



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ
www.cslab.ece.ntua.gr

ΠΡΟΗΓΜΕΝΑ ΘΕΜΑΤΑ ΑΡΧΙΤΕΚΤΟΝΙΚΗΣ ΥΠΟΛΟΓΙΣΤΩΝ
Ακ. έτος 2007-2008, 8ο εξάμηνο, Σχολή ΗΜ&ΜΥ

1η ΕΡΓΑΣΙΑ

Τελική Ημερομηνία Παράδοσης: 22 Απριλίου 2008 (δεν θα δοθεί παράταση)

1. Εισαγωγή

Αντικείμενο της εργασίας είναι η μελέτη της επίδρασης διαφόρων παραμέτρων της οργάνωσης της κρυφής μνήμης στην απόδοση των εφαρμογών. Για το σκοπό αυτό θα χρησιμοποιηθεί ο προσομοιωτής SESC. Πριν προχωρήσετε στην κυρίως εκφώνηση της άσκησης, είναι απαραίτητο να έχετε διαβάσει τις πληροφορίες σχετικά με τον προσομοιωτή, την εγκατάσταση και τη χρήση του, που υπάρχουν στη σελίδα:

http://www.cslab.ntua.gr/courses/advcomparch/sesc_installation.go

Η μεταγλώττιση και η εγκατάσταση του προσομοιωτή θα γίνει όπως περιγράφεται στην παραπάνω σελίδα, με τη διαφορά ότι στην εντολή `./configure` (βήμα 3) δε θα δώσετε το επιπλέον όρισμα `--enable-smp`. Επιπλέον, το όνομα του εκτελέσιμου του προσομοιωτή που θα παραχθεί μετά την μεταγλώττιση θα είναι το `"sesc.mem"` (αντί του `"sesc.smp"`).

Θα παραμετροποιηθούν διάφορα χαρακτηριστικά της ιεραρχίας κρυφής μνήμης του επεξεργαστή. Η ιεραρχία κρυφής μνήμης που προσομοιώνεται από τον SESC "by default", αποτελείται από L1 cache δεδομένων, L1 cache εντολών, και L2 cache ενοποιημένη (δεδομένων και εντολών).

Οι παράμετροι που καθορίζουν τα χαρακτηριστικά λειτουργίας της L1 D-cache (δεδομένων), προδιαγράφονται στο αρχείο `"sesc.conf"` (το οποίο έχει δημιουργηθεί με την εντολή `make sesc.conf` κατά την εγκατάσταση του SESC):

```
[DataL1]
deviceType = 'cache'
blockName = "Dcache"
MSHR = "DL1MSHR"
size = 16*1024
assoc = 4
skew = false
bsize = 64
replPolicy = 'RANDOM'
numPorts = $(memSizing)
portOccp = 1
hitDelay = 2
missDelay = 1
writePolicy = "WB"
lowerLevel = "CommonBus Bus shared"
```

Οι βασικότερες παράμετροι που καθορίζουν τα χαρακτηριστικά της L1 είναι οι εξής:

- **size**: το μέγεθος της cache σε bytes
- **assoc**: το associativity της cache
- **bsize**: το block size σε bytes
- **replPolicy**: η πολιτική αντικατάστασης blocks
- **hitDelay**: ο αριθμός κύκλων για την εξυπηρέτηση ενός hit
- **missDelay**: ο επιπλέον αριθμός κύκλων (προστίθεται στο hitDelay) για την εξυπηρέτηση ενός miss
- **writePolicy**: η πολιτική εγγραφής

Ομοίως, οι παράμετροι λειτουργίας της L1 I-cache (εντολών) προδιαγράφονται στο αρχείο "shared.conf" (το οποίο χρησιμοποιείται από το sesc.conf) :

```
[InstL1]
deviceType = 'icache'
blockName  = "Icache"
size       = 16*1024
assoc      = 2
bsize     = 64
writePolicy = 'WB'
replPolicy = 'LRU'
numPorts  = 2
portOccp  = 1
hitDelay  = 2
missDelay = 0
MSHR      = 'InstL1MSHR'
lowerLevel = "L2Cache L2 shared"
```

Η σημασία των παραμέτρων αυτών είναι η ίδια με αυτή της περίπτωσης της DataL1.

Τέλος, οι παράμετροι λειτουργίας της L2 cache προδιαγράφονται στο αρχείο "shared.conf":

```
[L2Cache]
deviceType = 'cache'
blockName  = "L2"
size       =1024*1024
assoc      = 8
bsize     = 64
writePolicy = 'WB'
replPolicy = 'LRU'
numPorts  = 1
portOccp  = 1
hitDelay  = 10
missDelay = 4
MSHR      = 'MSHRL2'
lowerLevel = "$ (L2l1) "
```

Κι εδώ, η σημασία των παραμέτρων αυτών είναι η ίδια με αυτή της περίπτωσης της DataL1.

2. Μετροπρογράμματα

Για την εκτέλεση της άσκησης παρέχονται 2 εκτελέσιμα από το πακέτο των SPEC CPU2000 benchmarks, τα mcf και crafty.

Το mcf είναι γραμμένο σε C και έχει σχεδιαστεί για την επίλυση προβλημάτων δρομολόγησης μέσω μαζικής μεταφοράς. Δέχεται σαν είσοδο τον αριθμό των δρομολογίων, την ώρα αναχώρησης και άφιξης καθώς και το κόστος κάθε δρομολογίου και υπολογίζει το βέλτιστο χρονοδιάγραμμα.

(Πιο λεπτομερής περιγραφή στο <http://www.spec.org/cpu/CINT2000/181.mcf/docs/181.mcf.txt>). Το πρόγραμμα δέχεται το αρχείο εισόδου σαν όρισμα. Επομένως μπορείτε να το εκτελέσετε ως εξής:

```
$ sesc.mem [SESC-OPTIONS] mcf.mips inp.in
```

Το crafty είναι ένα παιχνίδι σκακιού γραμμένο σε C. Δέχεται σαν είσοδο μια παρτίδα σκακιού καθώς και το βάθος στο οποίο επιτρέπεται ο υπολογιστής να ξεετάζει το δένδρο των πιθανών κινήσεων πριν αποφασίσει την επόμενη κίνηση. Η έξοδος του προγράμματος είναι το σύνολο των πιθανών κινήσεων και ποια είναι η καλύτερη κίνηση που επιλέγεται για τη συνέχεια του παιχνιδιού.

(Πιο λεπτομερής περιγραφή στο <http://www.spec.org/cpu/CINT2000/186.crafty/docs/186.crafty.txt>).

Το πρόγραμμα περιμένει τα δεδομένα στο stdin, επομένως πρέπει να το εκτελέσετε ως εξής:

```
$ sesc.mem [SESC-OPTIONS] crafty.mips <crafty.in
```

Τα εκτελέσιμα και τα αντίστοιχα αρχεία εισόδου είναι διαθέσιμα στα ακόλουθα URLs:

<http://www.cslab.ntua.gr/courses/advcomparch/sesc/askisil/crafty.mips>

<http://www.cslab.ntua.gr/courses/advcomparch/sesc/askisil/crafty.in>

<http://www.cslab.ntua.gr/courses/advcomparch/sesc/askisil/mcf.mips>

<http://www.cslab.ntua.gr/courses/advcomparch/sesc/askisil/inp.in>

3. Πειραματική διαδικασία

Στο πλαίσιο της εργασίας, θα διερευνηθεί η επίδραση των βασικότερων παραμέτρων ιεραρχίας κρυφής μνήμης στην απόδοση της εφαρμογής. Τα πειράματα που θα χρειαστεί να εκτελέσετε παρουσιάζονται στη συνέχεια.

3.1. L1 D-cache (δεδομένων) και L1 I-cache (εντολών)

Για όλες τις περιπτώσεις που εξετάζονται στο πείραμα αυτό, οι παράμετροι της L2 cache θα διατηρηθούν σταθερές και ίσες με τις default παραμέτρους που υπάρχουν στο configuration αρχείο (shared.conf).

3.1.a. Αρχικά, παραμετροποιούμε τις DataL1 και InstL1 ως εξής: hitDelay=1, missDelay=3, replPolicy='LRU', bsize=64. Προσομοιώστε και για τα 2 benchmarks τις παρακάτω περιπτώσεις:

size (L1D/L1I)	associativity (L1D/L1I)
8K	2
8K	4
16K	2
16K	4
32K	4
32K	8
64K	4
64K	8

3.1.b. Αλλάξτε το replacement policy των DataL1 και InstL1 σε 'RANDOM' και προσομοιώστε για τα 2 benchmarks τις παρακάτω περιπτώσεις:

size (L1D/L1I)	associativity (L1D/L1I)
32K	4
32K	8
64K	4
64K	8

3.1.c. Αλλάξτε πάλι το replacement policy των DataL1 και InstL1 σε 'LRU', και προσομοιώστε για τα 2 benchmarks τις παρακάτω περιπτώσεις:

size (L1D/L1I)	associativity (L1D/L1I)
16K	4
32K	4
32K	8
64K	4
64K	8

Για κάθε μία από τις παραπάνω περιπτώσεις χρησιμοποιείτε διαδοχικά τις ακόλουθες τιμές για το block size: 32, 128. Σημειώνουμε ότι πρέπει να χρησιμοποιήσετε την ίδια τιμή bsize κάθε φορά τόσο στις DataL1,InstL1 όσο και στην L2Cache.

3.2. L2 cache

Για όλες τις περιπτώσεις που θα εξετάσουμε εδώ, παραμετροποιούμε τις DataL1 και InstL1 ως εξής: size=32K, assoc=4, hitDelay=1, missDelay=3, replPolicy='LRU', bsize=64.

3.2.a. Αρχικά, παραμετροποιούμε την L2Cache ως εξής: hitDelay=16, missDelay=10, replPolicy='LRU', bsize=64. Προσομοιώστε και για τα 2 benchmarks τις παρακάτω περιπτώσεις:

size	associativity
256K	4
512K	4
512K	8
1024K	4
1024K	8
1024K	16
2048K	4
2048K	8
2048K	16

3.2.b. Αλλάξτε το `missDelay` της L2 σε 100 κύκλους και προσομοιώστε για τα 2 benchmarks τις εξής περιπτώσεις:

size	associativity
512K	4
1024K	4
1024K	8
2048K	8

4. Ζητούμενο

Σαν βασική μετρική απόδοσης θα χρησιμοποιήσετε το IPC (Instructions Per Cycle). Με την προϋπόθεση ότι ο κύκλος μηχανής και ο εκτελούμενος αριθμός εντολών παραμένουν σταθεροί κάθε φορά, μεγαλύτερες τιμές στο IPC υποδεικνύουν καλύτερη απόδοση (σημείωση: αυτό ισχύει μόνο στα πλαίσια της προσομοίωσης. Στην πράξη, οι διάφορες τροποποιήσεις στα μικροαρχιτεκτονικά χαρακτηριστικά του επεξεργαστή επιφέρουν συνήθως αλλαγές και στην διάρκεια του κύκλου ρολογιού).

Για κάθε ένα από τα παραπάνω ερωτήματα μελετήστε τις μεταβολές στο IPC και στο miss rate της cache της οποίας τις παραμέτρους μεταβάλλετε. Παρουσιάστε σε γραφικές παραστάσεις τις μεταβολές αυτές για κάθε περίπτωση. Για παράδειγμα, στο πείραμα 3.1.a, ζητούνται στην ουσία 2 γραφικές παραστάσεις, 1 για IPC και 1 για τα miss rates. Κάθε μία από αυτές τις γραφικές παραστάσεις θα έχει στον άξονα x την παράμετρο που μεταβάλλεται (στην περίπτωση αυτή `size,associativity`) και θα περιέχει τα αποτελέσματα και για τα 2 benchmarks.

Συνοψίστε τα συμπεράσματα των προηγούμενων ερωτημάτων. Ποιες από τις παραμέτρους που εξετάσατε έχουν τη μεγαλύτερη επίδραση στην απόδοση;

Παραδοτέο της άσκησης θα είναι ένα ηλεκτρονικό κείμενο (pdf, doc ή odt). Στο ηλεκτρονικό κείμενο να αναφέρετε στην αρχή τα στοιχεία σας (Όνομα, Επώνυμο, ΑΜ).

Η άσκηση θα παραδοθεί μόνο ηλεκτρονικά στην ιστοσελίδα:
<http://www.cslab.ece.ntua.gr/courses/comparch/submit>.

Δουλέψτε ατομικά. Έχει ιδιαίτερη αξία για την κατανόηση του μαθήματος να κάνετε μόνοι σας την εργασία. Μην προσπαθήσετε να την αντιγράψετε απλά από άλλους συμφοιτητές σας.

Μην αφήσετε την εργασία για το τελευταίο Σαββατοκύριακο, απαιτεί αρκετό χρόνο για την εκτέλεση των benchmarks, ξεκινήστε αμέσως!