οδήγησαν στις παρακάτω αλλαγές :

6.1.3.1 Ενοποίηση Πυρήνων Des και Triple Des.

Η υλοποίηση του αλγορίθμου DES εντός της Δυναμικά Αναδιατασσόμενης Περιοχής καταλαμβάνει μόνο το 8,21 %. Συνεπώς σε κάθε επιλογή του αλγορίθμου DES πρώτον αναδιατάσσεται ολόκληρη η περιοχή και δεύτερον το μεγαλύτερο ποσοστό διαθέσιμων πόρων της FPGA μένει ανεκμετάλλευτο. Ο πυρήνας DES υπάρχει σε τρία αντίγραφα εντός του Triple Des. Συνεπώς αποφασίστηκε η τροποποίηση του Αλγορίθμου Triple Des ώστε να μπορεί να λειτουργήσει και σαν DES χωρίς να απαιτείται η χρήση Δυναμικής Αναδιάταξης αλλά και χωρίς την μείωση της απόδοσης σε σχέση με τον αποκλειστικό πυρήνα DES. Αυτό θα επιτρέπει την γρήγορη εναλλαγή από τον αλγόριθμο DES σε Triple DES και αντίστροφα χωρίς να πληρώνουμε το χρονικό κόστος της αναδιάταξης της περιοχής ενώ το ποσοστό των ανεκμετάλλευτων

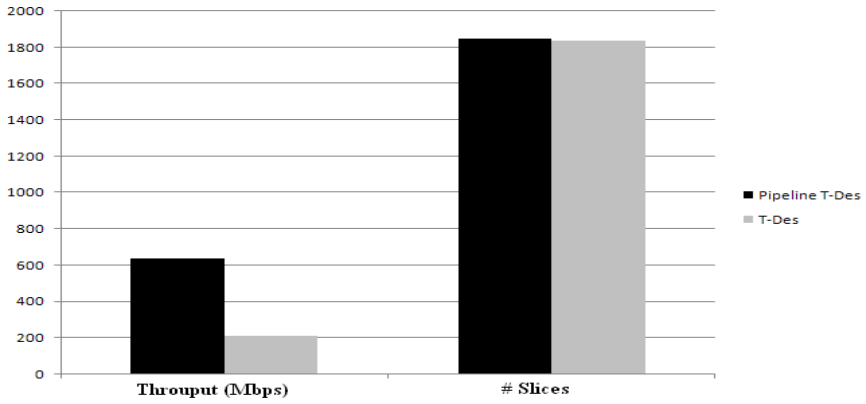
πόρων εντός της δυναμικά αναδιατασσόμενης περιοχής θα είναι μικρότερο. Η αλλαγή που γίνεται στο πυρήνα του Triple Des φαίνεται στο σχήμα 30. Η χρήση του πρώτου πυρήνα DES του αλγορίθμου Triple Des είναι διπλή. Χρησιμοποιείται κανονικά σαν ένας από τους τρείς πυρήνες που συνθέτουν τον Triple Des αλλά με την προσθήκη ενός πολυπλέκτη στην έξοδο δίνεται η δυνατότητα να παρακάμπτονται οι δύο υπόλοιπες βαθμίδες του συστήματος κάνοντας έξοδο του συστήματος την έξοδο της πρώτης βαθμίδας και προσομοιώνοντας ακριβώς την λειτουργία του αλγορίθμου DES. Ένα επιπλέον σήμα από τον καταχωρητή ελέγχου χρησιμοποιείται για τον έλεγχο του πολυπλέκτη και την επιλογή του αλγορίθμου. Αξιοσημείωτο είναι το γεγονός ότι μπορεί και ο αλγόριθμος Triple Des να λειτουργήσει χωρίς περαιτέρω τροποποίηση αν επιλέξουμε σαν είσοδο και για τα τρία κλειδιά που δέχεται ο αλγόριθμός το κλειδί που επιθυμούμε για DES κρυπτογράφηση. Τότε θα είχαμε κρυπτογράφηση με το πρώτο κλειδί, αποκρυπτογράφηση με το δεύτερο κλειδί άρα οι δυο πρώτες βαθμίδες αλληλοαναιρούνται και η τρίτη βαθμίδα θα πραγματοποιούσε κρυπτογράφηση με το επιθυμητό κλειδί. Αυτό θα προκαλούσε μείωση της διεκπεραιωτικής ικανότητας του αλγορίθμου DES κατά τρεις φόρες.



Σχήμα 30: Μετατροπή του Αλγορίθμου Triple Des για υποστήριξη και λειτουργικότητας DES .

6.1.3.2 Εφαρμογή Ομοχειρίας στα τρία στάδια κρυπτογράφησης του Αλγορίθμου Triple Des.

Η ύπαρξη τριών ανεξαρτήτων πυρήνων DES εντός του πυρήνα Triple DES έκανε αυτονόητη την ανάγκη τροποποίησης της μονάδος ελέγχου του αλγορίθμου ώστε να υποστηρίζεται ομοχειρία σε επίπεδο πυρήνων DES. Μόλις ολοκληρωθεί η λειτουργία του πρώτου πυρήνα DES το σύστημα είναι έτοιμο να δεχτεί νέα δεδομένα προς κρυπτογράφηση στην είσοδο. Τα παραπάνω ισχύουν για την περίπτωση που τα κλειδιά για τα οποία εκτελείται η κρυπτογράφηση παραμένουν τα ίδια. Στην περίπτωση ύπαρξης συνεχούς ροής δεδομένων έχουμε κρυπτογραφημένα δεδομένα κάθε 21 κύκλους ρολογιού όσο δηλαδή και η καθυστέρηση του αλγορίθμου DES και άρα η διεκπεραιωτική ικανότητα του αλγορίθμου είναι 640 Mbps ενώ η αύξηση της χωρικής κατανάλωσης από την προσθήκη ομοχειρίας και την ενοποίηση τών πυρήνων ητάν αμελητέα. Τα παραπάνω συγκριτικά αποτελέσματα παρουσιάζονται στο σχήμα 31.



Σχήμα 31: Σύγκριση Σχεδιαστικών Παρεμβάσεων στον Αλγόριθμο Triple Des.

6.1.4 Ανάπτυξη Λογισμικού για Υποστήριξη Αλγορίθμων

Πέραν της υποστήριξης της Δυναμικής Αναδιάταξης για την εναλλαγή των πυρήνων εντός της Δυναμικά Αναδιατασσόμενης Περιοχής ήταν απαραίτητη η ανάπτυξη του κατάλληλου λογισμικού που θα εκτελείτε στον επεξεργαστή του συστήματος και θα διαχειρίζεται την Είσοδο−Έξοδο αλλά και την λειτουργία των συνεπεξεργαστών κρυπτογραφίας. Οι συναρτήσεις λογισμικού που υλοποιήθηκαν φαίνονται στον πίνακα 19. Για κάθε αλγόριθμο υλοποιήθηκαν δύο συναρτήσεις. Η πρώτη φροντίζει για την αρχικοποίηση των κλειδιών, για την επιλογή κρυπτογράφησης ή αποκρυπτογράφησης και έπειτα για την αποστολή του μηνύματος εισόδου και την εξαγωγή

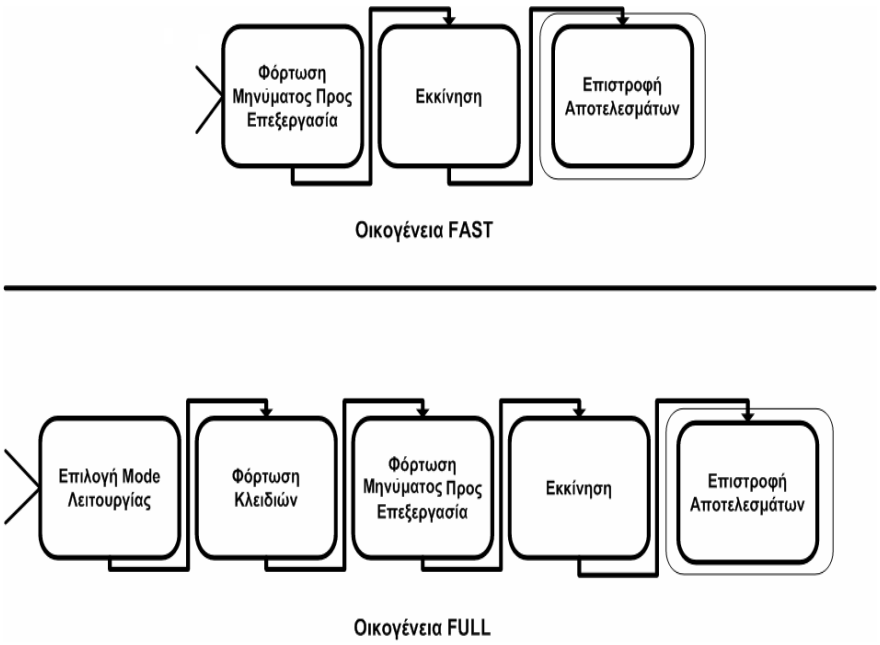
του αποτελέσματος. Η δεύτερη απλώς αποστέλλει το μήνυμα εισόδου στον συνεπεξεργαστή κρυπτογραφίας και επιστρέφει το αποτέλεσμα. Η επιλογή αυτή έγινε λόγω του γεγονότος ότι οι περισσότερες εφαρμογές κρυπτογραφίας λειτουργούν κρυπτογραφώντας μεγάλους όγκους δεδομένων με της ίδιες παραμέτρους κρυπτογραφίας άρα δεν απαιτείτε συνεχώς η αλλαγή των παραμέτρων η οποία φυσικά επιβαρύνει την απόδοση του συστήματος. Στο σχήμα 32 φαίνεται η διαφοροποίηση των δύο οικογενειών συναρτήσεων που υλοποιήσαμε για κάθε αλγόριθμο.

Συνάρτηση	Τύπος Επιστροφής	Ορίσματα
1. aes_full	128 bit Message	1. 128 bit Input Message 2. 128 bit Key 3. Mode
2. aes_fast	128 bit Message	1. 128 bit Input Message
3. des_full	64 bit Message	1. 64 bit Input Message 2. 64 bit Key 3. Mode
4. des_fast	64 bit Message	1. 64 bit Input Message
5. TripeDes_full	64 bit Message	1. 64 bit Input Message 2. 64 bit Key1 3. 64 bit Key2 4. 64 bit Key3 1. Mode
6. TripleDes_fast	64 bit Message	64 bit Input Message

Πίνακας 19: Συναρτήσεις Επεξεργαστή PPC για την υποστήριξη τών Αλγορίθμων Κρυπτογραφίας.

6.2 Υποστήριξη Δυναμικής Αναδιάταξης

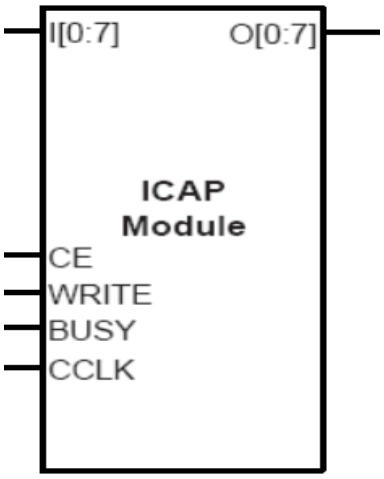
Στην υποενότητα αυτή παρουσιάζεται ο τρόπος με τον οποίο το σύστημα μας υποστηρίζει την Δυναμική Αναδιάταξη. Αρχικά παρουσιάζεται αναλυτικότερα η βαθμίδα ICAP που εκτελεί την Δυναμική Αναδιάταξη ενώ στην συνέχεια παρουσιάζεται ολόκληρη η διαδικασία καθώς και το λογισμικό που την εκτελεί.



Σχήμα 32: Λειτουργικότητα Συναρτήσεων Υποστήριξης Κρυπτογραφικών Συνεπεξεργαστών.

6.2.1 Internal Reconfiguration Access Port (ICAP)

Η βαθμίδα του ICAP [49] χρησιμοποιείται για την εκτέλεση της Δυναμικής Αναδιάταξης στην FPGA. Η βαθμίδα εκτελεί μόνο μερική αναδιάταξη και δεν δύναται να εκτελέσει αναδιάταξη ολόκληρης της συσκευής. Με τον ICAP μπορεί να γίνει είτε εγγραφή είτε ανάγνωση των δεδομένων διαμόρφωσης της FPGA. Οι θύρες εξόδου και εισόδου έχουν πλάτος ενός byte ενώ η μέγιστη συχνότητα ρολογιού που μπορεί να λειτουργήσει χωρίς την χρήση του σήματος Busy είναι 66 MHz. Ο ICAP είναι συνδεμένος με μία Bram μεγέθους 16 Kbytes η οποία χρησιμοποιείται σαν προσωρινή μνήμη διαμόρφωσης. Από τα 16Kbyte μόνο τα 2,048 Bytes μπορούν να χρησιμοποιηθούν για προσωρινή θέση αποθήκευσης χωρίς να υπάρχει δυνατότητα μεταβολής του μεγέθους αυτού. Η τροποποίηση της μνήμης διαμόρφωσης σύμφωνα με τα δεδομένα ενός Partial Bitstream γίνεται σε επίπεδο frame. Το frame αποτελεί το ελάχιστο τμήμα της μνήμης διαμόρφωσης που μπορεί να αναγνωστεί και να τροποποιηθεί κάθε φορά. Αφού γίνει ανάγνωση ενός frame της μνήμης διαμόρφωσης της FPGA αυτό αποθηκεύεται στην Bram όπου τροποποιείται σύμφωνα πάντα με το Partial Bitstream και στην συνέχεια επανεγράφεται πίσω. Περισσότερες λεπτομέρειες δεν παρέχονται από την Xilinx ενώ η διεπαφή της βαθμίδας φαίνεται στο σχήμα 33.



Σχήμα 33: Διεπαφή Βαθμίδας ICAP

6.2.2 Διαδικασία Δυναμικής Αναδιάταξης και Υποστήριξη της από Λογισμικό

Για την πραγματοποίηση της Δυναμικής Αναδιάταξης απαιτείται η μεταφορά δεδομένων από τα Partial Bitstream, που βρίσκονται αποθηκευμένα στην εξωτερική Compact Flash, στην βαθμίδα του ICAP ώστε αυτός να τροποποιήσει αντίστοιχα την μνήμη διαμόρφωσης της FPGA. Η διαδικασία αυτή απαιτεί σταδιακά την:

1. Μεταφορά των δεδομένων του Partial Bitstream από την εξωτερική Compact Flash στην μνήμη Bram του Power PC.

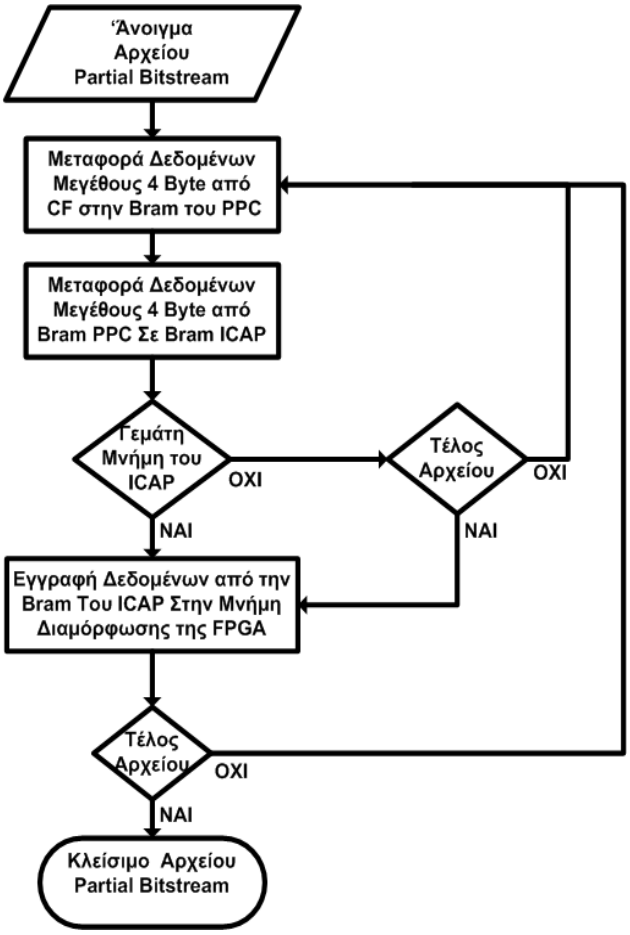
2. Μεταφορά από την μνήμη Bram του Power PC στην μνήμη Bram της βαθμίδας ICAP.

3. Εγγραφή δεδομένων της μνήμης του ICAP στην μνήμη διαμόρφωσης της FPGA μέσω της διαδικασίας Ανάγνωσης, Τροποποίησης, Εγγραφής.

Για την υλοποίηση της παραπάνω λειτουργικότητας χρησιμοποιήθηκαν οι συναρτήσεις που φαίνονται στον πίνακα 20 μαζί με την λειτουργικότητά τους. Στο διάγραμμα 34 παρουσιάζεται ολόκληρη η διαδικασία για την ροή των δεδομένων του partial bitstream στην μνήμη διαμόρφωσης της FGPA .

Βιβλιοθήκες	Συναρτήσεις	Περιγραφή Λειτουργικότητας
xsysace.h	sysace_fopen()	Άνοιγμα του Partial Bitstream
xsysace_l.h sysace_stdio.h	sysace_fread()	Μεταφορά δεδομένων απο το Partial Bitstream στην μνήμη Bram του PPC
	sysace_fclose()	Κλείσιμο του Partial Bitstream
xhwicap.h xhwicap_parse.h	XHwIcap_StorageBufferWrite()	Μεταφορά Δεδομένων στήν Bram του ICAP
xhwicap_i.h	XHwIcap_DeviceWrite()	Εγγραφή Δεδομένων στην Μνήμη Διαμόρφωσης τής FPGA

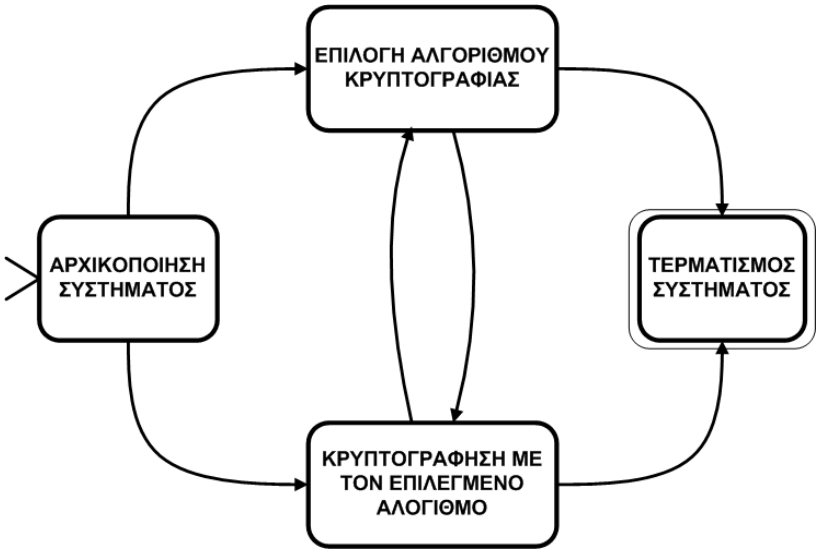
Πίνακας 20: Συναρτήσεις Επεξεργαστή PPC για την υποστήριξη Δυναμικής Αναδιάταξης.



Σχήμα 34: Διάγραμμα Ροής της Διαδικασίας Δυναμικής Αναδιάταξης

6.2.3 Λογισμικό Διαχείρισης Συστήματος

Στις προηγούμενες ενότητες έχουν ήδη περιγραφεί ο τρόπος που υποστηρίζεται με λογισμικό τόσο οι πυρήνες κρυπτογραφίας όσο και η διαδικασία της Δυναμικής Αναδιάταξης. Το σύστημα όπως έχουμε αναφέρει αρχικοποιείται κατά την έναρξη τροφοδοσίας της πλατφόρμας από ένα στατικό bitstream πού βρίσκεται αποθηκευμένο στην Compact Flash. Ανά πάσα στιγμή το σύστημα μπορεί να εκτελέσει κρυπτογράφηση με το αλγόριθμο που είναι υλοποιημένος εκείνη την χρονική στιγμή είτε να αναδιατάξει μερικώς την δυναμικά αναδιατασσόμενη περιοχή υλοποιώντας διαφορετικό αλγόριθμο. Η επιλογή του αλγορίθμου κρυπτογραφίας που υλοποιείται κατά την εκκίνηση του συστήματος έχει γίνει με κριτήρια που περιγράφονται στην επόμενη ενότητα. Στο σχήμα 35 παρουσιάζεται ένα γενικό πλάνο της λειτουργικότητας που προσφέρει στο σύστημα το λογισμικό που εκτελείτε στον PPC .



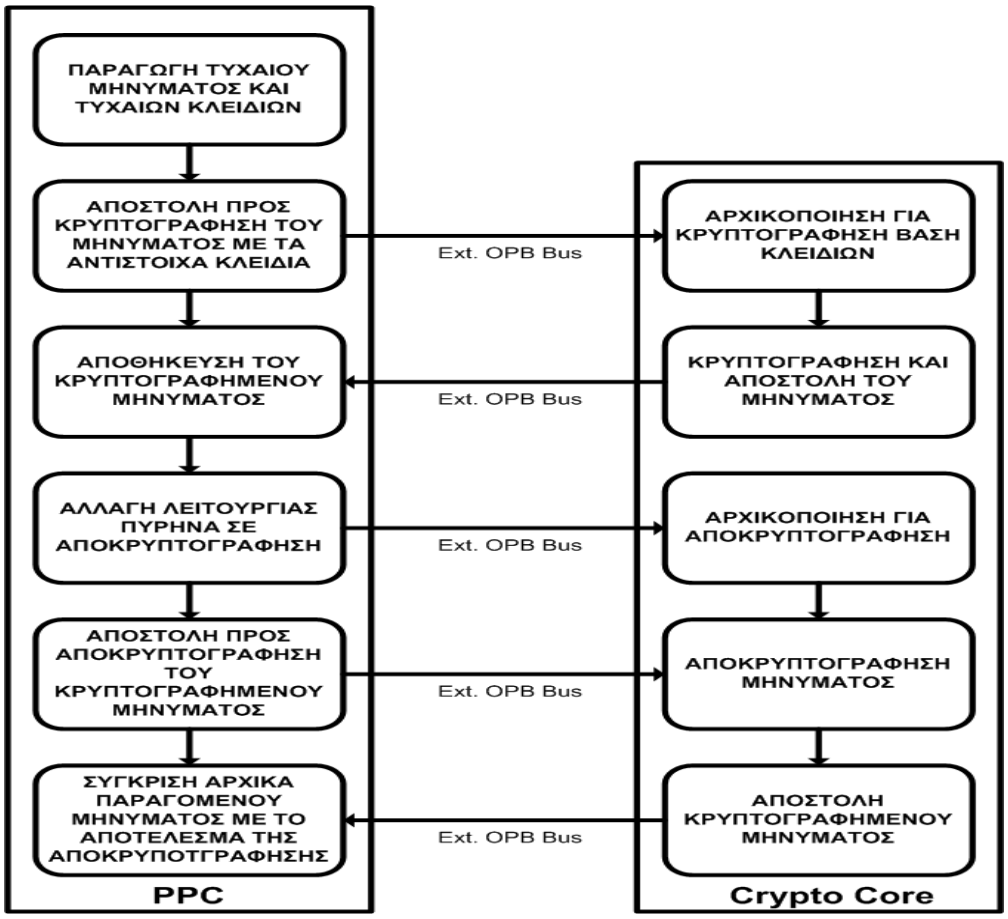
Σχήμα 35: Γενική Περιγραφή Λογισμικού Διαχείρησης Συστήματος

6.3 Επιβεβαίωση Λειτουργίας

Θέλοντας να εξασφαλίσουμε την ορθότητα λειτουργίας τόσο ολόκληρου του συστήματος όσο και των αλγορίθμων κρυπτογραφίας χρησιμοποιήσαμε τρείς ξεχωριστές μεθόδους επαλήθευσης που περιγράφονται παρακάτω.

6.3.1 Εσωτερική Επαλήθευση Λειτουργίας

Ολόκληρη η διαδικασία εκτελείται εσωτερικά του PPC. Για την επαλήθευση της ορθότητας λειτουργίας πραγματοποιείται δημιουργία από τον επεξεργαστή ψευδοτυχαίων μηνυμάτων προς κρυπτογράφηση και ψευδοτυχαίων κλειδιών. Τα μηνύματα αποστέλλονται στο υλοποιημένο πυρήνα κρυπτογραφίας την δεδομένη χρονική στιγμή ο οποίος επιστρέφει το κρυπτογραφημένο πλέον μήνυμα. Ο επεξεργαστής αφού αποθηκεύσει προσωρινά τα κρυπτογραφημένο μήνυμα αλλάζει την λειτουργία του πυρήνα από κρυπτογράφηση σε αποκρυπτογράφηση και αποστέλλει το προσωρινά αποθηκευμένο μήνυμα χωρίς να προβεί σε αλλαγή των κλειδιών κρυπτογραφίας. Το λαμβανόμενο αποκρυπτογραφημένο μήνυμα συγκρίνεται με το αρχικό και αν είναι όμοια η διαδικασία συνεχίζεται ειδάλλως ο χρήστης ενημερώνεται για την ύπαρξη λάθους. Όταν εκτελεστεί ένας προκαθορισμένος από τον χρήστη αριθμός επαναλήψεων τότε το σύστημα αναδιατάσσεται μερικώς αλλάζοντας τον υλοποιημένο πυρήνα κρυπτογραφίας και επαναλαμβάνει την ίδια διαδικασία. Το κύριο σκέλος της παραπάνω διαδικασία παρουσιάζεται στο σχήμα 36.



Σχήμα 36: Εσωτερική Επιβεβαίωση Λειτουργίας Συστήματος

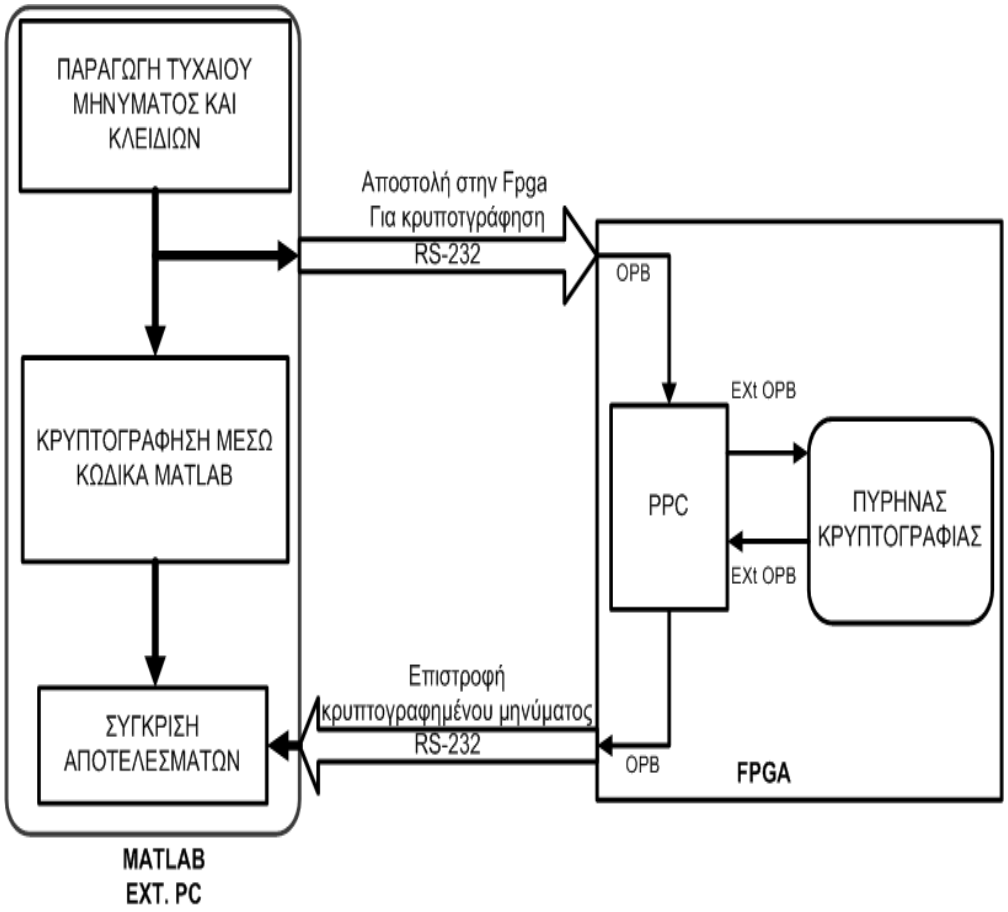
Η παραπάνω διαδικασία χρησιμοποιήθηκε και στην δημιουργία μια συνάρτησης αυτοελέγχου. Η συνάρτηση με το όνομα selftest() εκτελείται μετά την ολοκλήρωση δυναμικής αναδιάταξης ώστε να επιβεβαιώσει ότι η αλλαγή στην βαθμίδα κρυπτογραφίας έγινε επιτυχώς. Στον χρήστη παρέχεται η δυνατότητα να απενεργοποιήσει τον ενσωματωμένο αυτοέλεγχο κατά την διάρκεια λειτουργίας του συστήματος όπως και να καθορίσει τον αριθμό των μηνυμάτων που θα χρησιμοποιούνται για έλεγχο.

6.3.2 Εξωτερική Επιβεβαίωση Αποτελεσμάτων

Εκτός από την εσωτερική επιβεβαίωση αναπτύχθηκε και μια διαδικασία εξωτερικής επιβεβαίωσης ώστε να επαληθευτεί η ορθότητα της λειτουργίας ολόκληρου του συστήματος συμπεριλαμβανομένων των πυρήνων κρυπτογραφίας του επεξεργαστή και του διαύλου επικοινωνίας σειριακής θύρας. Καταρχήν υλοποιήθηκε σε περιβάλλον Matlab ένα πρωτόκολλο επικοινωνίας με το σύστημα υπερκαλύπτοντας ενσωματωμένες συναρτήσεις της Matlab για την επικοινωνία με σειριακή θύρα. Η διεπαφή του πρωτοκόλλου επικοινωνίας καθώς και ο τρόπος λειτουργίας του φαίνεται στον κώδικα που ακολουθεί:

```
%--------------------------------------------------------------
% Initialize I/O (RS232)
delete(instrfindall);
%--------------------------------------------------------------
MAX_NUM = 32;                %BYTE OF HANDSHAKING
TIMEOUT = 5;                 %TIMEOUT OF PPC_RESPONSE
BRAUDRATE = 115200;
s = serial('com1','BaudRate',BRAUDRATE,'InputBufferSize',MAX_NUM,...
          'OutputBufferSize',MAX_NUM,'Timeout',TIMEOUT);
fopen(s);
%--------------------------------------------------------------
profile on
%------------------SEND PLAINTEXT-------------------------------
% variable plaintext contains the message that will be encrypted.
fwrite(s, plaintext);
%--------------------------------------------------------------
%----------------RECIEVE CIPHERTEXT----------------------------
data_in = fread(s);
%--------------------------------------------------------------
```

Η διαδικασία περιλαμβάνει την δημιουργία τυχαίων δεδομένων στην Matlab και την ταυτόχρονη κρυπτογράφηση τους, τόσο στο PC εκτελώντας τον αντίστοιχο κώδικα Matlab για το δεδομένο κρυπτογραφικό αλγόριθμο, όσο και στο σύστημα κρυπτογραφίας με αποστολή των δεδομένων και λήψη του αποτελέσματος από την σειριακή θύρα. Η σύγκριση του τελικών αποτελεσμάτων των δύο οδών κρυπτογράφησης οδηγεί σε συμπέρασμα για την ορθότητα λειτουργίας του συστήματος. Η παραπάνω διαδικασία εκτελείτε για συγκεκριμένο αριθμό επαναλήψεων έως ότου δοθεί εντολή μέσω της Matlab να υλοποιηθεί διαφορετικός αλγόριθμος από το σύστημα. Αφού ολοκληρωθεί η δυναμική αναδιάταξη η διαδικασία επαναλαμβάνεται για τον υλοποιημένο αλγόριθμο χρησιμοποιώντας βέβαια την αντίστοιχη συνάρτηση της Matlab όπως παρουσιάζεται στο σχήμα 37.



Σχήμα 37: Εξωτερική Επιβεβαίωση Λειτουργίας Συστήματος

6.3.3 Επαλήθευση με χρήση Διανυσμάτων Δοκιμών

Εκτός από τις διαδικασίες που αναφέρθηκαν προηγουμένως και βασίζονται στην κρυπτογράφηση και αποκρυπτογράφηση τυχαίων μηνυμάτων και σύγκριση των αποτελεσμάτων μεταξύ τους ακολουθήθηκε και μια τρίτη διαδικασία επαλήθευσης της ορθότητας του συστήματος. Η διαδικασία βασίζεται σε δημοσιεύσεις του Διεθνή Ινστιτούτου Προτύπων και Τεχνολογίας [50] [51]. Στις δημοσιεύσεις αυτές υπάρχουν διανύσματα ελέγχου για τους αλγόριθμους DES και AES αντίστοιχα. Τα διανύσματα αυτά περιλαμβάνουν τριάδες δεδομένων της μορφής Μήνυμα, Κλειδί, Κρυπτογραφημένο μήνυμα, όπου το κρυπτογραφημένο μήνυμα είναι το αποτελέσματα της ορθής κρυπτογράφησης του μηνύματος με το αντίστοιχο κλειδί. Το σύστημα δοκιμάστηκε για όλα τα παραπάνω διανύσματα δοκιμών.

6.3.4 Επιδιόρθωση Προβλημάτων και Αποκατάσταση της Ορθότητας Λειτουργίας

Κατά την επαλήθευση του συστήματος ενώ οι αλγόριθμοι DES και Triple Des πέρασαν επιτυχώς και τις τρείς διαδικασίες επαλήθευσης ο αλγόριθμος AES παρουσίαζε συνεχώς εσφαλμένη διαδικασία Κρυπτογράφησης/Αποκρυπτογράφησης. Η αναζήτηση της πηγής του προβλήματος οδήγησε στην υλοποίηση του συστήματος στατικά χωρίς την υποστήριξη δυναμικής αναδιάταξης με υλοποιημένο αλγόριθμο κρυπτογραφίας τον AES. Η ορθότητα λειτουργίας του συστήματος επαληθεύτηκε για την συχνότητα λειτουργίας που χρησιμοποιήθηκε και στα δυναμικά αναδιατασσόμενο σύστημα μας. Η επιβεβαίωση του γεγονότος ότι το πρόβλημα δεν οφείλοταν σε λάθος προγραμματισμού της μνήμης διαμόρφωσης της Fpga κατά την δυναμική αναδιάταξη αλλά σε ζητήματα διανομής ρολογιού έγινε όταν το δυναμικά αναδιατασσόμενο σύστημα λειτούργησε ορθά για συχνότητες ρολογιού δέκα φορές μικρότερες από την προβλεπόμενη συχνότητα λειτουργίας. Μια απλή τοποθέτηση μετρητών ελέγχου σε διαφορές βαθμίδες του αλγορίθμου AES και την χρήση του προγράμματος Chipscope αλλά και λογικού αναλυτή, έδειξε ότι το πρόβλημα οφείλονταν σε λανθασμένο συγχρονισμό των διάφορων βαθμίδων του αλγορίθμου που προέρχονταν από το φαινόμενο της καθυστέρηση ρολογιού (Clock Skew). Μια ανάλυση του συστήματος με αποτυπωμένο τον αλγόριθμο AES εντός της δυναμικά αναδιατασσόμενης περιοχής φαίνεται στο σχήμα 38. Παρατηρούμε ότι ο αλγόριθμος AES χρησιμοποιεί και τις δέκα περιοχές ρολογιού της FPGA. Για την σύγκλιση των συχνοτήτων ρολογιού λειτουργίας του αλγόριθμου AES εντός του συστήματος μας, με τις προσδοκώμενες τιμές βάση των αποτελεσμάτων της σύνθεσης

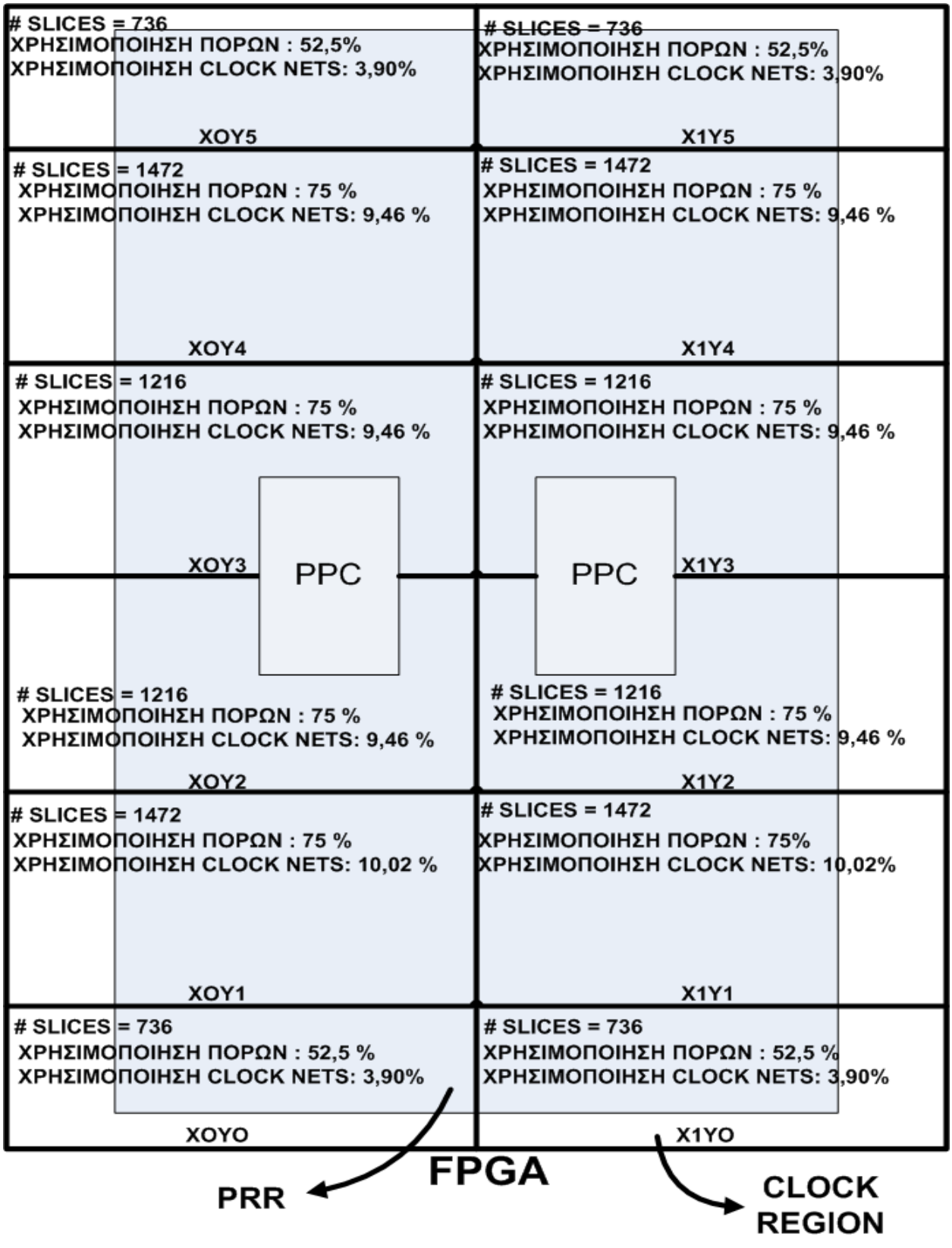
και της υλοποίησης του πυρήνα στο ISE πραγματοποιήσαμε τα εξής:

- Τοποθέτηση των DCM σε διαθέσιμες θέσεις ώστε να είναι πλησιεστέρα στην δυναμικά αναδιατασσόμενη περιοχή (σχήμα 25).
- Χρησιμοποίηση Clock Buffers για την οδήγηση των σημάτων ρολογιού των αλγορίθμου AES και DES/Triple-DES (σχήμα 25).
- Δήλωση στο αρχείο περιορισμών χρήστη (UCF) των συχνοτήτων που απαιτούμε να έχουν τα σήματα στα οποία έχουμε αναθέσει τον διαμοιρασμό του ρολογιού των τριών κρυπτογραφικών μας αλγορίθμων.
- Επιλογή High Optimization Effort στις επιλογές σύνθεσης του εργαλείου ISE .
- Δήλωση σαν προεπιλεγμένο αλγόριθμο κατά την εκκίνηση του συστήματος τον αλγόριθμο AES .

Και οι πέντε αλλαγές οδήγησαν σε βελτίωση του timing score . Στοιχεία της βελτίωσης παρουσιάζονται στον πίνακα 21. Με τις παραπάνω αλλαγές ο δυναμικά αναδιατασσόμενος πυρήνας κρυπτογραφίας AES λειτούργησε ορθά στην συχνότητα λειτουργίας που αναμενόταν.

ΒΑΘΜΙΔΑ	TIMING SCORE ΧΩΡΙΣ ΒΕΛΤΙΩΣΕΙΣ	TIMING SCORE ΜΕ ΒΕΛΤΙΩΣΕΙΣ	ΒΕΛΤΙΩΣΗ
Στατικό Σύστημα	11475	10640	1,08
Αλγόριθμος AES	9340006	7353340	1,56
Αλγόριθμος DES TDES	5971056	1086536	6,76

Πίνακας 21: Απόδοση των βελτιώσεων στην διανομή Ρολογιού στο Σύστημα.



Σχήμα 38: Κατανομή Αλγορίθμου AES στα Clock Region της FGPA

7 Πειραματικά Αποτελέσματα.

Η παρούσα ενότητα παρουσιάζει τα πειραματικά αποτελέσματα που εξήχθησαν κατά την λειτουργία του συστήματος .

7.1 Περιγραφή της Μεθοδολογίας Μετρήσεων

Για τις μετρήσεις χρόνων που περιγράφονται παρακάτω χρησιμοποιήθηκαν δύο προσεγγίσεις. Η πρώτη προσέγγιση περιλάμβανε την χρήση μετρητών λογισμικού. Χρησιμοποιήθηκε ο ειδικός καταχωρητής χρόνου του Power PC σε συνεργασία με την συνάρτηση XTime_GetTime(&time) που βρίσκεται στην βιβλιοθήκη " xtime.h ". Η δεύτερη προσέγγιση περιλάμβανε την χρήση μετρητών που έχουν υλοποιηθεί στο παράλληλο σύστημα όπως αυτοί περιγράφονται στο Κεφάλαιο 5. Τα αποτελέσματα ήταν συναφή και για τις δύο προσεγγίσεις που χρησιμοποιήσαμε με απόκλιση μικρότερη του 0,01% συνεπώς στις παρακάτω μετρήσεις εμφανίζεται μόνο μια τιμή και όχι δύο για κάθε μέτρηση.

7.2 Ανάλυση του Χρόνου για Δυναμική Αναδιάταξη

Η διαδικασία δυναμικής αναδιάταξης έχει ήδη περιγραφεί στο Κεφάλαιο 6. Για την πλήρη κατανόηση του χρόνου που απαιτείται για την ολοκλήρωση της όλης διαδικασίας θα ορίσουμε επιμέρους χρόνους, των οποίων το άθροισμα θα μας δώσει την συνολική καθυστέρηση για την εναλλαγή ενός πυρήνα κρυπτογραφίας :

- $t_{CFtoPPC}$: Ο συνολικός χρόνος που απαιτείται για την μεταφορά των δεδομένων διαμόρφωσης από την Compact Flash στην μνήμη του επεξεργαστή.
- $t_{PPCtoICAP-BRAM}$: Ο συνολικός χρόνος που απαιτείται για την μεταφορά των δεδομένων διαμόρφωσης από την μνήμη του επεξεργαστή στην μνήμη του ICAP.
- $t_{ICAP-WRITE}$: Ο χρόνος για τον προγραμματισμό των δεδομένων διαμόρφωσης από την προσωρινή μνήμη του ICAP στην μνήμη διαμόρφωσης της FPGA.
- t_{REST}: Ο χρόνος που απαιτείται για επιμέρους διαδικασίες εντός του PPC όπως άνοιγμα και κλείσιμο αρχείων, έλεγχοι δεδομένων κατά τις μεταφορές, έλεγχος συνθηκών τερματισμού διαδικασιών .

- t_{Total}: Ο συνολικός χρόνος από την εκκίνηση της διαδικασίας έως την στιγμή που ο νέος πυρήνας κρυπτογραφίας είναι έτοιμος για λειτουργία. Είναι προφανές ότι ο συνολικός χρόνος της αναδιάταξης των πυρήνων κρυπτογραφίας θα προκύπτει από τον παρακάτω τύπο

$$t_{Total} = t_{CFtoPPC} + t_{PPCtoICAP-BRAM} + t_{ICAP-WRITE} + t_{REST} \quad (4)$$

Στον πίνακα 22 παρουσιάζονται ποσοτικά στοιχεία για τα Partial Bitstream καθώς και το μέγεθος τους. Στον πίνακα 24 παρουσιάζονται οι συνολικοί χρόνοι για την δυναμική αναδιάταξη καθώς και οι επιμέρους χρόνοι για κάθε διαδικασία.. Η κατανομή του χρόνου στις επιμέρους διαδικασίες παρουσιάζεται τόσο στον πίνακα 23 όσο και στο γράφημα 39. Είναι φανερό ότι η μεταφορά των δεδομένων από την CF προκαλεί την μεγαλύτερη καθυστέρηση στην ολοκλήρωση της διαδικασίας. Η αποθήκευση των partial bitstream σε ένα γρηγορότερο μέσο θα επέφερε δραματική μείωση του χρόνου της όλης διαδικασίας.

Partial Bitstream	Χρήσιμοι Πόροι Slices	% Χρήση	Μέγεθος Bytes
Aes.bit	9671 από 9920	97,46 %	749737
Des-TripleDES.bit	1846 από 9920	18,61 %	744037
Blank.bit	9920 απο 9920	100 %	673895

Πίνακας 22: Ποσοτικά στοιχεία και μέγεθος των *PartialBitstream*.

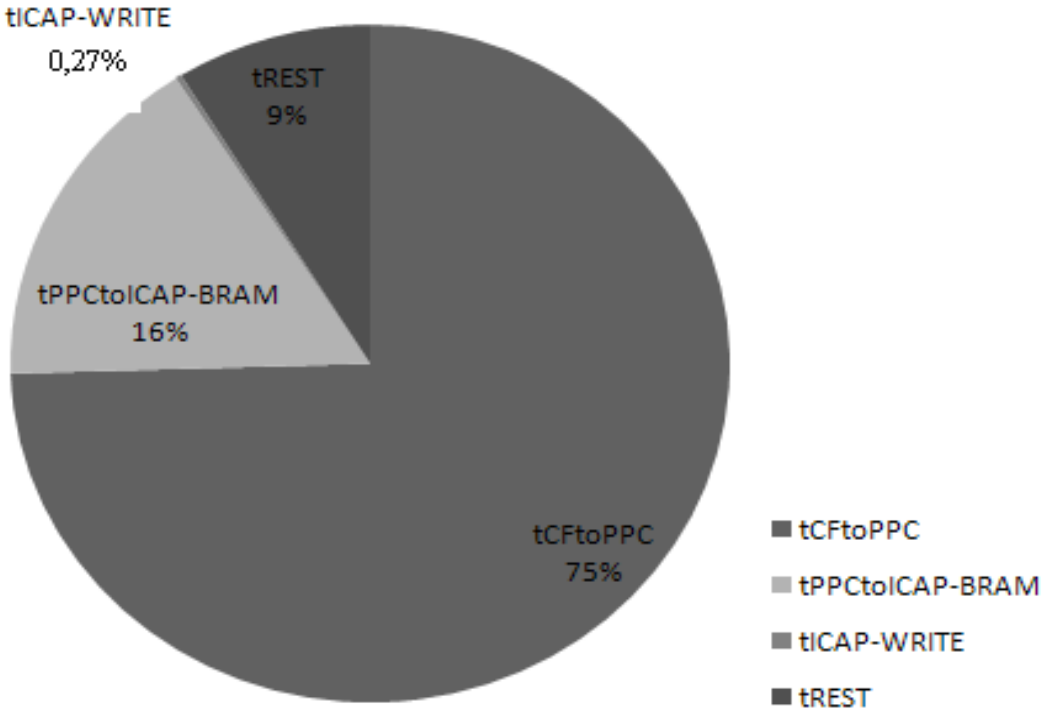
Partial Bitstream	$t_{CFtoPPC}$ (sec)	$t_{PPCtoICAP-BRAM}$ (sec)	$t_{ICAP-WRITE}$ (sec)	t_{REST} (sec)	t_{Total} (sec)
Aes.bit	2,78204320	0,61	0,01012	0,33000099	3,73216419
Des-TripleDES.bit	2,720074641	0,596412569	0,00909628	0,32336797	3,649749762
Blank.bit	2,500616886	0,548293535	0,00909628	0,301183984	3,359190684

Πίνακας 23: Χρόνοι Δυναμικής Αναδιάταξης.

Παρατηρούμε ότι ενώ το μέγεθος της δυναμικά αναδιατασσόμενης περιοχής δεν αλλάζει και κάθε φορά προγραμματίζεται η ίδια περιοχή το μέγεθος των Partial Bitstream είναι διαφορετικό. Το γεγονός αυτός οφείλεται στα εργαλεία που παράγουν τα Partial Bitstream. Βάση

Partial Bitstream	$t_{CFtoPPC}$	$t_{PPCtoICAP-BRAM}$	$t_{ICAP-WRITE}$	t_{REST}
Aes.bit	74,5422 %	16,3444 %	0.2711 %	8,8433 %
Des-TripleDES.bit	74,5277 %	16,3411 %	0,2711 %	8,8611 %
Blank.bit	74,4410 %	16,3221 %	0.2707 %	8,9659 %

Πίνακας 24: Καταμερισμός των επιμέρους χρόνων της διαδικασίας της Δυναμικής Αναδιάταξης.



Σχήμα 39: Γράφημα Παρουσίασης Επιμέρους Χρόνων για τον Αλγόριθμο *Aes*.

αυτού, εκτιμούμε ότι σε μερικά σημεία εκτελείτε Diffence Based προσέγγιση χωρίς όμως να μπορούμε να το επιβεβαιώσουμε δεδομένης της απουσίας επαρκούς τεκμηρίωσης από την Xilinx. Οι διαφορετικοί χρόνοι λοιπόν που παρουσιάζονται ανάλογα με την βαθμίδα που προγραμματίζονται, οφείλονται κυρίως στην διαφορά του μεγέθους των Partial Bitstream. Στον πίνακα 25 εμφανίζονται ο μέσος όρος χρήσιμων ποσοτικών μεγεθών που προέκυψαν από τα πειραματικά αποτελέσματα, καθώς και μία σύγκριση με σχετική έρευνα σε Δυναμικά Αναδιατασσόμενο σύστημα Κρυπτογραφίας [22]. Τα παρακάτω μπορούν να χρησιμοποιηθούν για μια εκτίμηση του χρόνου μερικής αναδιάταξης μελλοντικών σχεδιάσεων. Ο χρόνος μερικής αναδιάταξης ανά ColumnSlice αναφέρεται γιατί στην FPGA που χρησιμοποιήσαμε, όπως και στις υπόλοιπες της

ίδιας οικογένειας, για τον προγραμματισμό έστω και ενός Slice προγραμματίζεται ολόκληρη η στήλη στην οποία ανήκει. Η σχεδίαση μας αποτελείται από 72 ColumnSlices.

Μετρούμενη	Χρόνος Δυναμικής Αναδιάταξης	
Ποσότητα	Παρούσα Εργασία	Lagger [22]
Χρόνος ανά Slice του PRR	0,360924215 ms	0,526666667 ms
Χρόνος ανά Kbyte του Partial Bitstream	5,074729498 ms	4,966666667 ms
Χρόνος ανά ColumnSlice	49,72733628 ms	–

Πίνακας 25: Μέσος Όρος Ποσοτικών Μεγεθών για Χρόνους Δυναμικής Αναδιάταξης.

Γνωρίζουμε τόσο από την τεκμηρίωση της Xlinx όσο και από σχετικές δημοσιευμένες έρευνες [28] ότι η θεωρητική διεκπεραιωτική ικανότητα του ICAP είναι 0,75 Gbit/s ποσό που αφορά μόνο την εγγραφή στην μνήμη διαμόρφωσης της FPGA ενώ ο μέγιστη διεκπεραιωτική ικανότητα ολόκληρου του περιφερειακού OPB-ICAP είναι 32,4 Mbits/s. Επίσης ο μέγιστος ρυθμός διαμεταγωγής της Compact Flash που χρησιμοποιήσαμε ήταν 8 MB/s. Στον παρακάτω πίνακα 26 συγκρίνονται οι παραπάνω τιμές με αυτές που βρέθηκαν πειραματικά.

	Πειραματικά Αποτελέσματα	Μέγιστες Θεωρητικές Τιμές
ICAP (Write)	0,55 Gbit/s	$0,75 Gbit/s$
ICAP (Total)	9,22 Mbit/s	$32,4 Mbit/s$
CF	0,26 MB/s	$8 MB/s$

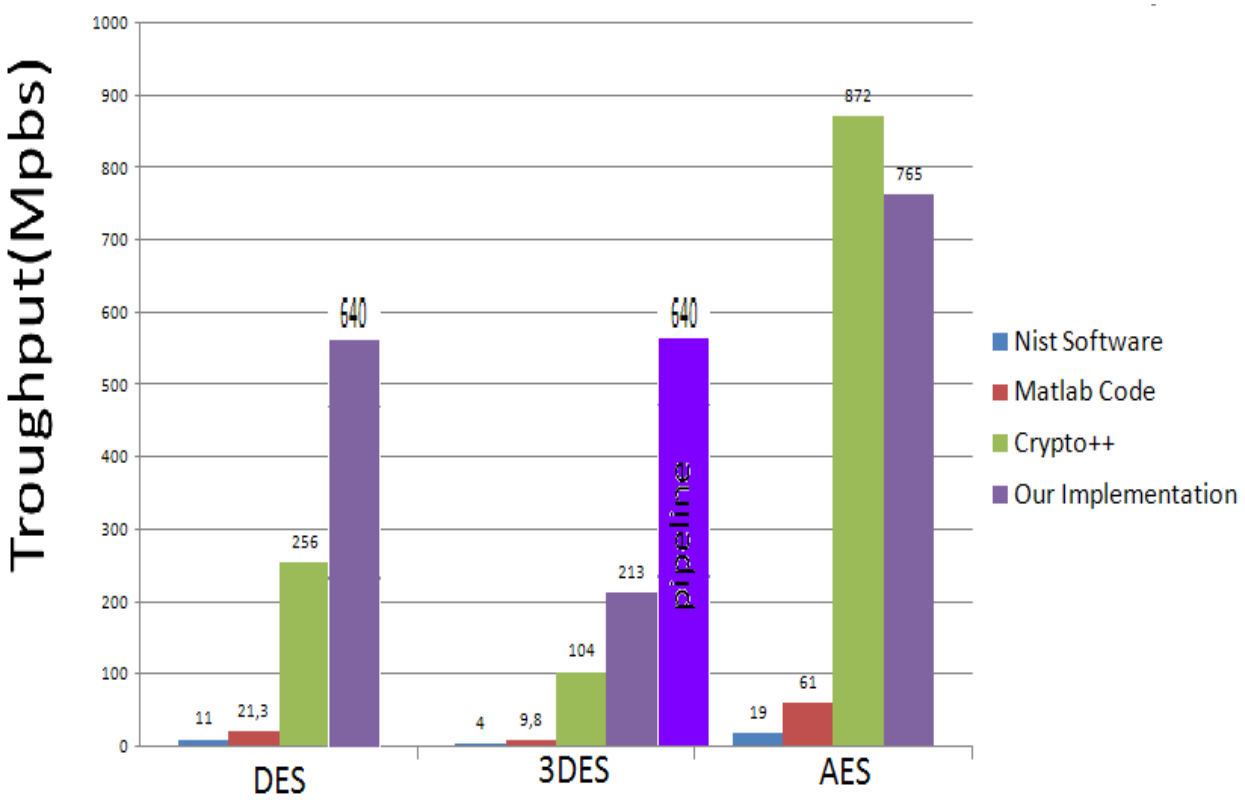
Πίνακας 26: Σύγκριση Πειραματικών και Θεωρητικών Αποτελεσμάτων.

7.3 Σύγκριση απόδοσης Κρυπτογραφικών Αλγορίθμων

Σκοπός της παρούσας εργασίας δεν ήταν η βελτιστοποίηση της απόδοσης των αλγορίθμων κρυπτογραφίας αλλά η επιβεβαιωμένη λειτουργία του συστήματος υποστήριξης πολλαπλών κρυπτογραφικών αλγορίθμων και η μελέτη της δυναμικής αναδιάταξης μέσω αυτού. Τα παρακάτω αποτελέσματα συγκρίνουν την απόδοση των πυρήνων κρυπτογραφίας υλοποιημένων σε υλικό του συστήματος μας, με την απόδοση τον ίδιων αλγορίθμων που εκτελούνται σε επεξεργαστές γενικού σκοπού. Μίας και η απόδοση των αλγορίθμων κρυπτογραφίας δεν αποτελεί αυτοσκοπό της εργασίας, η παρακάτω σύγκριση γίνεται ώστε να μπορέσει ο αναγνώστης να κατατοπιστεί σε σχέση με το μέγεθος της απόδοσης των υλοποιημένων πυρήνων κρυπτογραφίας. Η σύγκριση έγινε για εκτέλεση των αλγορίθμων υλοποιημένων σε Java σε ένα Intel Pentium Pro 200 MHz CPU που αποτελεί την πλατφόρμα αναφοράς του Διεθνούς Ινστιτούτου Προτύπων και Τεχνολογίας (NIST) για σύγκριση κρυπτογραφικών αλγορίθμων [52]. Επιπλέον γίνεται σύγκριση με την απόδοση των Κρυπτογραφικών αλγορίθμων για εκτέλεση σε ένα Intel Core 2 Duo 1,83 Mhz CPU σε περιβάλλον Matlab χρησιμοποιώντας τον κώδικα που αναπτύξαμε για την επαλήθευση του συστήματος, καθώς επίσης για κώδικα των αλγορίθμων σε γλώσσα C++ που περιέχονταν στην βιβλιοθήκη Crypto++. Στον πίνακα 27 και στο σχήμα 40 παρουσιάζονται τα παραπάνω καθώς και η απόδοση του pipeline Triple Des, ο οποίος όταν λειτουργεί με συμπληρωμένα με δεδομένα και τα τρία στάδια της διοχέτευσης και τροφοδοτείται συνεχώς με νέα δεδομένα έχει απόδοση όμοια με τον DES.

Αλγόριθμος	Throughput (Mbps)			
	Nist Software	Matlab Code	Crypto++	Our Impl.
DES	11	21,3	256	640
T-DES	4	9,8	104	213/640
AES	19	61	872	765

Πίνακας 27: Σύγκρίση Πυρήνων Κρυπτογραφίας υλοποιημένους σε Υλικό με εκτέλεση σε Λογισμικό.



Σχήμα 40: Γράφημα Σύγκρισης Υλοποίησης Αλγορίθμων Κρυπτογραφίας σε Υλικό και Λογισμικό.

8 Συμπεράσματα και Μελλοντική Εργασία.

Η ραγδαία αύξηση της χρήσης των ενσωματωμένων συστημάτων τα τελευταία χρόνια κάνει επιτακτική αλλά και εξαιρετικά ενδιαφέρουσα την χρήση τεχνολογιών που θα αυξήσουν την απόδοση και θα επεκτείνουν την λειτουργικότητα των συστημάτων αυτών. Στην παρούσα εργασία αναπτύξαμε ένα πλήρως αυτόνομο δυναμικά αναδιατασσόμενο σύστημα υποστήριξης πολλαπλών κρυπτογραφικών αλγορίθμων.

Τόσο η σχεδίαση όσο και ο τρόπος υλοποίησης περιγράφηκαν αναλυτικά, με ιδιαίτερη έμφαση στο τρόπο αντιμετώπισης προβλημάτων που προέκυψαν κυρίως από την έλλειψη επαρκούς τεκμηρίωσης από την Xilinx για την τεχνολογία αυτή. Το σύστημα αναπτύχθηκε με τέτοιο τρόπο ώστε να είναι εφικτή η μελλοντική του επέκταση ώστε να υποστηρίζει οποιοδήποτε κρυπτογραφικό αλγόριθμο με τους ελάχιστους δυνατούς περιορισμούς.

Το εύρος της παρούσας εργασίας δεν πρέπει να περιοριστεί μόνο στης κρυπτογραφικές εφαρμογές μιας και αυτές χρησιμοποιήθηκαν μόνο για την μελέτη της εφαρμογής της Δυναμικής Αναδιάταξης. Η επιβεβαιωμένη λειτουργία του συστήματος κάνει ενδιαφέρουσα την χρήση του συστήματος και σε άλλες εφαρμογές που θα ωφεληθούν από την επιπλέον λειτουργικότητα που προσφέρει. Εξάλλου η υψηλή πολυπλοκότητα και απαίτηση πόρων των αλγορίθμων που αναπτύξαμε στην παρούσα εργασία δείχνει ότι η εφαρμογή της Δυναμικής Αναδιάταξης είναι εφικτή σε μια μεγάλη γκάμα εφαρμογών. Τα ενθαρρυντικά αποτελέσματα που προέκυψαν από την εργασία αυτή δίνουν το έναυσμα για μελλοντικές βελτιώσεις που μπορούν να γίνουν στο σύστημα μας. Είναι προφανές ότι το ευάλωτο σημείο του συστήματος είναι το μεγάλο χρονικό κόστος κατά την πραγματοποίηση της Δυναμικής Αναδιάταξης, γεγονός που κάνει και δυσκολότερη την λειτουργία του σε συστήματα πραγματικού χρόνου. Βελτίωση του χρόνου αυτού μπορει να πραγματοποιηθεί με

1. Μεταφορά του χώρου αποθήκευσης των Partial Bitstream από την Compact Flash σε μια DDR Ram η οποία υποστηρίζεται και από την πλατφόρμα που χρησιμοποιήσαμε στην εργασία μας. Τα Partial Bitsream θα αποθηκεύονται στην Compact Flash και θα μεταφέρονται στη DDR Ram κατά την εκκίνηση του συστήματος.

2. Υλοποίηση DMA ελεγκτή για την μεταφορά των δεδομένων από την DDR Ram στον ICAP χωρίς να απαιτείται παρέμβαση του επεξεργαστή.

3. Υλοποίηση διεπαφής HWICAP για λειτουργία σε συχνότητες μεγαλύτερες των 66Mhz

που προσφέρονται από την Xilinx .

4. Διάσπαση των δυναμικά αναδιατασσόμενων περιοχών και χρήση προφόρτωσης για επικάλυψη του χρόνου που απαιτείται.

Βελτιώσεις μπορούν να γίνουν επίσης και στους κρυπτογραφικούς αλγόριθμους αλλά και στο τρόπο εισόδου εξόδου του συστήματός μας . Πολλαπλά οφέλη θα υπήρχαν από

1. Βελτίωση της υλοποίησης των αλγορίθμων με σκοπό την αύξηση της απόδοσης τους.

2. Υλοποίηση βιβλιοθήκης αλγορίθμων κρυπτογραφίας τους οποίους το σύστημα θα μπορεί να υποστηρίξει.

3. Βελτίωση της Εισόδου Εξόδου του συστήματος με αντικατάσταση της σειριακής θύρας που χρησιμοποιείται με ταχύτερες όπως Ethernet ή PCI Express .

4. Χρησιμοποίηση λειτουργικού συστήματος για την διαχείριση όλων των παραπάνω.

Αναφορές

[1] Xilinx : Early Access Reoconfiguration Lounge, EARP Section http://www.xilinx.com

[2] Lagger A.,Self-reconfigurable platform for cryptographic application , 2006.

[3] Stephen D. Brown, Robert J. Francis, Jonathan Rose, Zvonko G. Vranesic, Field Programmable Gate Arrays, Springer, 1992.

[4] Rose Jonathan, Abbas El Gamal, Alberto Sangiovanni-Vincentelli,"Architecture of Field Programmable Gate Arrays", Proceedings of the IEEE 1993.

[5] "Virtex-II Pro / Virtex-II Pro X Complete Data Sheet" , Xilinx, May 2007, http://www.xilinx.com/support/documentation/data_sheets/ds083.pdf .

[6] S. Raaijmakers , S. Wong,"Run-time Placement and Routing on the Virtex 2 Pro".

[7] Digital Clock Manager (DCM) Module Xilinx DS485 August 13,2004.

[8] M. Dworkin. Recommendation for Block Cipher Modes of Operation , Methods and Techniques. National Institute of Standards and Technology December 2001. csrc.nist.gov/encryption/tkmodes.html.

[9] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone. Handbook of Applied Cryptography. CRC Press, Boca Raton, Florida, USA, 1997.

[10] S. Kumar, T. Wollinger, Fundumentals of Symmetric Cryptography.

[11] DES Modes of Operation, FIPS, Federal Information Processing Standard, Pub No. 81. Available at csrc.nist.gov/fips/change81.ps, December 1980.

[12] American National Standards Institute. ANSI X9.52-1998, Triple Data Encryption Algorithm Modes of Operation, 1998.

[13] National Institute of Standards and Technology, US Department of Commerce. Federal Information Processing Standards FIPS PUB 46-3, Data Encryption Standard (DES), October 25, 1999. Available at csrc.nist.gov/CryptoToolkit/tkencryption.html.

[14] NIST FIPS PUB 46-3. Data Encryption Standard. Federal Information Processing Standards, National Bureau of Standards, US Department of Commerce,1977.

[15] B. Schneier. Applied Cryptography. John Wiley and Sons Inc., New York, USA.

[16] Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, NIST , 2008.

[17] Modes of Operation Validation System for Triple Data Encryption Algorithm, NIST Special Publication 800-20.

[18] American National Standar Institute ANSI X9.52, Triple Data Encryption Algorithm Mode of Operation 1998.

[19] Daemen and V. Rijmen. The design of Rijndael: AES: the Advanced Encryption Standard. Springer 2002

[20] NIST FIPS PUB 46-3. Advanced Encryption Standard. Federal Information Processing Standards, National Institute of Standards and Technology,2001.

[21] C. Effraimidis, A. Dollas, A Self Reconfigurable Architecture To Support Multiple Fitness Functions In Genetic Algorithm, May 2009.

[22] A. Lagger, Self-Reconfigurable Platform for Cryptographic Application, June 2006.

[23] E. Custodio, B. Marsland, Self-Healing Partial Reconfiguration of an FPGA, April 2007.

[24] A. Anyfantis, K. Papademetriou, A. Dollas, Performance analysis of dynamic reconfiguration in modern integrated logic circuits.

[25] A. Anyfantis, K. Papademetriou, A. Dollas, Methology and Experimental Setup for the Determination of System-level Dynamic Reconfiguration Overhead.

[26] Early Acess Reconfiguration Lounge , Xilinx.

[27] Recops Project, Reconfigurable Programmabel Devices for Military Hardware.

[28] Recops Project, An Evaluation of Dynaminc Partial Reconfiguration for Signal and Image Processing in Professional Electronics Applications

[29] K. Papademetriou, A. Dollas, Performance Evaluation of a Preloading Model in Dynamically Reconfigurable Processors.

[30] K. Papademetriou, A. Anyfantis, A. Dollas An Effective Framework to Evaluate Dynamic Partial Reconfiguration in FPGA Systems.

[31] C. Claus, F. Muller, J. Zepenfeld, W. Stechele, A new framework to accelerate Virtex-II Pro dynamic Partial self-reconfiguration.

[32] P. Sedcole,B. Blodget, T. Becker, Modular Partial Reconfiguration in Virtex Fpga.

[33] I. Gonzalez, S. Lopez-Buedo, F. Gomez, J. Martinez Using Partial Reconfiguration in Cryptographic Application : An Implementation of the IDEA Agorithm.

[34] I. Gonzalez, S.Lopez-Buedo, F. Gomez, Implementation of secure applications in self-reconfigurable systems.

[35] J.M Granado et al., IDEA and AES, two cryptographic algorithms implemented using partial and dynamic reconfiguration. Microelectron J(2009).

[36] F.X. Standaert, G. Rouvroy, J.J. Quisquater, J.D. Legat, Efficient implementation of Rijndael encryption in reconfigurable hardware: improvements and design tradeoffs.

[37] J. Zambreno, D. Nguyen, A. Choudhary, Exploring area/delay tradeoffs in an AES FPGA implementation, in: 14th Field Programmable Logic and Applications.

[38] X. Zhang and K.K. Parhi, High-speed VLSI architectures for the AES algorithm, IEEE Transactions on Very Large Scale Integration (VLSI) Systems.

[39] K. Gaj, P. Chodowiec, Fast implementation and fair comparison of the final candidates for advanced encryption standard using field programmable gate arrays.

[40] Patterson, C.: High Performance DES Encryption in Virtex FPGAs using Jbits. In: Field-programmable custom computing machines.

[41] McLoone, M., McCanny, J.: High-performance FPGA implementation of DES using a novel method for implementing the key schedule.

[42] N. Saqib, F. Rodriguez-Honriquez, A. Diaz-Perez, A Compact and Efficient FPGA Implementation of DES Algorithm.

[43] Kaps, J., Paar, C.: Fast DES implementations for FPGAs and its application to a Universal key-search machine.

[44] Wong, K., Wark, M., Dawson, E.: A Single-Chip FPGA Implementation of the Data Encryption Standard (des) Algorithm.

[45] Xilinx ISE 9.1i Software Manuals and Help, www.xilinx.com/support/sw-manuals/xilinx9/index.htm

[46] EDK 9.1i Software Documentation , www.xilinx.com/support/documentation/dt-edk-edk9-1.htm

[47] PlanAhead 10.1 Release Notes , www.xilinx.com/ise/optional-prod/PlanAhead-Release-Notes-10.1.pdf

[48] VHDL Code foe AES , DES , TRIPLE DES , www.opencores.com

[49] Xilinx "Internal Configuration Accesss Port Module Virtex2 Documentation", http://toolbox.xilinx.com/docsan/xilinx6/books/data/docs/lib/lib0233_201.html

[50] NIST Special Publication 5002O, Validating the Correctness of Hardware Implementation of NBS Data Encryption Standar.

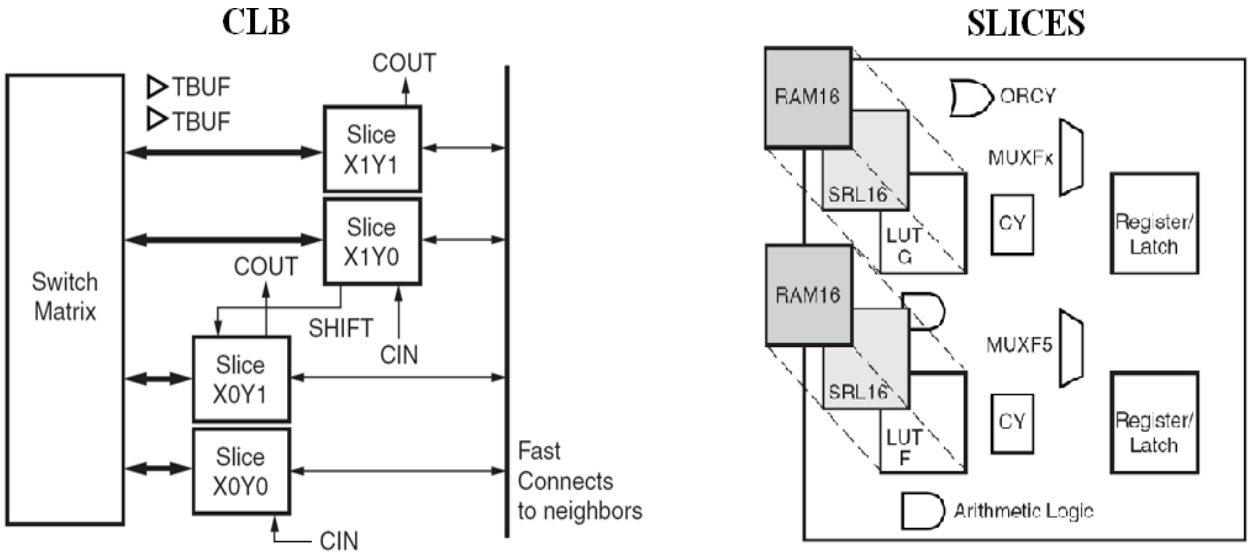
[51] NIST, The Advanced Encryption Standar Algorithm Validation Suite (AESAVS), Nov 15 2002.

[52] A.Sterbenz, P. Lipp, Performance of the AES Candidate Algorithms in Java.

Παράρτημα Α.

Στο παράρτημα αυτό παρουσιάζονται τεχνικές λεπτομέρειες και αναλυτικά στοιχεία σχετικά με την Xilinx Virtex-II Pro XC2VP30 Fpga που χρησιμοποιήθηκε στην εργασία.
Οι παρακάτω βαθμίδες απαιτούν ιδιαίτερη μεταχείριση κατα την εφαρμογή της δυναμικής αναδιάταξης και για τον λόγο αυτό παρουσιάζονται αναλυτικότερα :

- Τα CLB είναι η βασική μονάδα που υλοποιεί την ακολουθιακή και συνδιαστική λογική, καθώς και κάποια βασικά στοιχεία μνήμης. Κάθε CLB αποτελείται από τέσσερα Slices και απαιτεί 128 bit πληροφορίας της μνήμης διαμόρφωσης (Configuration Memory) για να υλοποιήσει την λειτουργία του (σχήμα 41).



Σχήμα 41: Δομή και διασύνδεση CLB και SLICES σε μία Virtex-II Pro Fpga

- Τα DCM είναι η βαθμίδα η οποία επιτρέπει την διαχείρηση των ρολογιών των σχεδιάσεων που υλοποιούνται στην FPGA. Προσφέρουν την δυνατότητα πολλαπλασιασμού η διαίρεσης της συχνότητας ενός ρολογιού αναφοράς, ολίσθηση φάσης, εισαγωγή καθυστέρησης, κλείδωμα φάσης [7].

- Παρόλο που τα CLB μπορούν να υλοποιήσουν τόσο στοιχεία μνήμης όσο και πολλαπλασιαστές, για αύξηση τόσο της απόδοσης όσο και της διαθέσιμης μνήμης μέσα στην FPGA υπάρχουν ήδη προτοποθετημένες μνήμες Ram διπλής θύρας μεγέθους 18 Kbit η κάθε μια (BRAM), καθώς και πρωτοποθετημένους πολλαπλασιαστές που εκτελούν έως και 18 bit x 18 bit πολλαπλασιασμούς.

Στο σχήμα 42 υπάρχει μία σύγκριση της FPGA που χρησιμοποιήσαμε με τα υπόλοιπα μέλη της οικογενείας Virtex II Pro .

Device[1]	RocketIO Transceiver Blocks	PowerPC Processor Blocks	Logic Cells[2]	CLB (1 = 4 slices = max 128 bits)		18 X 18 Bit Multiplier Blocks	Block SelectRAM+		DCMs	Maximum User I/O Pads
				Slices	Max Distr RAM (Kb)		18 Kb Blocks	Max Block RAM (Kb)		
XC2VP2	4	0	3,168	1,408	44	12	12	216	4	204
XC2VP4	4	1	6,768	3,008	94	28	28	504	4	348
XC2VP7	8	1	11,088	4,928	154	44	44	792	4	396
XC2VP20	8	2	20,880	9,280	290	88	88	1,584	8	564
XC2VPX20	8[4]	1	22,032	9,792	306	88	88	1,584	8	552
XC2VP30	8	2	30,816	13,696	428	136	136	2,448	8	644
XC2VP40	0[3], 8, or 12	2	43,632	19,392	606	192	192	3,456	8	804
XC2VP50	0[3] or 16	2	53,136	23,616	738	232	232	4,176	8	852
XC2VP70	16 or 20	2	74,448	33,088	1,034	328	328	5,904	8	996
XC2VPX70	20[4]	2	74,448	33,088	1,034	308	308	5,544	8	992
XC2VP100	0[3] or 20	2	99,216	44,096	1,378	444	444	7,992	12	1,164

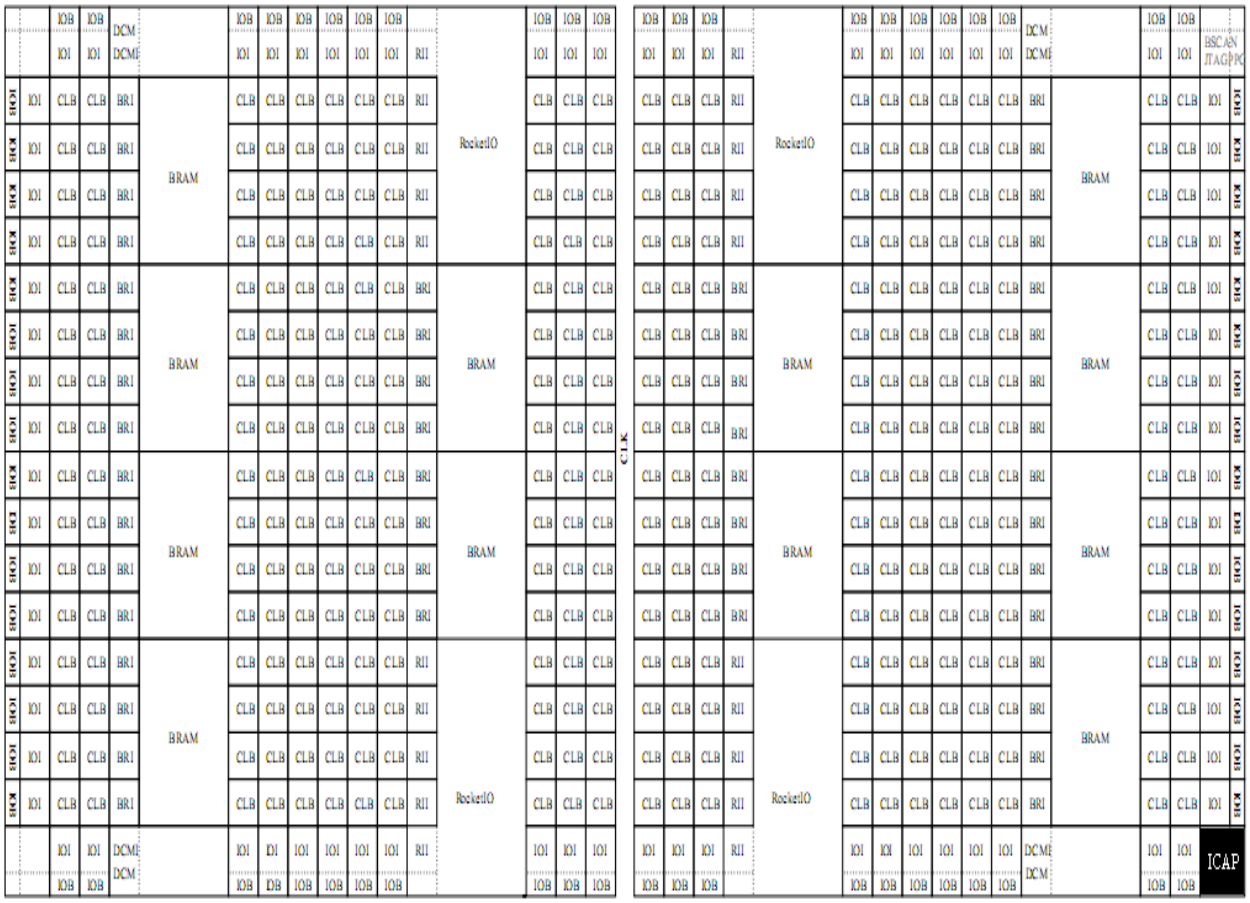
Σχήμα 42: Πόροι μελών οικογένειας Virtex-II Pro Fpga

Στο σχήμα 43 παρουσιάζεται ο αριθμός των στηλών που καταλαμβάνει κάθε βαθμίδα καθώς και ο αριθμός των frames που χρειάζεται για να προγραμματιστούν.

Column Type / Device	IOB		IOI		CLB		BRAM		BRAM Interconnect		GCLK	
	Columns per Device:	Frames per Column:	Columns per Device:	Frames per Column:	Columns per Device:	Frames per Column:	Columns per Device:	Frames: per Column	Columns per Device:	Frames per Column:	Columns per Device:	Frames per Column:
XC2VP30	2	4	2	22	46	22	8	64	8	22	1	4

Σχήμα 43: Κατανομή δομικών στοιχείων μίας Virtex-II Pro Fpga σε στήλες και Frames

Είναι γεγονός ότι η Xilinx παρέχει ελάχιστες πληροφορίες σχετικά με το πως δομείται η πληροφορία μέσα στο bitsteam που προγραμματίζει την FPGA. Η κατανομή της πληροφορίας του bitstream [6] στις επιμέρους βαθμίδες της FPGA φαίνεται στο σχήμα 44.



Σχήμα 44: Κατανομή της πληροφορίας για αναδιάταξη μέσα στο bitstream