



ΑΣΚΗΣΗ 3

Με Θέμα

«ΜΕΛΕΤΗ ΧΑΡΑΚΤΗΡΙΣΤΙΚΩΝ ΣΥΓΧΡΟΝΩΝ
SUPERSCALAR, OUT-OF-ORDER ΕΠΕΞΕΡΓΑΣΤΩΝ»

για το μάθημα

Προηγμένα Θέματα

Αρχιτεκτονικής Υπολογιστών

Γεώργιος Σκουρτσίδης (03114307)

ΕΙΣΑΓΩΓΗ

Στην άσκηση αυτή θα μελετήσουμε τα χαρακτηριστικά των σύγχρονων superscalar, out-of-order επεξεργαστών και την επίδραση τους στην απόδοση του συστήματος, την κατανάλωση ενέργειας και το μέγεθος του chip.

Τα χαρακτηριστικά που θα μελετήσουμε είναι το **dispatch width** και το **window size**. Το μέγεθος dispatch width αφορά τον αριθμό των εντολών που μπορούν να γίνουν issue ταυτόχρονα. Το window size είναι το μέγεθος του Re-Order Buffer (ROB).

Ο επεξεργαστής που προσομοιώνουμε ονομάζεται Gainestown. Βασίζεται στη μικροαρχιτεκτονική Nehalem της Intel, ενώ το μέγεθος των transistors βρίσκεται στα 45nm.

Χρησιμοποιήσαμε τα εξής 12 SPEC_CPU2006 benchmarks:

- | | | |
|------------------|---------------|-------------------|
| 1. 403.gcc | 5. 445.gobmk | 9. 459.GemsFDTD |
| 2. 429.mcf | 6. 450.soplex | 10. 471.omnetpp |
| 3. 434.zeusmp | 7. 456.hmmer | 11. 473.astar |
| 4. 436.cactusADM | 8. 458.sjeng | 12. 483.xalancbmk |

Στα παραπάνω benchmarks χρησιμοποιήθηκαν τα εξής configurations για τις 2 προαναφερθείσες παραμέτρους:

dispatch_width	1	2	4	8	16	32						
window_size	1	2	4	8	16	32	64	96	128	192	256	384

Δυστυχώς όμως αντιμετωπίσα το πρόβλημα που αναφέρεται σε πολλά e-mail στη mailing list του μαθήματος και για πολλά benchmarks δεν ολοκληρώθηκε στο 100% η εκτέλεσή τους. Συγκεκριμένα μέσα στο αρχείο **pinball_replay.app0.txt** βλέπουμε το εξής error:

```
pinball_replay.app0.txt - OUTPUTS - Visual Studio Code
Terminal Help
pinball_replay.app0.txt x
omnetpp > omnetpp.DW_04-WS_032.out > pinball_replay.app0.txt
38 RIP 0x00051c5a7
39 [0] ContextChangeCallback(0, 0, 0x7f4e13119bb8, 0x0000000000, 11) at PC: 0x00051c5a7
40 [0] Thread received fatal signal 11 @ icount 14110
41 [0] Dumped 0 events from the trace
42 [0] No other threads exist
43 ERROR: Thread 0 dying after unexpected exception @icount: 14110 PC 0x00051c5a7
44 Exiting...
45 |
```

Στη συνέχεια, σύμφωνα με σχετική οδηγία, αγνοήθηκαν προσομοιώσεις για τις οποίες το συνολικό ποσοστό εντολών που εκεκτελέστηκαν ήταν χαμηλότερο από 10%.

Τα ποσοστά του μέσου όρου εκτέλεσης των εντολών για το κάθε benchmark παρατίθενται παρακάτω (πραγματοποιήθηκε στρογγυλοποίηση στο 2ο δεκαδικό ψηφίο).

<u>mcf</u> :	100%
<u>zeusmp</u> :	100%
<u>cactusADM</u> :	100%
<u>GemsFDTD</u> :	89%
<u>sjeng</u> :	23.43%
<u>gcc</u> :	16.67%
<u>gobmk</u> :	14.08%
<u>hmmer</u> :	1.32%
<u>astar</u> :	0.4%

soplex: 0.09%

omnetpp: 0.01%

xalancbmk: 0.01%

Το ποσοστό εκτέλεσης των soplex, omnetpp, astar, hmmmer, xalancbmk δε κρίθηκε ικανοποιητικό και για αυτό το λόγο παραλήφθηκαν.

ΕΡΩΤΗΜΑ 1

Εκφώνηση

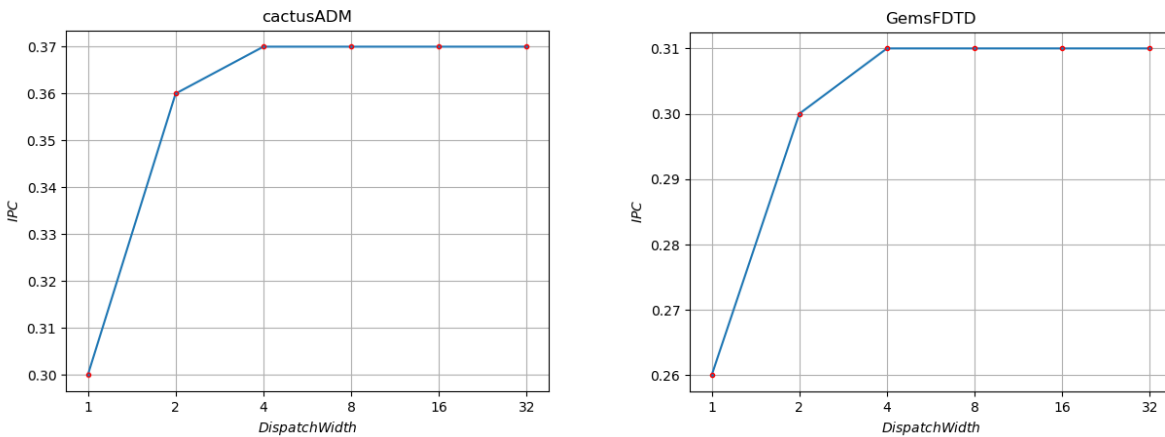
Χρειάζεται πραγματικά να προσομοιώσετε και τους 72 διαφορετικούς επεξεργαστές που προκύπτουν με βάση τις παραπάνω τιμές; Αν όχι, εξηγήστε ποιές περιπτώσεις μπορείτε να παραλείψετε και γιατί. Δικαιολογήστε την απάντησή σας όχι μόνο θεωρητικά αλλά και αποδεικνύοντας για ένα μικρό αριθμό αυτών των περιπτώσεων ότι καλώς τις παραλείψατε.

Λύση

Καθώς η παράμετρος **window size (WS)** αναφέρεται στο μέγεθος του Re-Order Buffer (ROB) ,ενώ η παράμετρος **dispatch windth(DW)** στον αριθμό των εντολών που μπορούν να γίνουν issue σε κάθε κύκλο ρολογιού,καταλήγουμε στο συμπέρασμα πως δεν χρειάζεται να προσομοιωθούν όλοι οι συνδυασμοί **dw** με **ws**.Καθώς ο συνολικός αριθμός εντολών που μπορεί να γίνει commit είναι μικρότερος ή ίσος του μεγέθους του reorder buffer, καταλήγουμε στο συμπέρασμα πως κρατώντας σταθερό το μέγεθος ws και αυξάνοντας το dw,το IPC αυξάνεται μόνο μέχρις ότου οι 2 παράμετροι να γίνουν ίσες (dw = ws).Περαιτέρω αύξηση του dw δε συνεπάγεται βελτίωση στο IPC καθώς όσες εντολές και αν γίνουν issue/κύκλο,θα είναι αδύνατον να γίνουν commit παραπάνω εντολές από όσες είναι το μέγεθος του ROB.

- Συνεπώς παραλείφθηκαν οι προσομοιώσεις για τις οποίες ισχύει :
DW > WS

Προκειμένου να αποδειχθεί ο παραπάνω ισχυρισμός,εκτελέστηκαν λίγες προσομοιώσεις κρατώντας σταθερό την παράμετρο **ws=4** και αυξάνοντας το **dw**.Τα αποτελέσματα παρουσιάζονται παρακάτω



Επιβεβαιώνεται ο ισχυρισμός που έγινε παραπάνω,καθώς βλέπουμε πως η μετρική IPC αυξάνεται μέχρις ότου **DS = WS** και στη συνέχεια παραμένει σταθερή

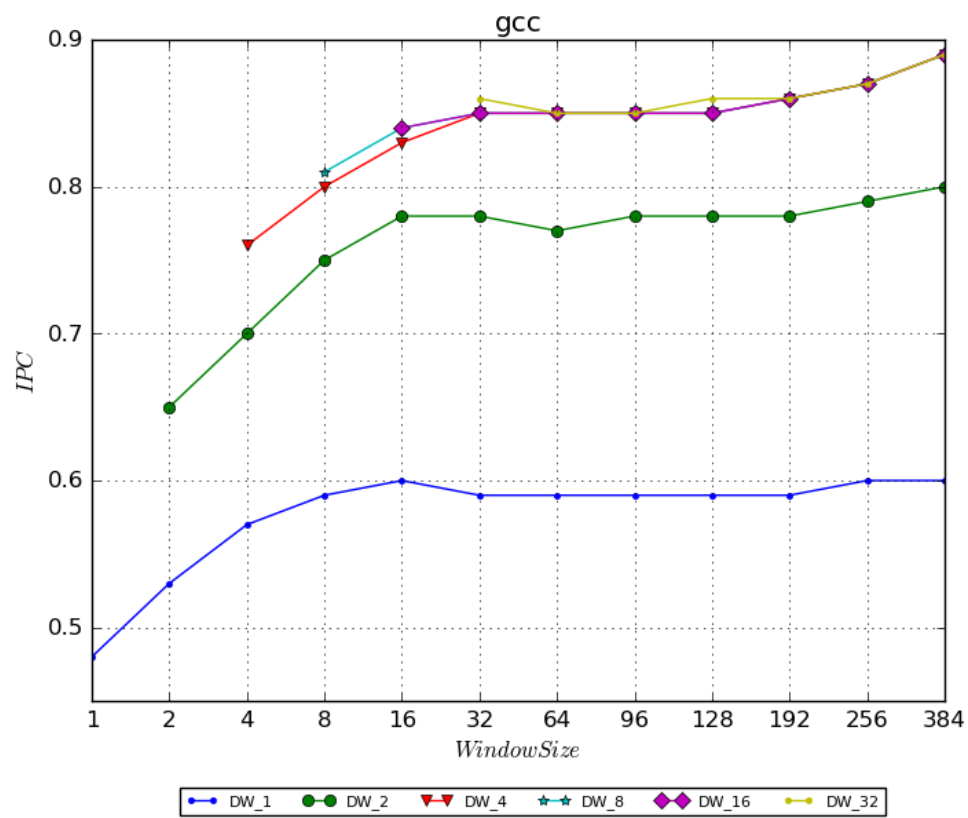
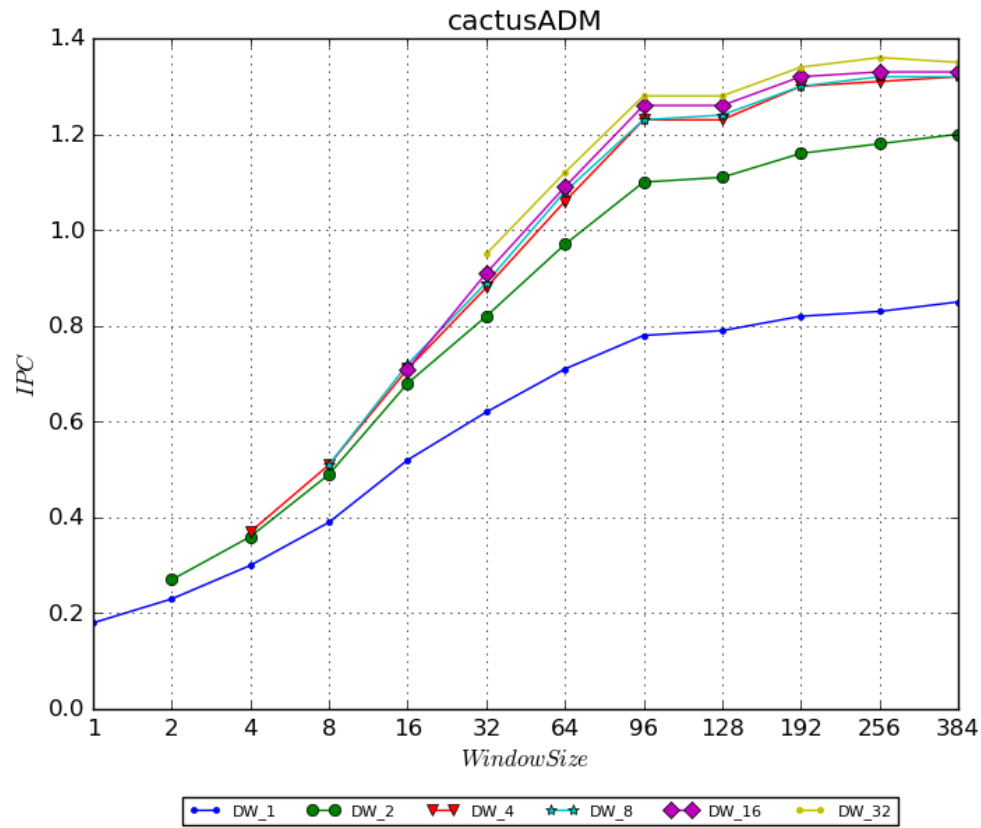
ΕΡΩΤΗΜΑ 2

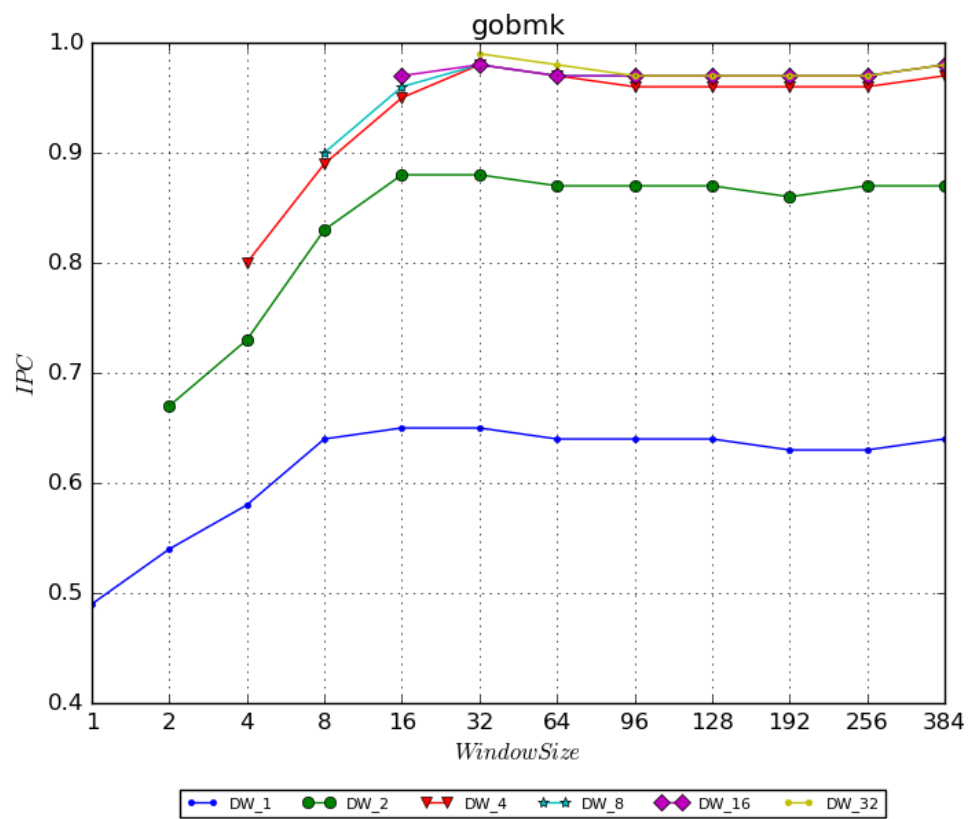
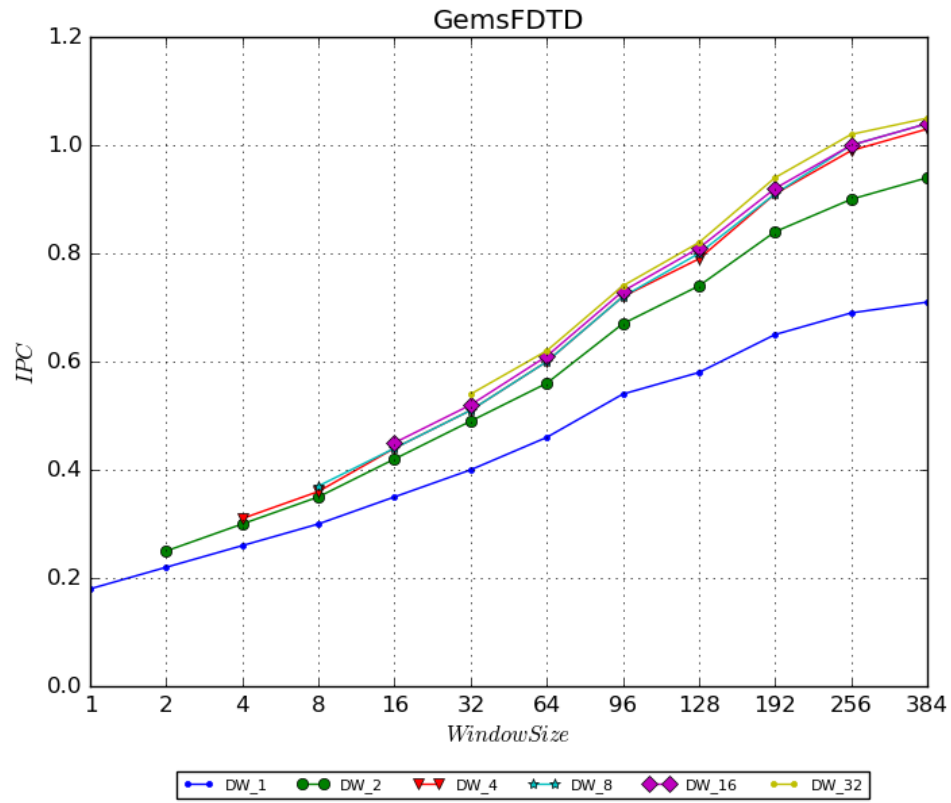
Εκφώνηση

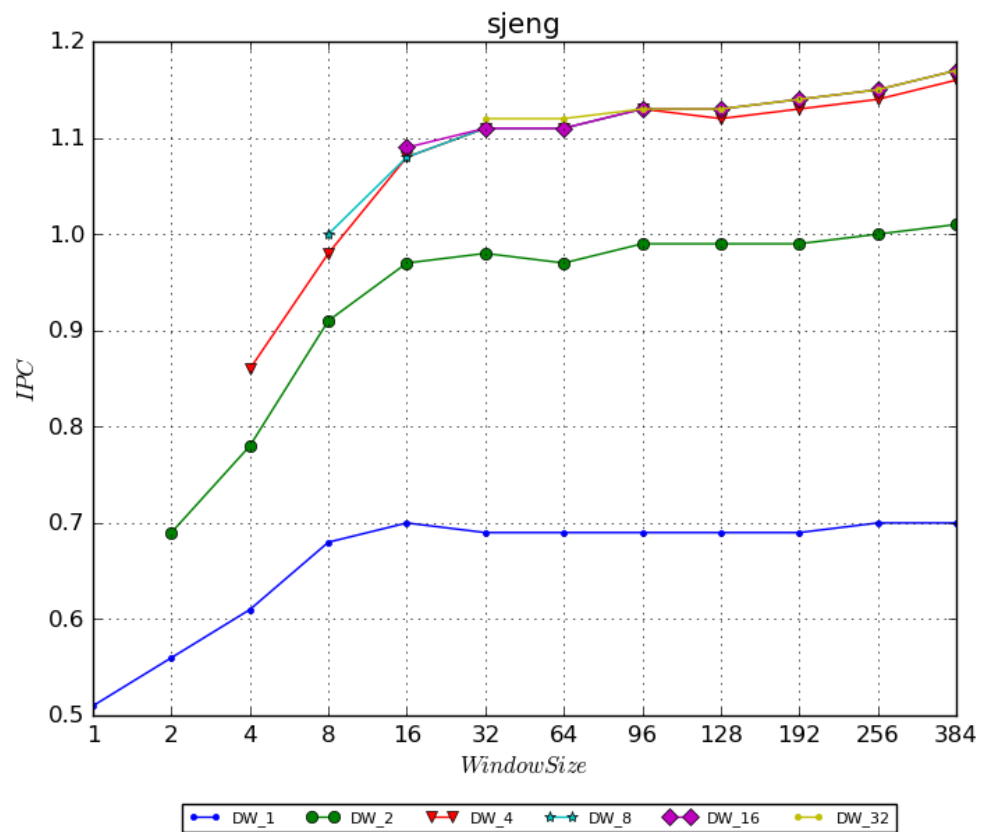
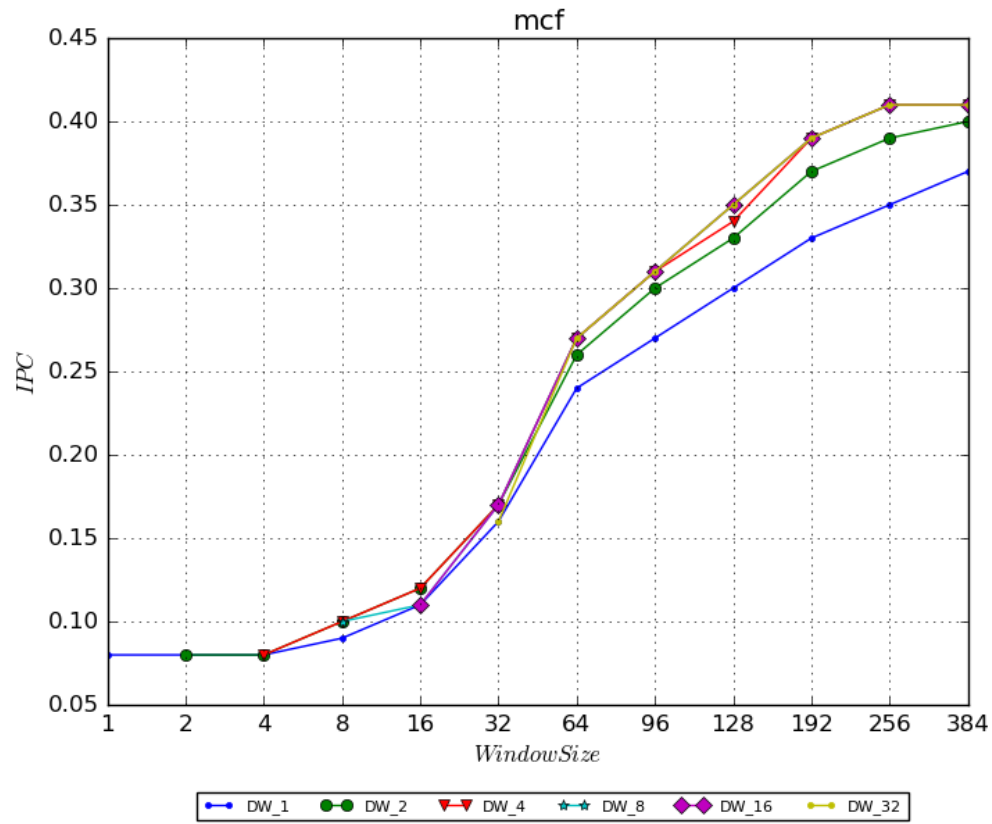
Πώς επηρεάζει η κάθε παράμετρος την απόδοση του επεξεργαστή; Σε ποιά συμπεράσματα μπορείτε να καταλήξετε ως προς το σχεδιασμό ενός superscalar, out-of-order επεξεργαστή;

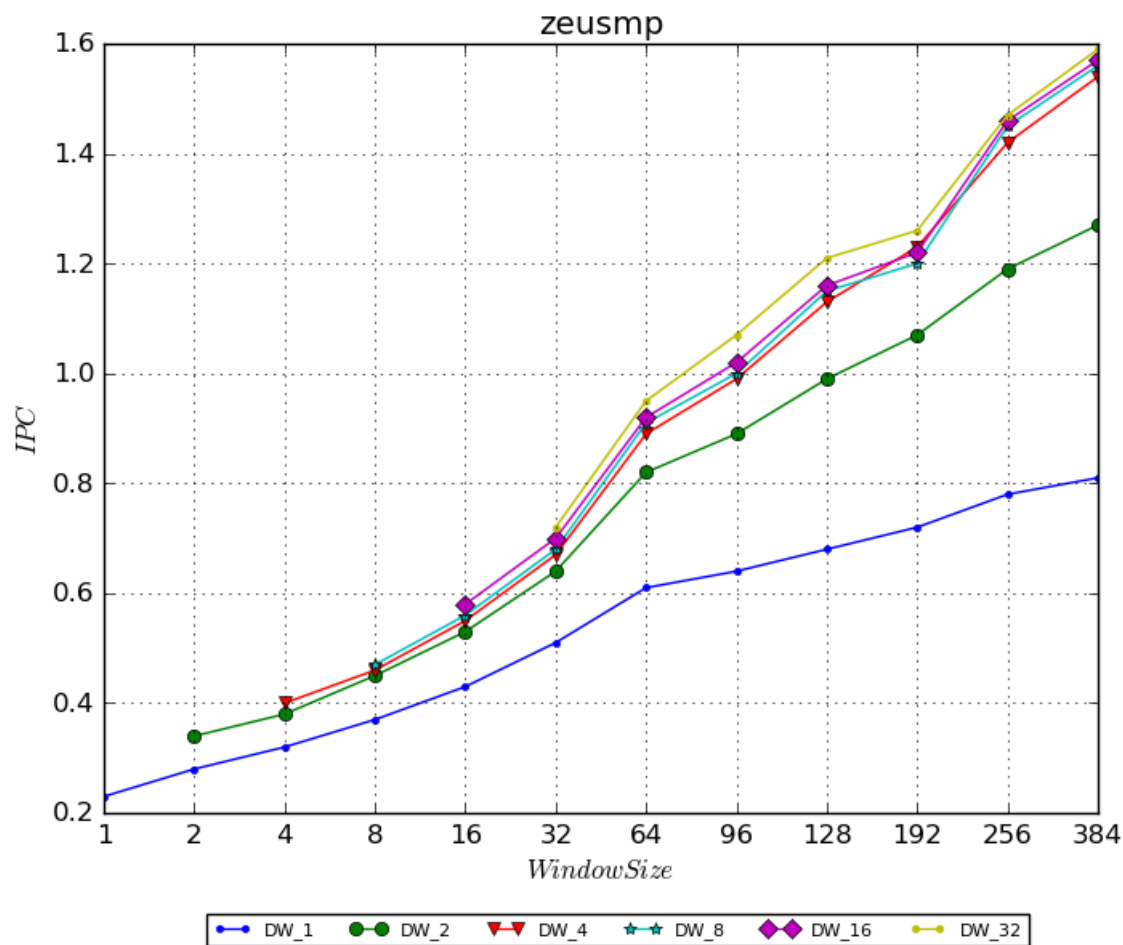
Λύση

Χρησιμοποιώντας τα εργαλεία που μας δώθηκαν για το σχεδιασμό των γραφημάτων από τα αποτελέσματα των benchmarks, σχεδιάστηκαν οι ακόλουθες γραφικές παραστάσεις.Φυσικά έγιναν οι κατάλληλες τροποποιήσεις στον κώδικα,απαραίτητες για την ορθή λειτουργία του.Επίσης,χρησιμοποιήθηκε ένα απλό bash script που καλεί το δοθέν αρχείο python ,δίνοντάς του κατάλληλα ορίσματα.Στη βιβλιοθήκη matplotlib επιλέχθηκε το classic theme για το design των γραφημάτων.









Σχολιασμός αποτελεσμάτων:

Καθώς αυξάνεται το μέγεθος DW παρατηρείται εντυπωσιακή αύξηση του IPC από την τιμή 1 στην τιμή 2. Λίγο μικρότερη (αλλά ακόμα μεγάλη) αύξηση βλέπουμε κατά την αύξηση από DW = 2 σε DW = 4. Από εκείνη την τιμή και έπειτα η αύξηση στην απόδοση είναι από σχεδόν έως και τελείως μηδαμινή, με μικρο-διαφορές ανάλογα το benchmark.

Κατά την αύξηση του μεγέθους WS, παρατηρείται στα περισσότερα benchmarks συνεχής αύξηση της απόδοσης. Σε πολλά από αυτά, η απόδοση αυξάνεται μέχρι WS = 8 ή WS = 16 και μετά παραμένει σταθερή, παρά την αύξηση του WS. Προφανώς σε αυτά τα benchmarks υπάρχει περιορισμένος αριθμός ανεξάρτητων εντολών, λόγω εξαρτήσεων.

Σε κάποια benchmarks όπως το zeusmp, το mcf και το GemsFTD ίσως υπάρχουν αρκετά long-latency events τα οποία είναι ο λόγος που δε παρατηρείται αύξηση στο IPC κατά την αύξηση του DW. Όταν συμβαίνει ένα βαθύ miss (πχ από L2 cache στην κύρια μνήμη), η καθυστέρηση είναι μεγάλη (της τάξης των εκατοντάδων κύκλων). Πολλά τέτοια γεγονότα περιέχουν τον κίνδυνο να γεμίσει ο ROB και να ανασταλλεί η λειτουργία του issue και κατ'επέκταση του commit^[1]. Καθώς όμως αυξάνεται το μέγεθος του ROB (WS) αυτά τα events γίνονται overlap, πράγμα που μας οδηγεί σε υψηλότερες τιμές IPC.

Συνοψίζοντας, η αύξηση του DW πάνω από 4 επέφερε αμελητέα αύξηση απόδοσης. Η αύξηση WS σε κάποια benchmarks επέφερε αύξηση του IPC μέχρι που παρατηρήθηκε ύπαρξη “πάνω φράγματος”. Στα περισσότερα όμως benchmarks η συνεχής αύξηση του WS επέφερε βελτίωση του IPC. Συμπερασματικά, δεν έχει νόημα η αύξηση του DW από κάποια τιμή και πάνω, ενώ ένας μεγαλύτερος ROB είναι πάντοτε επιθυμητός.

Σχεδιάζοντας έναν superscalar, out-of-order execution επεξεργαστή λαμβάνοντας υπ' όψη τα παραπάνω δεδομένα, επιλέγουμε **DW = 4** και **WS** όσο το δυνατόν μεγαλύτερο, δηλαδή 384.

ΕΡΩΤΗΜΑ 3

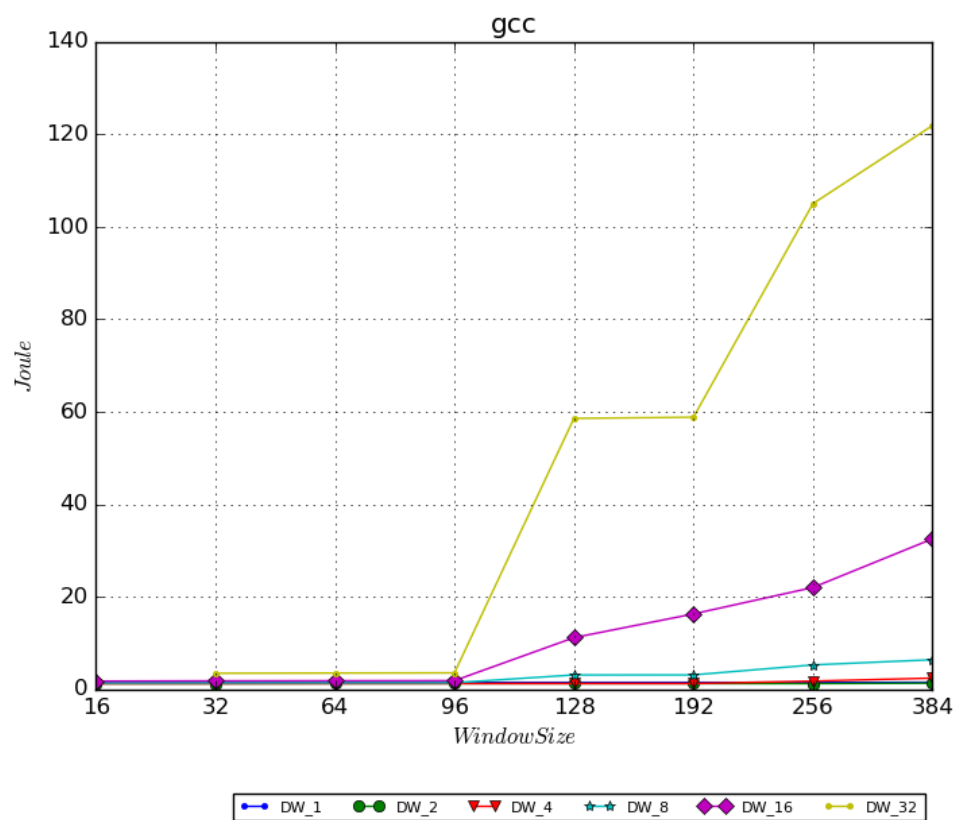
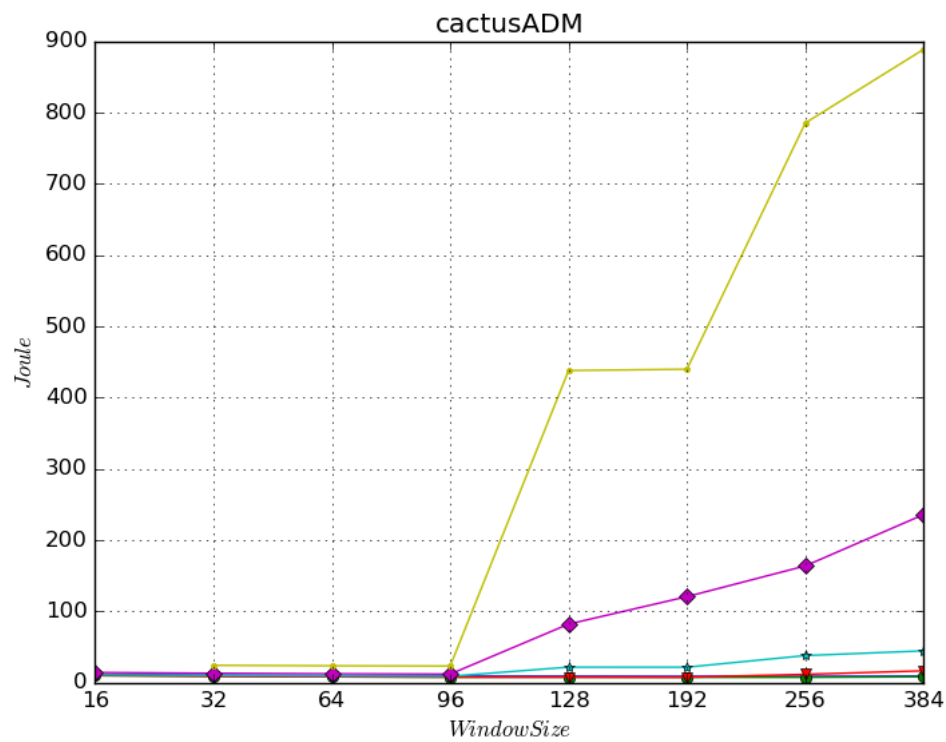
Εκφώνηση

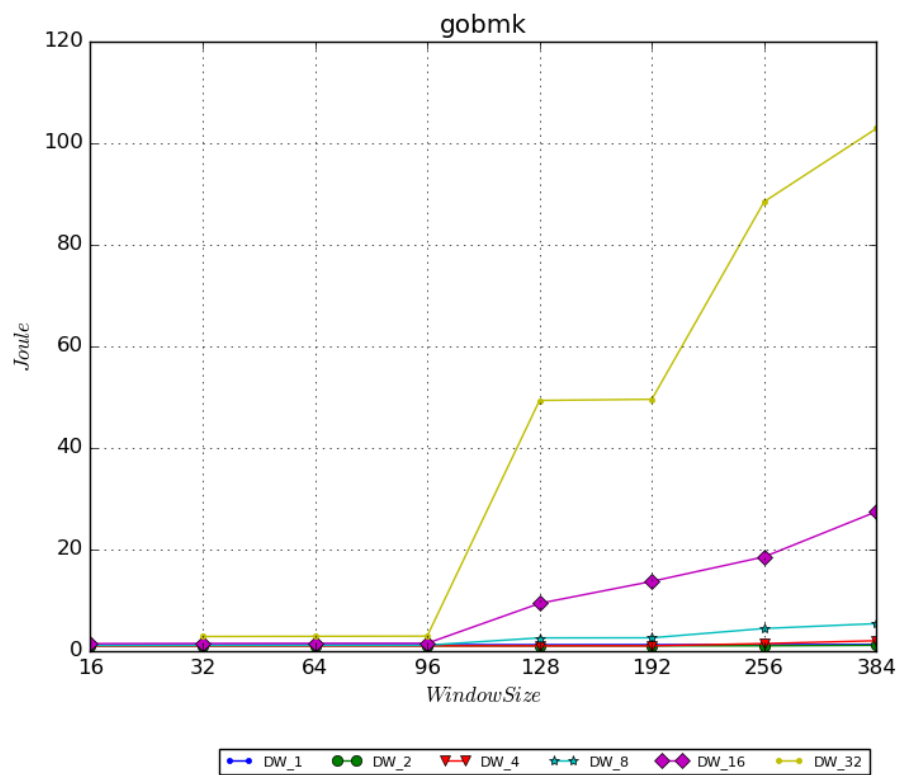
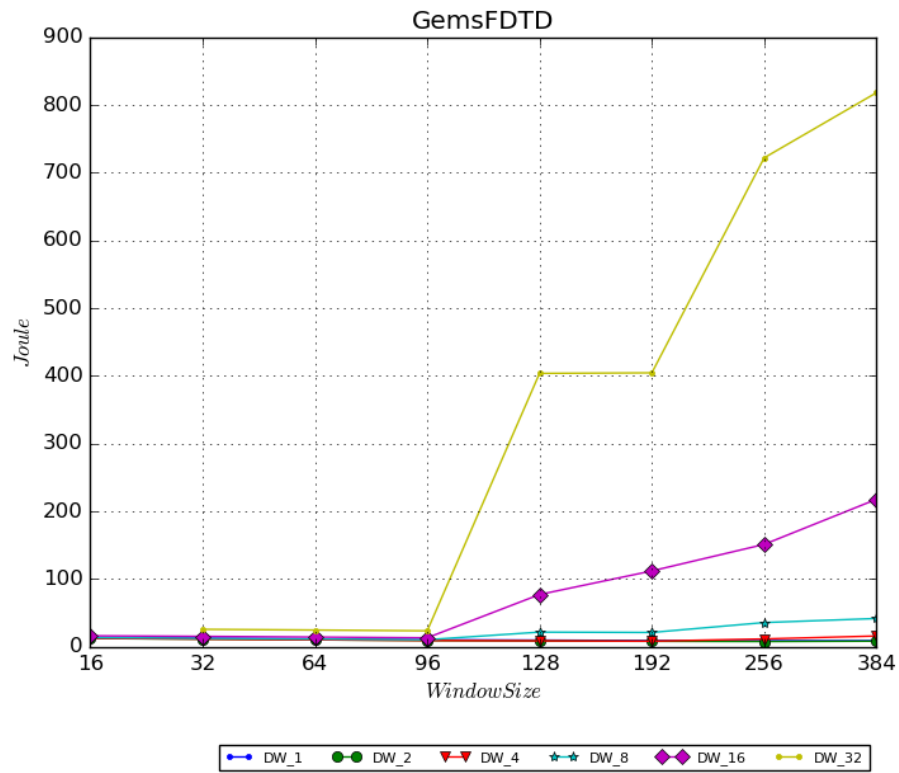
Πώς επηρεάζει η κάθε παράμετρος την κατανάλωση ενέργειας και το μέγεθος του τσιπ;

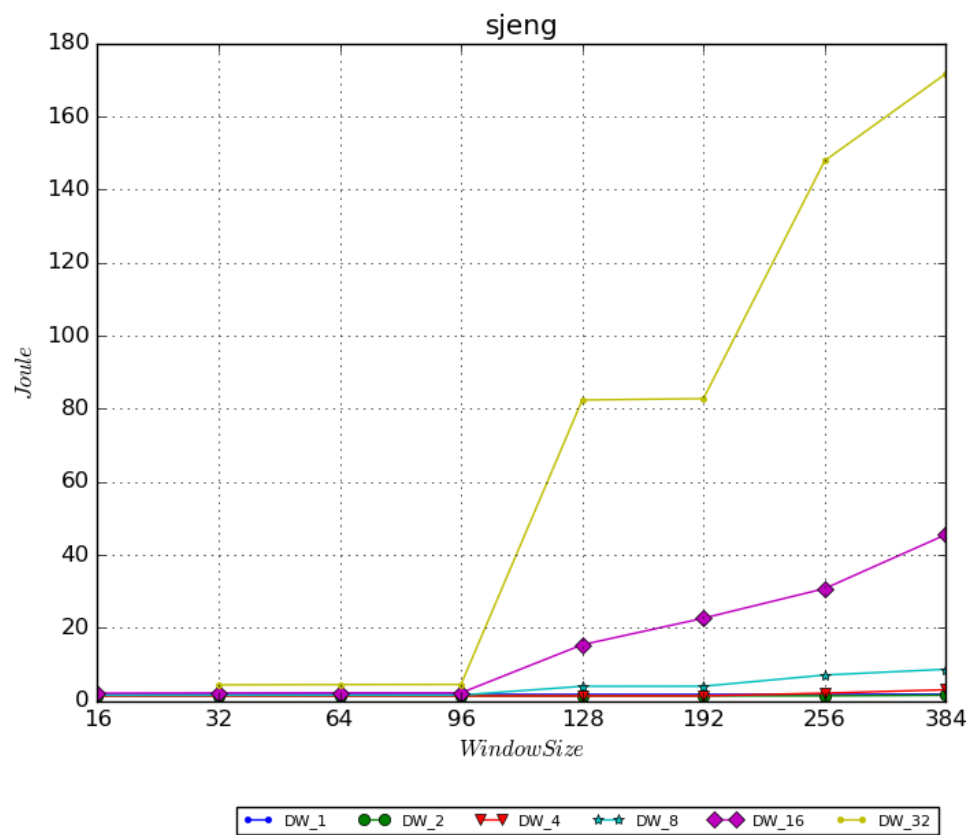
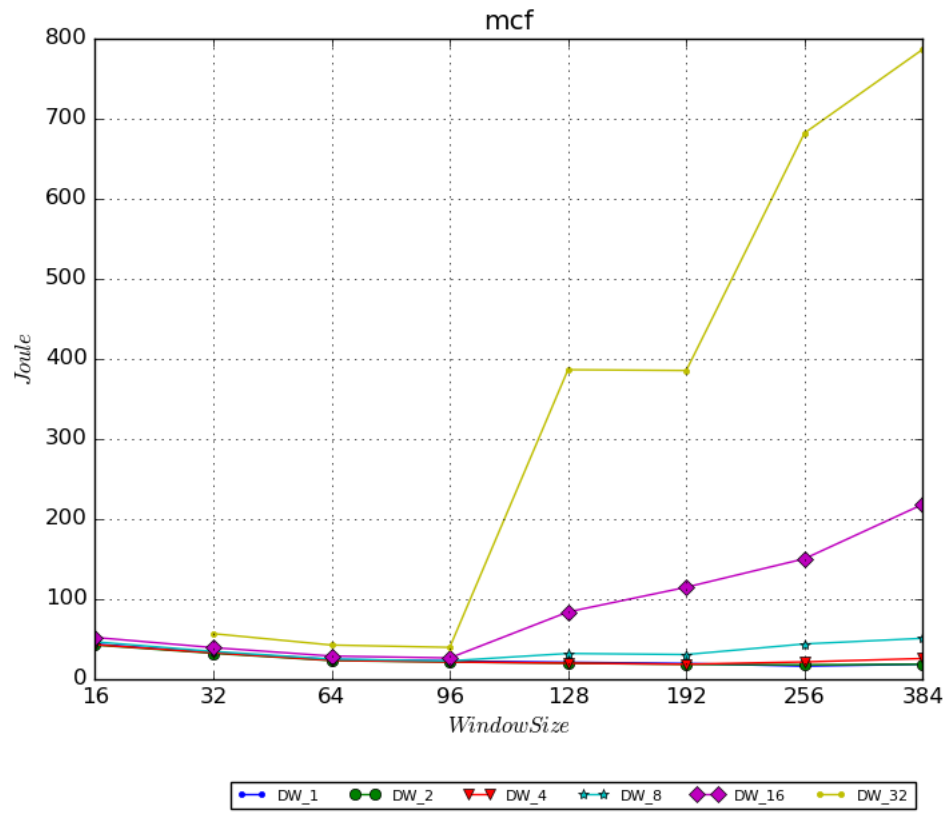
Λύση

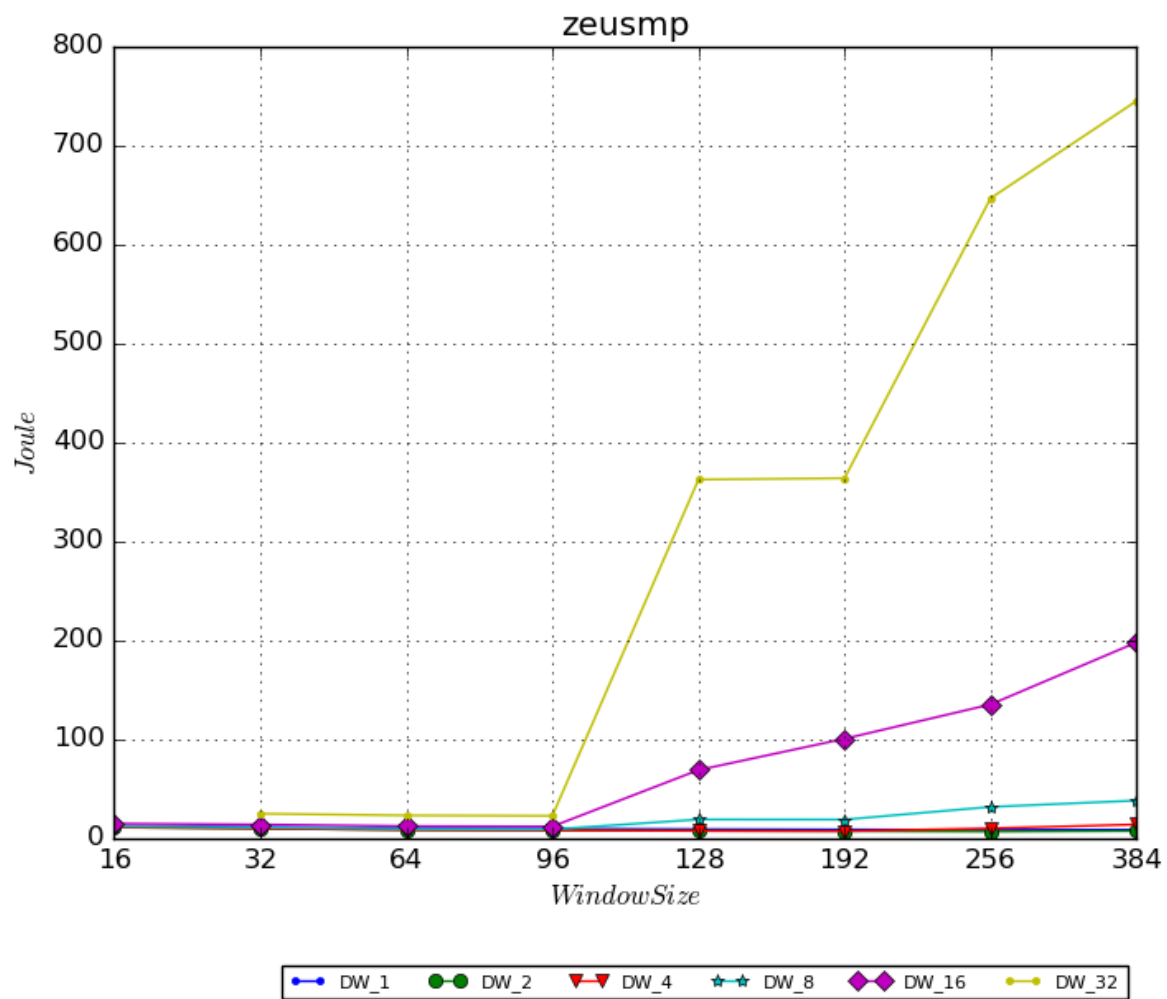
Αρχικά παρουσιάζονται τα διαγράμματα κατανάλωσης ενέργειας, EDP^2 και το μέγεθος του τσιπ συναρτήσει των παραμέτρων dw, ws .

Κατανάλωση ενέργειας (joule) :

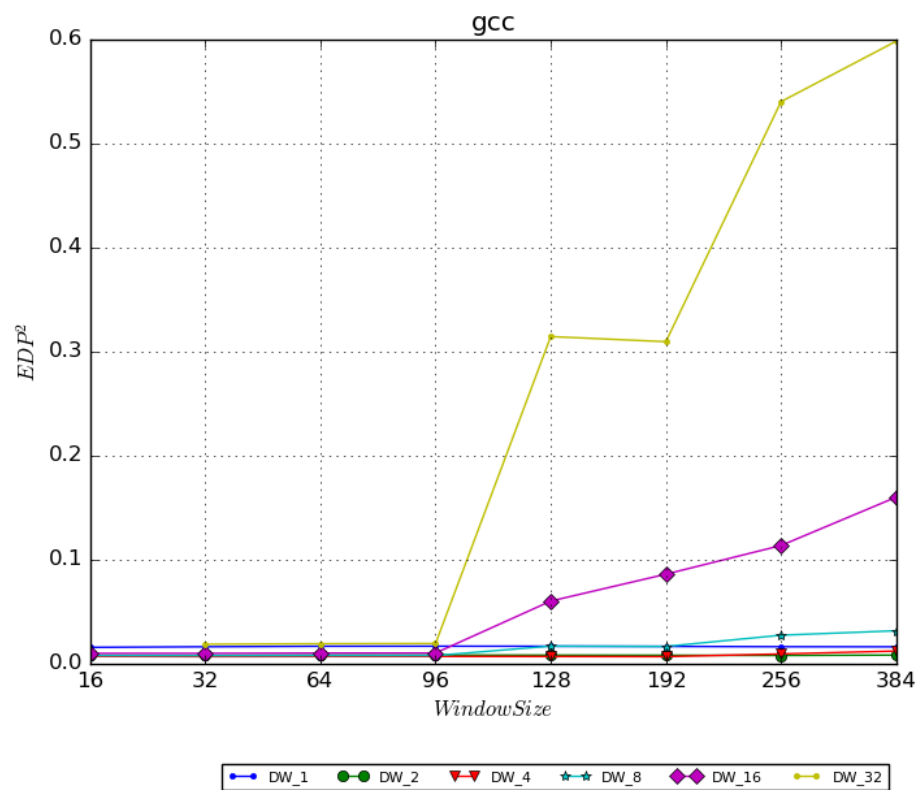
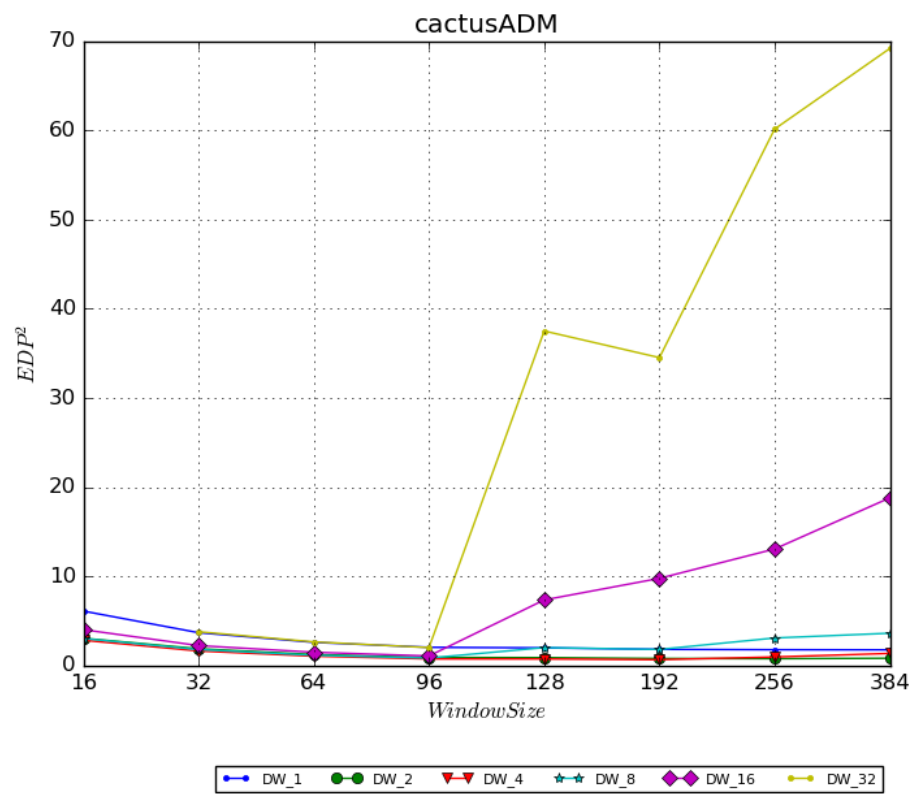


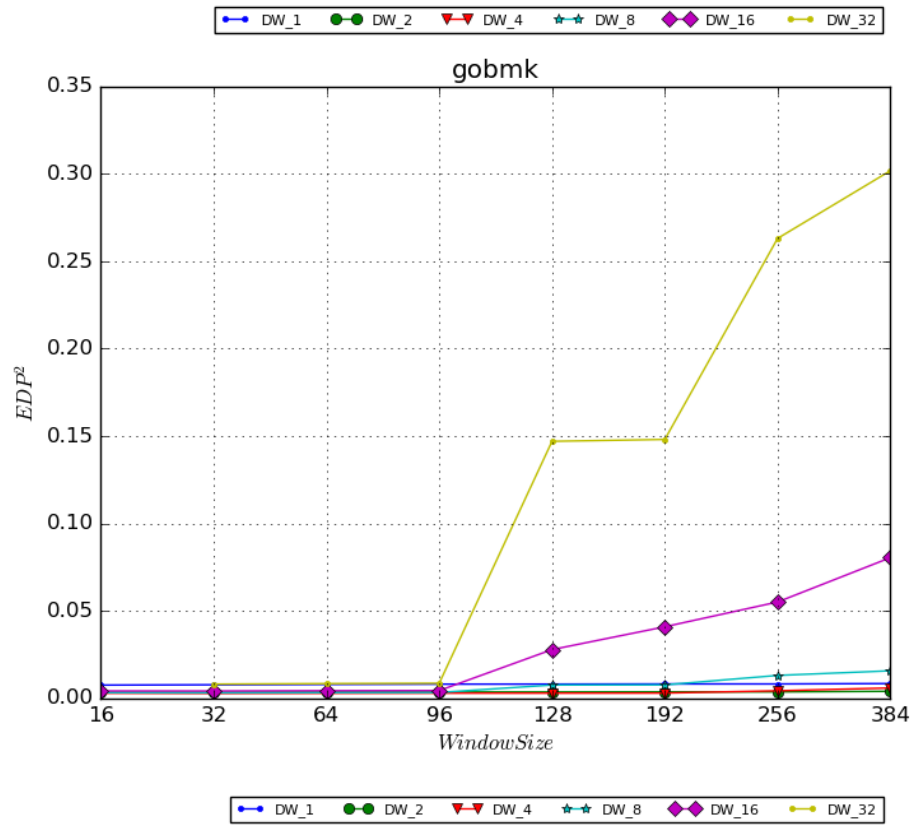
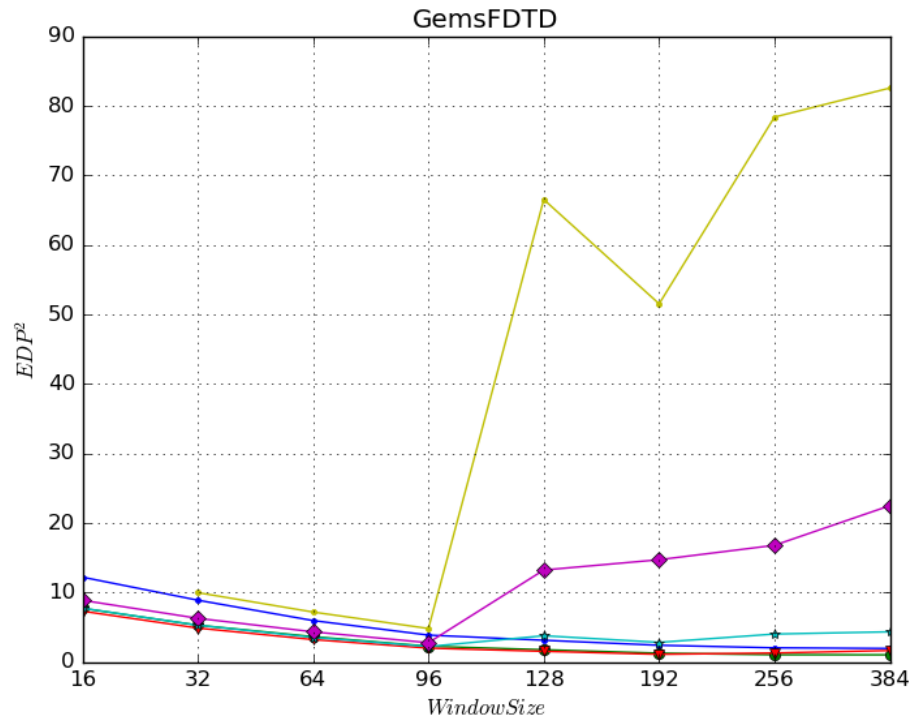


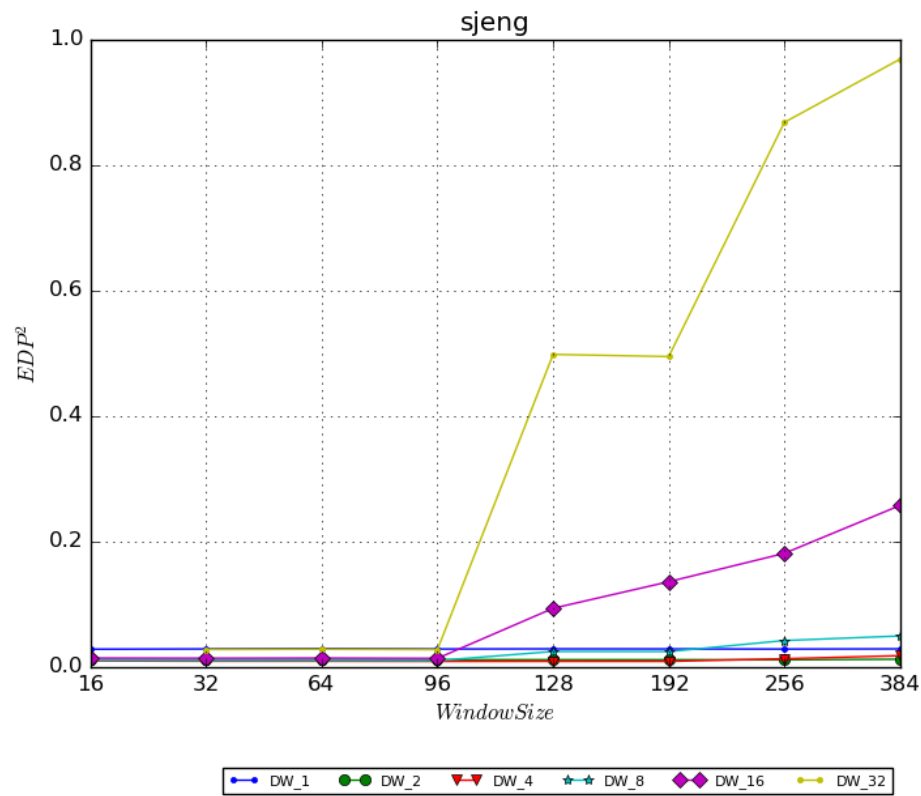
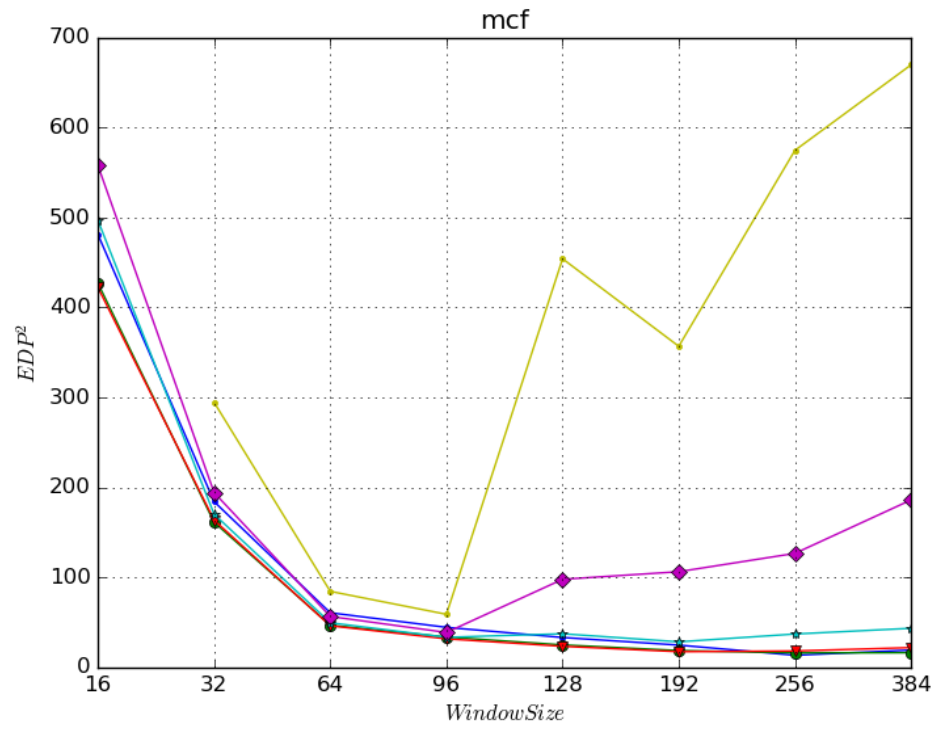


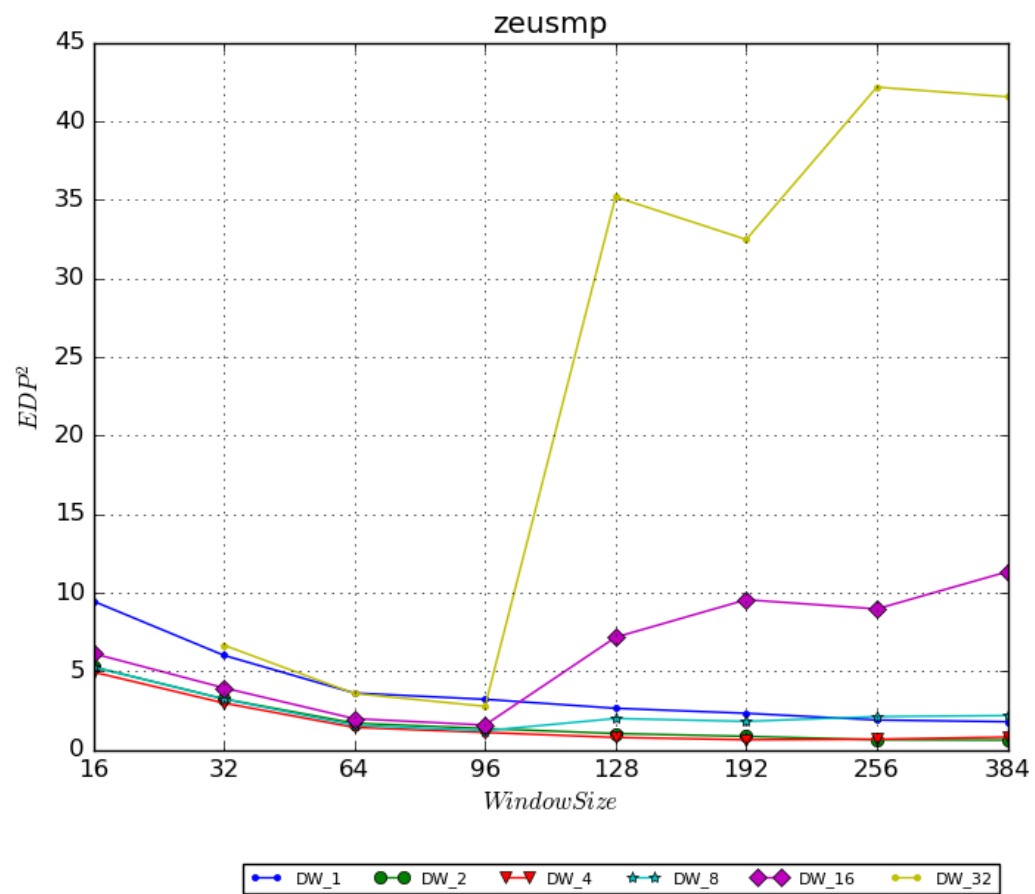


EDP² (Joule * sec²)

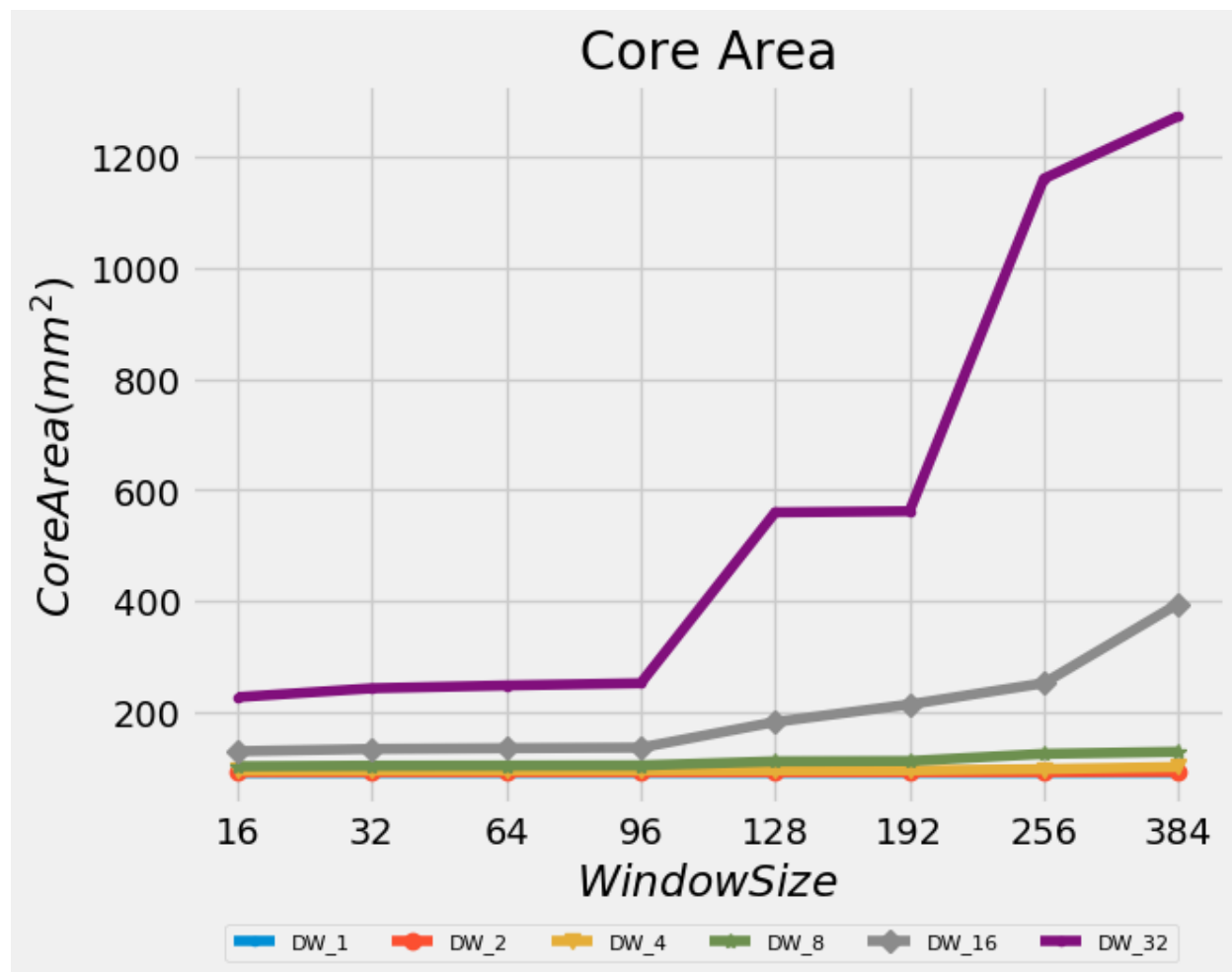








Μέγεθος του Τσιπ



Σχολιασμός αποτελεσμάτων

Τα διαγράμματα που αφορούν ενεργειακή κατανάλωση δείχνουν να έχουν παρεμφερή συμπεριφορά κατά την αλλαγή των παραμέτρων που εξετάζουμε. Η ενεργειακή κατανάλωση .Η κλιμακοσιμότητα της ενεργειακής κατανάλωσης δηλαδή σε συνάρτηση με τα ws,dw είναι περίπου το ίδιο για όλα τα benchmarks (παρ'οτι διαφέρουν max-min τιμές). Από αυτήν την παρατήρηση εξάγεται το συμπέρασμα πως οι 2 προαναφερθείσες παράμετροι είναι άμεσα συνδεδεμένες με την ενεργειακή κατανάλωση (πχ αύξηση σε κάποιες τιμές συνεπάγεται αύξηση στα καταναλισκόμενα Joule).

Σχετικά με το EDP², παρατηρούνται διαφοροποιήσεις σε κάθε benchmark, διότι το καθένα έχει εκτελεστεί για διαφορετικό χρόνο. Χρησιμοποιήθηκε το EDP² για να δοθεί έμφαση σε αυτό το γεγονός. Είναι πιθανό πως εάν οι προσομοιώσεις είχαν

εκτελεστεί για τον ίδιο χρόνο μεταξύ όλων των benchmarks, οι γραφικές να παρουσιάζαν παρεμφερείς τάσεις/μοτίβα με τα ενεργειακά γραφήματα.

Το μέγεθος του τσιπ μεταβάλλεται σε αντιστοιχία με τις μεταβολές που παρατηρούνται στα γραφήματα ενεργειακής κατανάλωσης.

Η τιμές $dw = 32$ και $dw = 16$ κρίνονται ασύμφωρες και απορρίπτονται, αφού παραρείται σημαντική αύξηση στην ενεργειακή κατανάλωση καθώς και στο μέγεθος του απαιτούμενου τσιπ, ενώ όπως είδαμε στο ερώτημα 2 οι βελτιώσεις στην απόδοση είναι αμελητέες. Η τιμή για το dw που επιλέξαμε προηγουμένως ($dw = 4$) παραμένει βέλτιστη, καθώς παρουσιάζει πολύ μικρή αύξηση στις απαιτήσεις ενέργειας και το μέγεθος πυρήνα, ενώ μας προσφέρει την ίδια απόδοση με τις τιμές 16 και 32.

ΕΡΩΤΗΜΑ 4

Εκφώνηση

Βρείτε τα αντίστοιχα χαρακτηριστικά (`dispatch_width`, `window_size`) για τον επεξεργαστή του προσωπικού σας υπολογιστή ή για κάποιον από τους σύγχρονους επεξεργαστές (π.χ. Broadwell, Skylake, Kabylake). Δικαιολογούνται οι τιμές που επιλέξαντε σε αυτά τα συστήματα οι αρχιτέκτονες με βάση τις προσομοιώσεις που εκτελέσατε και τα συμπεράσματα στα οποία καταλήξατε; Θα είχε νόημα να ήταν διαφορετικές (π.χ. μεγαλύτερο `window_size`); Για ποιό λόγο πιστεύετε δεν κάνανε κάποια άλλη επιλογή;

Απάντηση

Ο επεξεργαστής μου είναι ο Intel® Core™ i3-3120M, ο οποίος βασίζεται στην αρχιτεκτονική Ivy Bridge, η οποία βασίζεται στην αρχιτεκτονική Sandy Bridge. Η πιο βασική διαφορά είναι το μέγεθος των τρανζίστορ, καθώς στην πρώτη είναι στα 22nm, ενώ η 2η ήταν στα 32nm. Περισσότερες πληροφορίες βλέπουμε με την

εντολή lscpu.

```
File Edit Tabs Help
giorgos@nikos:~$ lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                4
On-line CPU(s) list:   0-3
Thread(s) per core:    2
Core(s) per socket:    2
Socket(s):              1
NUMA node(s):          1
Vendor ID:             GenuineIntel
CPU family:            6
Model:                 58
Model name:            Intel(R) Core(TM) i3-3120M CPU @ 2.50GHz
Stepping:              9
CPU MHz:               1197.370
CPU max MHz:           2500.0000
CPU min MHz:           1200.0000
BogoMIPS:              4988.83
Virtualization:        VT-x
L1d cache:             32K
L1i cache:             32K
L2 cache:              256K
L3 cache:              3072K
NUMA node0 CPU(s):     0-3
Flags:                 fpu vme de pse tsc msr pae mce cx8 apic sep mtrr
t_tsc arch_perfmon pebs bts rep_good nopl xtopology nonstop_tsc cpuid
x2apic popcnt tsc_deadline_timer xsave avx f16c lahf_lm cpuid_fault ep
arat pln pts md_clear flush_l1d
```


Δυστυχώς δεν κατάφερα να βρω ακριβής πληροφορίες για το μέγεθος dispatch width για την αρχιτεκτονική του επεξεργαστή μου,παρά μόνο για την αρχιτεκτονική που αντικατέστησε και αυτή που την διαδέχτηκε (Sandy Bridge και Haswell αντίστοιχα).Συνεπώς θα ασχοληθώ με τη 4η γενιά της Intel (αρχιτεκτονική Haswell),καθώς βρίσκεται και αυτή στα 22nm και ο i3-4120M είναι πολύ κοντά από πλευράς χαρακτηριστικών με τον δικό μου (2 φυσικούς πυρήνες και 4 threads με hyper threading,κοντινή συχνότητα ρολογιού κ.α.).Λογικά, τα συμπεράσματα για τον i3-4120M (δηλαδή την αρχιτεκτονική Haswell)πρέπει να είναι τα ίδια ή κοντινά και για τον δικό μου.

Όπως διαβάζουμε στο συνέδριο hotchips κατά την [παρουσίαση της αρχιτεκτονικής Haswell](#), το dispatch width είναι 8, ενώ το window size είναι 192.

Haswell Buffer Sizes

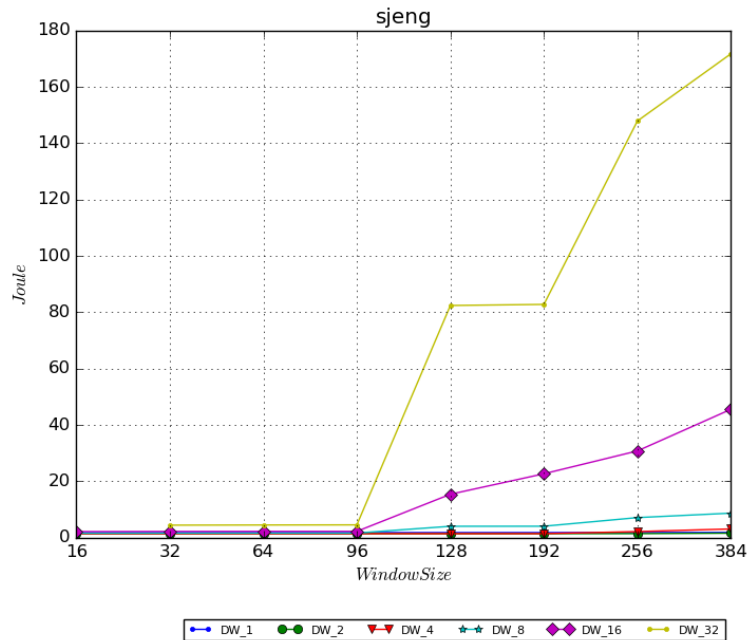
Extract more parallelism in every generation

	Nehalem	Sandy Bridge	Haswell
Out-of-order Window	128	168	192



Σύμφωνα με τα αποτελέσματα από τις προσομοιώσεις της παρούσας άσκησης, το window size και το dispatch width έχουν λογικές τιμές και συμβαδίζουν με τα αποτελέσματά μας. Από τα δικά μας νούμερα φαίνεται πως το dispatch width = 8 είναι μια πολύ λογική επιλογή, αφού περαιτέρω αύξηση κρίνεται ως ασύμφορη, ενώ το window size μπορεί να αυξηθεί τουλάχιστον μέχρι 384 entries, επιφέροντας αυξημένη απόδοση και μάλιστα με σχεδόν μηδαμινή αύξηση στην καταναλισκόμενη ενέργεια και το μέγεθος του τσιπ.

Οι λόγοι για τους οποίους οι αρχιτέκτονες παρέμειναν σε αυτό το μέγεθος ROB είναι πιθανώς οικονομικοί παράγοντες (μείωση του κόστους παραγωγής). Επιπλέον, για $dw = 8$ και $ws \geq 192$ παρατηρείται κάποια αύξηση στην ενέργεια που καταναλώνεται. Μια πιθανή εξήγηση για το μέγεθος είναι πως η αύξηση της συχνότητας ρολογιού του επεξεργαστή επέφερε σημαντικότερες βελτιώσεις στη συνολική απόδοση του συστήματος από ένα μεγαλύτερο μέγεθος του ROB, και η μικρή αύξηση της θερμοκρασίας (η αύξηση φαίνεται από τα γραφήματα ενέργειας) αποτελούσε ανασταλτικό παράγοντα σε αυτήν την βελτίωση. Συνεπώς το ws παρέμεινε στα 192, καθώς όπως φαίνεται σε όλα τα γραφήματα, το $ws = 192$ έχει την ίδια σχεδόν κατανάλωση και το ίδιο μέγεθος τσιπ με $ws = 128$.



Συνεπώς ίσως η τιμή $ws=192$ αποτελούσε τη “χρυσή τομή” σε σχέση με το κόστος παραγωγής, αλλά και σε σχέση με τα υπόλοιπα χαρακτηριστικά της αρχιτεκτονικής. Κλείνοντας, γίνεται σαφές για μία ακόμη φορά πως πάντα υπάρχει tradeoff σε οποιαδήποτε επιλογή και ένας αρχιτέκτονας πρέπει να λάβει υπόψη τον πολλούς και αντικρουόμενους παράγοντες.

Βιβλιογραφία

[1] S. Eyerman, L. Eeckhout, T. Karkhanis, and J. E. Smith, “A Mechanistic Performance Model for Superscalar Out-of-Order Processors,” *ACM Trans. Comput. Syst.*, vol. 27, no. 2, 2009, doi: 10.1145/1534909.1534910.