Ετήσια Τεχνική Έκθεση

Έτος 2014

ΘΑΛΗΣ – Πολυτεχνείο Κρήτης

Πλατφόρμα προηγμένων μαθηματικών μεθόδων και λογισμικού για την επίλυση προβλημάτων πολλαπλών πεδίων (multi physics, multidomain) σε σύγχρονες υπολογιστικές αρχιτεκτονικές: Εφαρμογή σε προβλήματα Περιβαλλοντικής Μηχανικής και Ιατρικής (MATENVMED- MIS 379416)

Δράση 3.2

ΥΛΟΠΟΙΗΣΗ ΣΕ FPGAs ΚΑΙ ΠΟΛΥΠΥΡΗΝΑ ΣΥΣΤΗΜΑΤΑ (MULTICORES / MANYCORES)

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Πίνακας Περιεχομένων

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

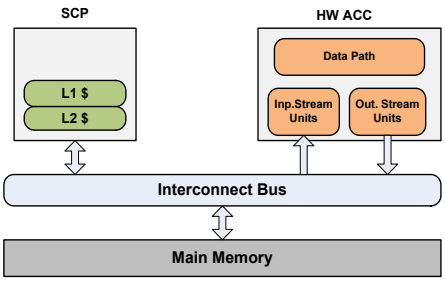
1. Σκοπός

Η έκθεση για το 2014 παρουσιάζει την πρόοδο της Δράσης 3.2 στις πλατφόρμες CPU και FPGA δίδοντας έμφαση στην ανάπτυξη εργαλείων που θα χρησιμοποιηθούν για υλοποίηση των elliptic PDE solvers σε FPGAs.

Θα παρουσιάσουμε τις βασικές αρχές, τον σχεδιασμό της αρχιτεκτονικής και την υλοποίηση του εργαλείου λογισμικού SOpenCL, που έχει αναπτυχθεί για τον σχεδιασμό επιταχυντών υλικού (hardware accelerators) από κώδικα OpenCL. Η βασική ιδέα του SOpenCL είναι να δώσει την δυνατότητα σε μηχανικούς λογισμικού χωρίς γνώσεις αρχιτεκτονικής και σχεδιασμό υλικού να δημιουργήσουν τέτοιους επιταχυντές για OpenCL εφαρμογές. Οι επιταχυντές αυτοί μπορούν να ενσωματωθούν σε ένα σύστημα που περιέχει ένα Host unit, συνήθως μία CPU γενικού σκοπού, και να λειτουργήσουν όπως και μία GPU επιταχύνοντας δηλ. τμήματα του κώδικα που έχουν υψηλές απαιτήσεις απόδοσης. Η SOpenCL έχει χρησιμοποιηθεί σε μία ευρεία γκάμα εφαρμογών και μερικές από αυτές θα περιγραφούν σε αυτήν την έκθεση.

2. Χρήση μοντέλου παράλληλου προγραμματισμού OpenCL για σύνθεση αρχιτεκτονικών (Silicon OpenCL)

Σε αυτήν την έκθεση θα περιγράψουμε το SOpenCL, ένα νέο εργαλείο και μεθοδολογία η οποία δημιουργεί επιταχυντές υλικού (hardware accelerators) που είναι τμήματα ενός ολόκληρου συστήματος (System On Chip, SoC), όπως φαίνεται και στην Εικόνα 1.



Εικόνα 1. Το τελικό μας σύστημα που περιλαμβάνει τον επιταχυντή υλικού (HW ACC) που παράγει η OpenCL και τον κύριο επεξεργαστή (Scalar Processor, SCP).

Η OpenCL είναι ένα βιομηχανικό standard για την δημιουργία παράλληλων προγραμμάτων με σκοπό την εκτέλεση σε ετερογενή συστήματα (heterogeneous systems) [1]. Απαλλάσσει τον προγραμματιστή από το να γράψει κώδικα για κάθε πιθανή ετερογενή πλατφόρμα και με το να ασχοληθεί με τεχνικές λεπτομέρειες

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

χαμηλού επιπέδου (low level details) δίδοντάς του την ευκαιρία να εστιάσει στον ίδιο τον αλγόριθμο και στο πρόβλημα που καλείται να επιλύσει. Έχουμε αναφερθεί λεπτομερώς στην OpenCL στην έκθεση του MATENVMED για το 2012.

Υπάρχουν κάποιες σημαντικές τεχνικές δυσκολίες στον σχεδιασμό και την υλοποίηση του SOpenCL [5]. Η OpenCL χρησιμοποιεί παράλληλους πυρήνες (kernels) για να εκφράσει υπολογισμούς σε πολύ χαμηλό επίπεδο (granularity), που είναι το επίπεδο του work-item (ή thread όπως έχουμε δει στην CUDA). Αυτό είναι ιδιαίτερα χρήσιμο γιατί φανερώνει τον παραλληλισμό της εφαρμογής στο χαμηλότερο επίπεδο και διευκολύνει την εκτέλεση του προγράμματος από CPU και GPU. Παρόλα αυτά, η δημιουργία επιταχυντών υλικού στο επίπεδο του work-item μπορεί να μην είναι ιδανική λύση λόγω του ότι κάθε τέτοιος επιταχυντής θα κάνει πολύ λίγη δουλειά κάθε φορά που καλείται.

Σαν πρώτη φάση, το εργαλείο SOpenCL περιέχει έναν μεταγλωττιστή source-to-source που μεταφράζει τον κώδικα OpenCL σε κώδικα C σε διαφορετικό όμως επίπεδο παραλληλισμού από τον κώδικα OpenCL. Στην συγκεκριμένη περίπτωση ο μεταγλωττιστής μετατρέπει τον κώδικα σε επίπεδο work-group (ή block), έτσι ώστε κάθε κλήση του επιταχυντή να εκτελεί εργασία που να αντιστοιχεί σε ένα work-group.

Η δεύτερη φάση του SOpenCL μετατρέπει τον C κώδικα σε έναν επιταχυντή υλικού σε synthesizable Verilog. Ο επιταχυντής ακολουθεί ένα συγκεκριμένο πρότυπο αρχιτεκτονικής (architectural template) το οποίο παίρνει συγκεκριμένη μορφή ανάλογα με τις απαιτήσεις απόδοσης του χρήστη και την συγκεκριμένη εφαρμογή που υλοποιείται στην FPGA. Η δεύτερη φάση του SOpenCL αποτελείται από μία σειρά από βελτιστοποιήσεις του compiler όπως predication, code slicing και modulo scheduling που εφαρμόζονται διαδοχικά στον κώδικα C. Στο τελικό στάδιο το εργαλείο SOpenCL δημιουργεί synthesizable Verilog.

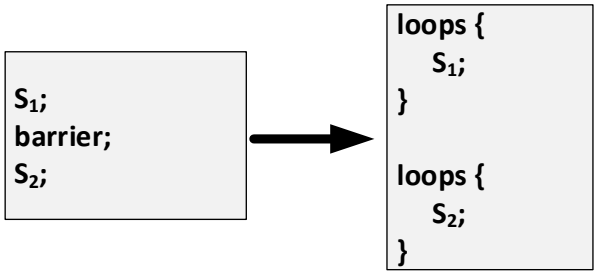
2.1. Πρώτη φάση του SOpenCL

Για να επιτύχουμε την αποδοτική εκτέλεση παράλληλων πυρήνων OpenCL, εφαρμόζουμε μια αλληλουχία από μετατροπές source to source για να ανεβάσουμε το επίπεδο του πυρήνα από το work-item στο work-group. Μετά τις μετατροπές αυτές στον πηγαίο κώδικα, ο νέος παράλληλος πυρήνας εκφράζει τον υπολογισμό που εκτελεί το work-group.

Η πρώτη φάση του SOpenCL αποτελείται από δύο βήματα:

1. Σειριοποίηση των λογικών νημάτων (Logical Thread Serialization). Αυτό το βήμα περικλείει τον κώδικα του πυρήνα με ένα τριπλό loop και με αυτόν τον τρόπο μετατρέπει τον υπολογισμό ενός work-item σε υπολογισμό ενός work-group.
2. Απαλοιφή εντολών συγχρονισμού (barriers). Η OpenCL δίνει την δυνατότητα στον προγραμματιστή να συγχρονίσει την εκτέλεση των work-items που εκτελούν έναν κώδικα OpenCL μέσω των εντολών barrier. Μία εντολή barrier αναγκάζει όλα τα work-items ενός work-group να εκτελέσουν την εντολή αυτή πριν μπορέσει οποιοδήποτε work-item να συνεχίσει με την εκτέλεση των

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Εικόνα 2. Απαλοιφή εντολών barriers με την χρήση loop fission.

εντολών μετά την barrier. Αντίστοιχα, η εντολή barrier μέσα σε ένα loop αναγκάζει όλα τα work-items να περιμένουν στην εντολή αυτή πριν μπορέσουν να συνεχίσουν με την επόμενη επανάληψη του loop.

Για να εξασφαλίσουμε σωστή λειτουργικότητα κατά την μετατροπή του κώδικα από OpenCL σε C σε περίπτωση που υπάρχουν εντολές barrier, χωρίζουμε τον κώδικα σε δύο μπλόκς, ένα πριν την barrier και ένα μετά έτσι ώστε αυτά τα μπλόκς να μην περιέχουν την εντολή barrier. Μετά εφαρμόζουμε την τεχνική loop fission για να χωρίσουμε το αρχικό τριπλό loop σε δύο επιμέρους τριπλά loops για να εξασφαλίσουμε την σωστή μετατροπή του κώδικα σε C (Εικόνα 2).

Περισσότερες λεπτομέρειες για τον μετατροπές που εφαρμόζει ο compiler υπάρχουν στο [5].

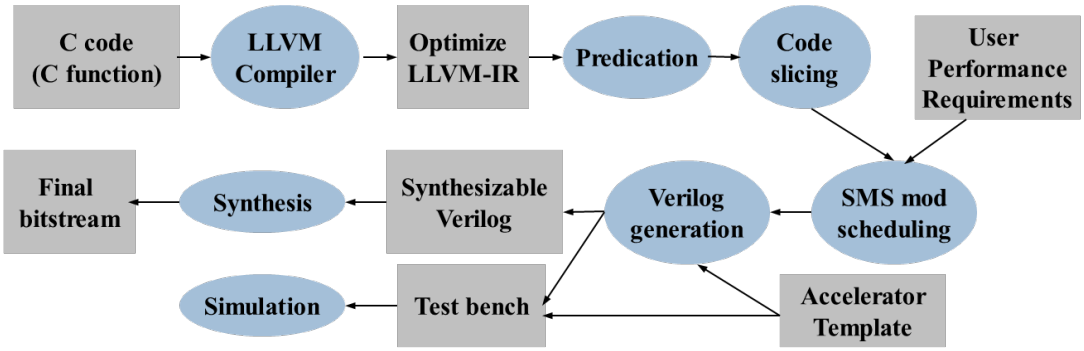
2.2. Δεύτερη φάση του SOpenCL

Μετά την πρώτη φάση, το SOpenCL μεταφράζει τον κώδικα σε Verilog όπως φαίνεται και στην Εικόνα 3. Ο αλγόριθμος αυτός χρησιμοποιεί και επεκτείνει τον LLVM compiler [3]. Ένα σημαντικό χαρακτηριστικό του SOpenCL είναι ότι χρησιμοποιεί ένα καλά σχεδιασμένο αρχιτεκτονικό πρότυπο (template) για να δημιουργήσει υλικό όπως φαίνεται στην Εικόνα 4.

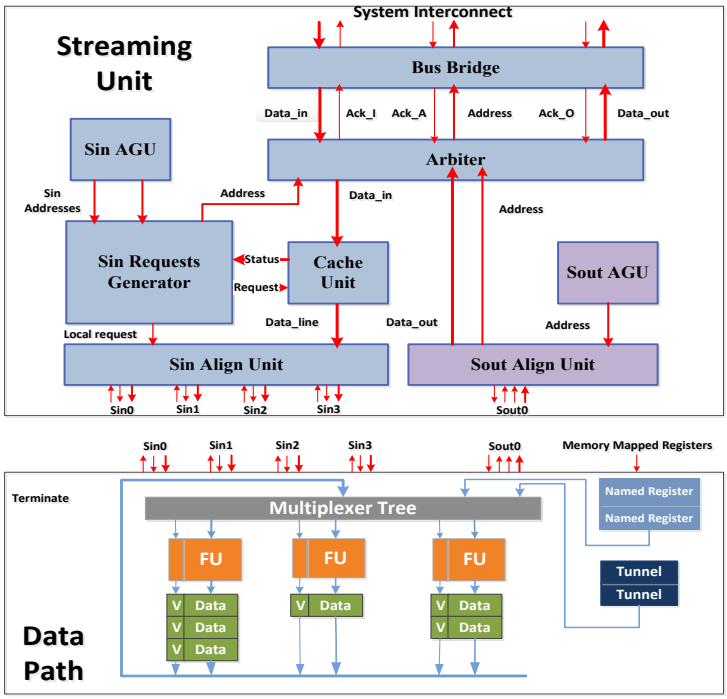
Πρότυπο αρχιτεκτονικής

Η μονάδα δεδομένων της Εικόνα 4 αποτελείται από ένα δίκτυο από λειτουργικές μονάδες (Functional Units, FUs) που αναλαμβάνουν το κυρίως έργο της επεξεργασίας δεδομένων. Οι μονάδες αυτές λαμβάνουν δεδομένα από τις μονάδες Εισόδου/Εξόδου Ροής σε μορφή FIFO και παράγουν τα δεδομένα εξόδου πάλι σε μορφή FIFO. Η διασύνδεση μεταξύ των λειτουργικών μονάδων γίνεται μέσω κυκλωμάτων ουρών FIFO που αποτελούνται από πολυπλέκτες και buffers για να κατευθύνουν δεδομένα από τις μονάδες που παράγουν στις μονάδες που καταναλώνουν ενδιάμεσα αποτελέσματα.

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Εικόνα 3. Ο αλγόριθμος δημιουργίας επιταχυντή υλικού από κώδικα C.



Εικόνα 4. Το πρότυπο αρχιτεκτονικής που χρησιμοποιούμε περιλαμβάνει την μονάδα Δεδομένων (Data Path) και την μονάδα Εισόδου/Εξόδου Ροής (Streaming Unit)

Το SOpenCL διαμορφώνει τον τελικό επιταχυντή υλικού μεταβάλλοντας τον τύπο του μίγματος των FUs (δηλ. πόσους multipliers, adders, shifters, κοκ), την συγκεκριμένη λειτουργία που επιτελούν, την διασύνδεση μεταξύ των FUs καθώς και το bandwidth μεταξύ της μονάδας δεδομένων και της μονάδας Εισόδου/Εξόδου.

Η μονάδα Εισόδου/Εξόδου Ροής αναλαμβάνει όλες τις αρμοδιότητες που έχουν να

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

κάνουν με την μεταφορά δεδομένων μεταξύ της κύριας μνήμης και του επιταχυντή. Αυτές οι αρμοδιότητες είναι η δημιουργία των διευθύνσεων μνήμης, ευθυγράμμιση και ταξινόμηση δεδομένων από και προς τον επιταχυντή καθώς και προσαρμογή μεταξύ του επιταχυντή και του διαύλου του συστήματος (system bus). Η μονάδα Εισόδου/Εξόδου Ροής αποτελείται από μία ή περισσότερες μονάδες Εισόδου και Εξόδου. Όπως και για την μονάδα Δεδομένων, έτσι και εδώ το SOpenCL συνθέτει το κύκλωμα έτσι ώστε να ταιριάζει στα χαρακτηριστικά της εφαρμογής και στις απαιτήσεις του υπόλοιπου συστήματος.

Ροή του αλγορίθμου SOpenCL

Αναφερόμενος στην Εικόνα 3, βλέπουμε ότι το εργαλείο SOpenCL ακολουθεί μια σειρά από βήματα στον μεταγλωττιστή LLVM για να παράγει το τελικό κύκλωμα του επιταχυντή σε Verilog. Εν συντομία, τα πιο σημαντικά βήματα είναι τα εξής:

a) Compiler optimizations and Predication. Το πρώτο βήμα είναι να μετατρέψουμε τον κώδικα C σε εσωτερική μορφή του LLVM. Ο ίδιος ο LLVM εφαρμόζει μια σειρά από στάνταρντ βελτιστοποιήσεις του κώδικα όπως constant propagation, dead code elimination, strength reduction, κοκ. Το predication είναι μία τεχνική του μεταγλωττιστή η οποία μετατρέπει εξαρτήσεις ελέγχου (control dependencies) σε εξαρτήσεις δεδομένων (data dependencies). Το predication εξαλείφει όλες τις εντολές διακλάδωσης (branches) στον κώδικα C με την χρήση επιπλέον μεταβλητών του 1-bit οι οποίες ονομάζονται Boolean guards. Ο βασικός σκοπός του predication είναι η εξάλειψη όλων των διακλαδώσεων που προέρχονται από εντολές if-then-else, και η προετοιμασία του κώδικα για την επόμενη φάση του modulo scheduling.

b) Code slicing. Ο κώδικας C μετά το πρώτο βήμα εμπεριέχει εντολές Εισόδου/Εξόδου δηλ. εντολές φορτώματος και αποθήκευσης στην μνήμη καθώς και εντολές υπολογισμού. Το βήμα του code slicing διαχωρίζει τον κώδικα σε 3 κομμάτια: (i) το κομμάτι κώδικα που κάνει μόνο υπολογισμούς, (ii) το κομμάτι κώδικα που περιέχει όλες τις εντολές φορτώματος (load) από την μνήμη και τις εντολές υπολογισμού της διεύθυνσης φορτώματος, και (iii) το κομμάτι κώδικα που περιέχει όλες τις εντολές αποθήκευσης (store) από την μνήμη και τις εντολές υπολογισμού της διεύθυνσης αποθήκευσης.

Ο λόγος που εκτελούμε αυτό το βήμα είναι επειδή κάθε ένα από τα 3 κομμάτια του κώδικα αντιστοιχεί και σε διαφορετικό κομμάτι της αρχιτεκτονικής: το υπολογιστικό κομμάτι αντιστοιχεί στην μονάδα επεξεργασίας δεδομένων, ενώ τα κομμάτια για load και store αντιστοιχούν στην μονάδα Εισόδου/Εξόδου Ροής. Με το να ξεχωρίσουμε τον κώδικα σε 3 διακριτά μέρη, μπορούμε να διαμορφώσουμε κάθε ένα τμήμα του επιταχυντή ανεξάρτητα χρησιμοποιώντας πληροφορία από το αντίστοιχο τμήμα του κώδικα.

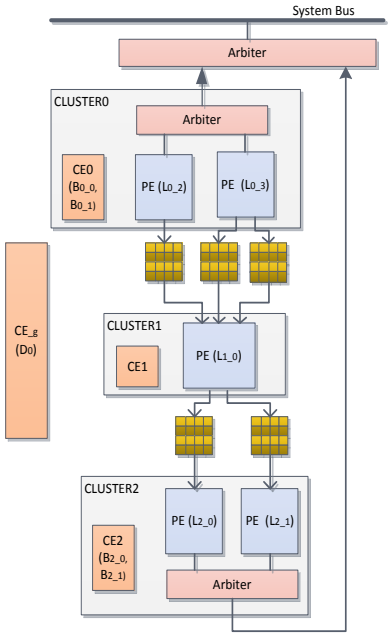
c) Swing Modulo Scheduling (SMS). Η τεχνική SMS είναι μια τεχνική χρονοπρογραμματισμού εντολών (instruction scheduling) η οποία

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

εκμεταλλεύεται τον παραλληλισμό μεταξύ εντολών που ανήκουν σε διαφορετικές επαναλήψεις ενός loop και τις προγραμματίζει να εκτελεσθούν παράλληλα [SMS]. Ο αλγόριθμος SMS χρησιμοποιεί ευριστικές μεθόδους για να ελαχιστοποιήσει το Iteration Interval (II), που είναι το σταθερό διάστημα αριθμού κύκλων μηχανής στο οποίο ξεκινάμε διαδοχικές επαναλήψεις του loop. Το SMS εφαρμόζεται σε κάθε loop και χωριστά σε κάθε ένα από τα τρία τμήματα του κώδικα όπως αυτά έχουν παραχθεί από το code slicing.

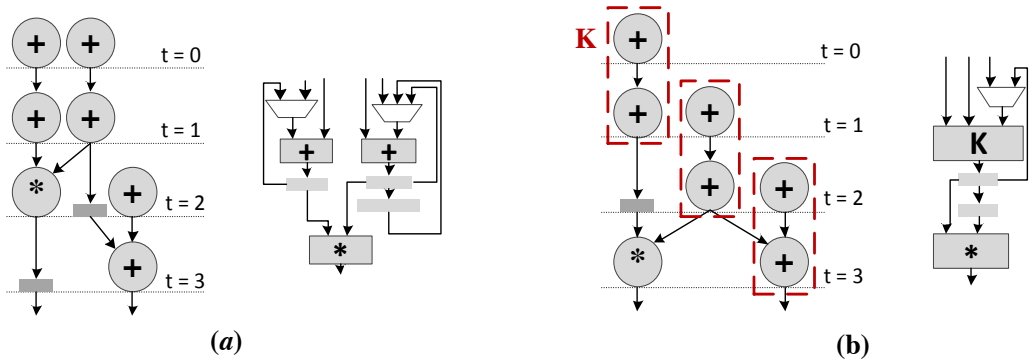
d) Δημιουργία υλικού. Το τελικό βήμα της δεύτερης φάσης του SOpenCL είναι η δημιουργία του επιταχυντή υλικού σε Verilog χρησιμοποιώντας την έξοδο που δημιουργεί το SMS και το αρχιτεκτονικό πρότυπο που περιεγράφηκε προηγουμένως. Εκτός από το υλικό του επιταχυντή, το SOpenCL δημιουργεί ακόμα και ένα αρχείο ελέγχου (testbench) το οποίο χρησιμοποιείται για την αυτοματοποίηση της διαδικασίας ελέγχου του παραχθέντος κυκλώματος.

Η Εικόνα 5 δείχνει το διάγραμμα του επιταχυντή που υλοποιεί τον αλγόριθμο για LU Decomposition. Ο συγκεκριμένος αλγόριθμος αποτελείται από πολλαπλά loops τα οποία δημιουργούν και παραπάνω του ενός επιταχυντές.



Εικόνα 5. Το αρχιτεκτονικό διάγραμμα ενός επιταχυντή που υλοποιεί τμήμα ενός αλγορίθμου LU Decomposition.

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Εικόνα 6. Χρονοδρομολόγηση εντολών σε ένα Data Flow Graph (DFG) με a) βασικές και απλές εντολές και b) με μακροεντολές. Η μακροεντολή K υλοποιείται με το MFU K που είναι ένας 3-bit προσθετής με αρχιτεκτονική διοχέτευσης (pipelined).

2.3. Instruction clustering

Μία επιπλέον βελτιστοποίηση του SOpenCL είναι η ομαδοποίηση (clustering) εντολών για την δημιουργία Macro Functional Units (MFUs), δηλ. λειτουργικών μονάδες που συνδυάζουν παραπάνω από μια απλούστερες λειτουργικότητες [6]. Ο λόγος που δημιουργούμε τέτοια MFUs είναι για να μειώσουμε τον αριθμό των διασυνδέσεων μεταξύ μικρών λειτουργικών μονάδων. Με άλλα λόγια, θέλουμε να μειώσουμε την επιβάρυνση των διασυνδέσεων σε σχέση με τις λειτουργικές μονάδες. Είναι γνωστό μάλιστα ότι τις περισσότερες φορές, ή έλλειψη επαρκούς αριθμού καναλιών διασύνδεσης είναι και ο κυριότερος λόγος για το ότι ένα κύκλωμα δεν μπορεί να υλοποιηθεί σε μια FPGA.

Χρησιμοποιούμε μία μέθοδο που βασίζεται σε grammar-induction τεχνικές για να ομαδοποιήσουμε βασικές εντολές σε μακρο-εντολές (macroinstructions) οι οποίες με την σειρά τους υλοποιούνται σαν MFUs με μικρότερες απαιτήσεις σε πολυπλοκότητα διασυνδέσεων.

Η Εικόνα 6 δείχνει ένα παράδειγμα για το πώς δύο διαδοχικές προσθέσεις (μέσα στα κόκκινα κουτιά) μπορούν να συγχωνευθούν για εκτέλεση σε ένα MFU αντί να εκτελεσθούν σε ένα FU που κάνει πρόσθεση. Η βασική ιδέα είναι ότι μία μεγαλύτερη λειτουργική μονάδα μειώνει την ανάγκη για πολλές διασυνδέσεις μεταξύ μικρότερων μονάδων και συνεπώς αυξάνει την πιθανότητα να έχουμε ένα κύκλωμα το οποίο να μπορεί να πραγματοποιηθεί σε μία FPGA.

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

2.4. Αποτελέσματα του SOpenCL στα μετροπρογράμματα OpenDwarfs

Το εργαλείο SOpenCL χρησιμοποιήθηκε για την υλοποίηση επιταχυντών υλικού για εφαρμογές που ανήκουν στα Dwarfs. Τα 13 Dwarfs είναι σύνολα εφαρμογών που κάθε ένα από αυτά τα σύνολα μπορούν να χαρακτηρισθούν από ένα κοινό πρότυπο (pattern) υπολογισμών και επικοινωνίας. Για παράδειγμα, ένα από τα 13 Dwarfs είναι το Dense Linear Algebra το οποίο περιλαμβάνει αλγορίθμους που επεξεργάζονται πυκνούς πίνακες (dense arrays). Ένας τέτοιος αλγόριθμος που έχουμε χρησιμοποιήσει για να αναπτύξουμε επιταχυντές υλικού είναι ο LU Decomposition [3]. Για να είμαστε πιο ακριβείς, χρησιμοποιούμε ένα σύνολο από μετροπρογράμματα, τα OpenDwarfs, που είναι open-source υλοποιήσεις σε OpenCL των Dwarfs.

Χωρίς να μπούμε σε λεπτομέρειες σχετικά με την υλοποίηση των OpenDwarfs με την χρήση του SOpenCL, τα αποτελέσματα δείχνουν ότι οι FPGAs είναι ανταγωνιστικές με τις GPUs όσον αφορά την ταχύτητα εκτέλεσης τέτοιων αλγορίθμων. Εάν λάβουμε υπόψιν μας και την κατανάλωση ενέργειας, στις οποίες οι FPGA υπερτερούν εμφανώς των GPUs, μπορούμε να βγάλουμε το συμπέρασμα ότι η χρήση εργαλείων σαν το SOpenCL έχουν την δυνατότητα να κάνουν τις FPGA πολύ ελκυστικές για την υλοποίηση παρόμοιων εφαρμογών. Περισσότερες πληροφορίες σχετικά με το SOpenCL και τις FPGAs μπορείτε να βρείτε στο [3].

3. Βιβλιογραφία

1. Khronos OpenCL Working Group. Editor: A. Munshi, “The OpenCL Specification”, Version: 1.1 Document Revision: June 11, 2010.
2. Konstantinos Krommydas, Wu-Chun Feng, Muhsen Owaida, Christos D. Antonopoulos and Nikolaos Bellas. On the Portability of OpenCL Dwarfs on Fixed and Reconfigurable Parallel Platforms. *25th IEEE International Conference on Application-specific Systems, Architectures and Processors (ASAP)*. June 18-20, 2014, Zurich, Switzerland.
3. C. Lattner and V. Adve. “LLVM: A Compilation Framework for Lifelong Program Analysis Transformation, *“Proceedings of the International Symposium on Code Generation and Optimization (CGO'04)*, March 2004, pp. 75-86, Palo Alto, CA.
4. J. Llosa, A. Gonzalez, E. Ayguade, and M. Valero. “Swing Modulo Scheduling: A Lifetime-Sensitive Approach, *“Proceedings of the 1996 Conference on Parallel Architectures and Compilation Techniques (PACT)*, October, 1996, pp. 80-86, Boston, MA.

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

5. Muhsen Owaida, Nikolaos Bellas, Konstantis Daloukas, Christos D Antonopoulos. Synthesis of Platform Architectures from OpenCL Programs. *IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM)*, May 1-3, 2011, Salt Lake City, UT

6. Muhsen Owaida, Christos D. Antonopoulos, Nikolaos Bellas. A Grammar Induction Method for Clustering of Operations in Complex FPGA Designs. *IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM), Regular paper.* May 11-13, 2014. Boston, MA

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης