Βαμβακίδης Δημήτρης - 9501
Δημητρίου Μάριος - 9235

# Αρχιτεκτονική Υπολογιστών

## Εργαστήριο 1

## Ερώτημα 1

Ανοίγοντας το αρχείο starter_se.py στο κομμάτι main() βρήκαμε τις διαθέσιμες ρυθμίσεις για το συγκεκριμένο config του gem5 (το ίδιο επιτύχαμε και με εκτέλεση του command –h). Ενδεικτικά τρέξαμε το αρχείο hello με την εντολή που δόθηκε στο pdf του εργαστηρίου ($ ./build/ARM/gem5.opt -d hello_result configs/example/arm/starter_se.py --cpu="minor" "tests/test-progs/hello/bin/arm/linux/hello") και με τη προσθήκη arguments όπως --mem-size, --mem-channels, --cpu-freq, --num-cores (με τον αντίστοιχο αριθμό duplicates για το πρόγραμμα “hello world”). Στο συγκεκριμένο config ο τύπος της cache δεν μπορεί να αλλάξει από arguments παρά μόνο αν τροποποιηθεί από το config file.

Επιπλέον τα cpu types που ήταν διαθέσιμα για αυτό το αρχείο ήταν atomic, minor και hpi, όπου το atomic ήταν το default.

```
rio@rio-Laptop ~/my_gem5> ./build/ARM/gem5.opt -d hello_result configs/example/arm/starter_se.py --cpu-freq=2GHz --mem-channels=4 --mem-size=6GB
"tests/test-progs/hello/bin/arm/linux/hello"
gem5 Simulator System.  http://gem5.org
gem5 is copyrighted software; use the --copyright option for details.

gem5 compiled Nov  8 2019 17:52:44
gem5 started Nov 19 2019 20:46:57
gem5 executing on rio-Laptop, pid 3726
command line: ./build/ARM/gem5.opt -d hello_result configs/example/arm/starter_se.py --cpu-freq=2GHz --mem-channels=4 --mem-size=6GB tests/test-p
rogs/hello/bin/arm/linux/hello

info: 1. command and arguments: ['tests/test-progs/hello/bin/arm/linux/hello']
Global frequency set at 1000000000000 ticks per second
warn: DRAM device capacity (8192 Mbytes) does not match the address range assigned (2048 Mbytes)
warn: DRAM device capacity (8192 Mbytes) does not match the address range assigned (2048 Mbytes)
warn: DRAM device capacity (8192 Mbytes) does not match the address range assigned (2048 Mbytes)
warn: DRAM device capacity (8192 Mbytes) does not match the address range assigned (2048 Mbytes)
0: system.remote_gdb: listening for remote gdb on port 7000
info: Entering event queue @ 0.  Starting simulation...
Hello world!
exiting with last active thread context  @  2915000
rio@rio-Laptop ~/my_gem5>
```

**Terminal commands and results**

```
system.clk_domain.clock                                 1000
system.cpu_cluster.voltage_domain.voltage           1.200000
system.cpu_cluster.clk_domain.clock                      500
```

**Αποτελέσματα αρχείου stats.txt**

Παραπάνω φαίνεται για ενδεικτικές τιμές του cpu frequency, mem-size και mem-channels ότι οι τιμές οι οποίες δόθηκαν ως command arguments πράγματι είναι αυτές που προσομοιώθηκαν όπως φαίνεται και από το ενδεικτικό κομμάτι του αρχείου stats.txt . Ανάλογα με τις παραμέτρους και ιδιαίτερα με το cpu frequency και mem channels επηρεαζότανε και ο χρόνος εκτέλεσης του simulation.

Βαμβακίδης Δημήτρης - 9501
Δημητρίου Μάριος - 9235

## Ερώτημα 2

```
[system]
type=System
children=clk_domain cpu_cluster dvfs_handler mem_ctrls0 mem_ctrls1 mem_ctrls2 mem_ctrls3 membus voltage_domain
boot_osflags=a
cache_line_size=64
eventq_index=0
exit_on_work_items=false
init_param=0
kernel=
kernel_addr_check=true
kernel_extras=
load_addr_mask=18446744073709551615
load_offset=0
mem_mode=atomic
mem_ranges=0:6442450943
memories=system.mem_ctrls0 system.mem_ctrls1 system.mem_ctrls2 system.mem_ctrls3
mmap_using_noreserve=false
multi_thread=false
num_work_ids=16
readfile=
redirect_paths=
symbolfile=
thermal_components=
thermal_model=Null
work_begin_ckpt_count=0
work_begin_cpu_id_exit=-1
work_begin_exit_count=0
work_cpus_ckpt_count=0
work_end_ckpt_count=0
work_end_exit_count=0
work_item_id=-1
system_port=system.membus.slave[0]
```

**System στο αρχείο config.ini**

```
[system.clk_domain]
type=SrcClockDomain
clock=1000
domain_id=-1
eventq_index=0
init_perf_level=0
voltage_domain=system.voltage_domain

[system.cpu_cluster]
type=SubSystem
children=clk_domain cpus voltage_domain
eventq_index=0
thermal_domain=Null

[system.cpu_cluster.clk_domain]
type=SrcClockDomain
clock=500
domain_id=-1
eventq_index=0
init_perf_level=0
voltage_domain=system.cpu_cluster.voltage_domain
```

**CPU και System Clock**

Παραπάνω φαίνεται όπως και στο αρχείο stats.txt τα 4 channels της μνήμης, η χωρητικότητα των 6GB της μνήμης (6442450943 bytes στη πρώτη φωτογραφία) καθώς και τα cpu και system clock στη δεύτερη φωτογραφία. Επιβεβαιώνεται δηλαδή ότι προσομοιώθηκαν οι τιμές που δόθηκαν σαν ορίσματα.

Βαμβακίδης Δημήτρης - 9501
Δημητρίου Μάριος - 9235

**Ερώτημα 3**

***MinorCpuModel:*** Το συγκεκριμένο μοντέλο αποτελεί τη βάση για επέκταση από το χρήστη με σκοπό τη μοντελοποίηση πολυπλοκότερων μοντέλων cpu. Το μοντέλο ακολουθεί το πρότυπο του pipelining και χωρίζεται στα εξής στάδια: fetch1 – fetch2 – decode – execute.

***Fetch1-Fetch2:*** Στα δύο αυτά στάδια γίνεται το fetching των εντολών από την i-cache σε μορφή lines of cache με δυνατότητα prefetching, ο διαχωρισμός τους σε instructions και η προσκόμισή τους σε μορφή vectors στο στάδιο decode. Ακόμα σε αυτά τα στάδια συναντιούνται και ο BTB(branch target buffer) – fetch1 και ο BP(branch predictor)- fetch2, που είναι υπεύθυνοι αντίστοιχα για τον άμεσο προσδιορισμό του PC για ένα Taken branch και την διαχείριση των καταχωρήσεων αυτού. Το αν ένα branch είναι πραγματικά taken η όχι στέλνεται στις δύο αυτές μονάδες από το execution stage αφού γίνει το evaluation αυτού.

***Decode:*** Όπως και το όνομα του σταδίου υπονοεί στο συγκεκριμένο στάδιο γίνεται αποκωδικοποίηση των εντολών και έπειτα αυτές στέλνονται στο στάδιο execute. Το συγκεκριμένο στάδιο παρέχει τη δυνατότητα αποστολής vector εντολών με σκοπό το in-order issue πολλών εντολών ανά clock cycle.

***Execution:*** Το συγκεκριμένο στάδιο χωρίζεται σε δύο υποστάδια: issue, commit. Υπάρχει ακόμα και το στάδιο Advance που διαχειρίζεται το stalling η όχι των pipelined functional units. Πιο αναλυτικά: οι εντολές παιρνούν στο στάδιο issue εφόσον δεν υπάρχουν dependencies που να αποτρέπουν την εκτέλεση τους(το συγκεκριμένο μοντέλο χρησιμοποιεί απλό scoreboarding και όχι και register renaming  οπότε απουσιάζουν οι reservation stations). Ταυτόχρονα οι εντολές εισάγονται σε ένα queue ώστε να διασφαλιστεί το in order commit (ολοκλήρωση εντολής και εγγραφή στα registers ή τη μνήμη) αυτών επιτρέποντας την out of order εκτέλεση τους. Σε περίπτωση που μια εντολή δεν μπορεί να κάνει commit τότε το στάδιο Advance κάνει stall το Pipeline στο οποίο βρίσκεται η εντολή. Εντολές όπως κενές ή exceptions μπορούν να κατευθυνθούν σε οποιαδήποτε functional unit.

Ειδικά για τις εντολές Load και Store υπάρχει στα functional units τους ένα ειδικό buffer που επιτρέπει την εξαγωγή των store από τον αρχικό Buffer εκτέλεσης memory εντολών με σκοπό την δυνατότητα εκτέλεσης επόμενων load εντολών που δεν αφορούν τη συγκεκριμένη θέση μνήμης (υποστήριξη non Blocking Cache).

Η επικοινωνία των σταδίων του pipeline του επεξεργαστή γίνεται μέσω buffers που είναι υπεύθυνοι για τη μεταφορά δεδομένων και την επιβολή stall του pipeline σε περίπτωση που κάποιο στάδιο είναι γεμάτο.

***SimpleCpuModel:*** Το εν λόγω μοντέλο cpu αποτελεί το απλούστερο μοντέλο cpu στον gem5 και αποτελείται από έναν απλό cpu που κάνει fetch και εκτελεί μια εντολή τη φορά, χωρίς pipelining ή χρήση οποιασδήποτε άλλης μεθόδου επιτάχυνσης της εκτέλεσης. Χωρίζεται σε δύο βασικές κατηγορίες(υπάρχει και τρίτη αλλά δεν αναφέρεται εδώ) που διαφέρουν ως προς την υλοποίηση της μνήμης cache.

***AtomicSimpleCpu:*** Το συγκεκριμένο μοντέλο δεν κάνει χρήση πραγματικού μοντέλου μνήμης cache παρά η προσπέλαση γίνεται πρακτικά άμεσα με προσέγγιση του χρόνου που θα χρειαζόταν η πραγματική προσπέλαση της μνήμης.

***TimingSimpleCpu:*** Εδώ με εξαίρεση το απλούστατο μοντέλο εκτέλεσης εντολών η μνήμη cache προσομοιώνεται κανονικά. Επιτρέπει ρεαλιστικό υπολογισμό των χρόνων προσπέλασης η αδυναμίας απόκρισης του υλικού(π.χ. απασχολημένη μνήμη).

Τα δύο παραπάνω μοντέλα χρησιμοποιούνται γενικά για τον έλεγχο σωστής λειτουργίας benchmarks και warm up της cache.

a.

```
rio@rio-Laptop ~/my_gem5> arm-linux-gnueabihf-gcc --static hello_result/testx.c -o hello_result/testX
rio@rio-Laptop ~/my_gem5> ./build/ARM/gem5.opt -d hello_result configs/example/se.py --cpu-type=TimingSimpleCPU --caches --l1i_size=64kB --l1d_si
ze=16kB  --cmd="hello_result/testX"
gem5 Simulator System.  http://gem5.org
gem5 is copyrighted software; use the --copyright option for details.

gem5 compiled Nov  8 2019 17:52:44
gem5 started Nov 19 2019 21:54:13
gem5 executing on rio-Laptop, pid 6623
command line: ./build/ARM/gem5.opt -d hello_result configs/example/se.py --cpu-type=TimingSimpleCPU --caches --l1i_size=64kB --l1d_size=16kB --cm
d=hello_result/testX

Global frequency set at 1000000000000 ticks per second
warn: DRAM device capacity (8192 Mbytes) does not match the address range assigned (512 Mbytes)
0: system.remote_gdb: listening for remote gdb on port 7000
**** REAL SIMULATION ****
info: Entering event queue @ 0.  Starting simulation...
info: Increasing stack size by one page.
A
Exiting @ tick 36362000 because exiting with last active thread context
rio@rio-Laptop ~/my_gem5>
```

**CPU Type = TimingSimpleCPU**

| | |
|---|---|
| final_tick | 36362000 |
| host_inst_rate | 300183 |
| host_mem_usage | 707296 |
| host_op_rate | 338412 |
| host_seconds | 0.03 |
| host_tick_rate | 1337606769 |
| sim_freq | 1000000000000 |
| sim_insts | 8102 |
| sim_ops | 9190 |
| sim_seconds | 0.000036 |
| sim_ticks | 36362000 |

**Αρχείο stats.txt από εκτέλεση cpu-type=TimingSimpleCPU**

```
rio@rio-Laptop ~/my_gem5> ./build/ARM/gem5.opt -d hello_result configs/example/se.py --cpu-type=MinorCPU --caches --l1i_size=64kB --l1d_size=16kB
  --cmd="hello_result/testX"
gem5 Simulator System.  http://gem5.org
gem5 is copyrighted software; use the --copyright option for details.

gem5 compiled Nov  8 2019 17:52:44
gem5 started Nov 19 2019 21:56:16
gem5 executing on rio-Laptop, pid 6666
command line: ./build/ARM/gem5.opt -d hello_result configs/example/se.py --cpu-type=MinorCPU --caches --l1i_size=64kB --l1d_size=16kB --cmd=hello
_result/testX

Global frequency set at 1000000000000 ticks per second
warn: DRAM device capacity (8192 Mbytes) does not match the address range assigned (512 Mbytes)
0: system.remote_gdb: listening for remote gdb on port 7000
**** REAL SIMULATION ****
info: Entering event queue @ 0.  Starting simulation...
info: Increasing stack size by one page.
A
Exiting @ tick 31079000 because exiting with last active thread context
rio@rio-Laptop ~/my_gem5>
```

**CPU Type = MinorCPU**

| | |
|---|---|
| final_tick | 31079000 |
| host_inst_rate | 75615 |
| host_mem_usage | 711648 |
| host_op_rate | 86043 |
| host_seconds | 0.11 |
| host_tick_rate | 287585199 |
| sim_freq | 1000000000000 |
| sim_insts | 8158 |
| sim_ops | 9296 |
| sim_seconds | 0.000031 |
| sim_ticks | 31079000 |

**Αρχείο stats.txt από εκτέλεση cpu-type=MinorCPU**

b. Όπως παρατηρούμε και από τα αποτελέσματα το MinorCPU model είναι σαφέστατα ποιο γρήγορο από το TimingSimpleCPU καθώς παρόλο που έχουν τον ίδιο τύπο cache το πρώτο υποστηρίζει pipelining και out-of-order functional execution ενώ το δεύτερο είναι ένα απλό μοντέλο σειριακής εκτέλεσης μιας εντολής τη φορά.

c. Παρατηρούμε ότι αυξάνοντας το μέγεθος της cache όπως ήταν αναμενόμενο, για μεγάλο όγκο δεδομένων, παρατηρείτε μείωση του χρόνου simulation. Κατά αντιστοιχία το ίδιο παρατηρείτε και με αύξηση του clock του συστήματος.

Από τις παρακάτω εικόνες μπορούμε να επιβεβαιώσουμε πως για αλλαγή της cache στο επίπεδο L1 έχουμε τις ανάλογες αλλαγές στο χρόνο εκτέλεσης.

| | |
|---|---|
| final_tick | 50612180 |
| host_inst_rate | 149955 |
| host_mem_usage | 684912 |
| host_op_rate | 165398 |
| host_seconds | 0.87 |
| host_tick_rate | 57929706 |
| sim_freq | 1000000000000 |
| sim_insts | 130993 |
| sim_ops | 144503 |
| sim_seconds | 0.000051 |
| sim_ticks | 50612180 |

**Stats για εκτέλεση MinorCPU και μεγάλη Cache**

| | |
|---|---|
| final_tick | 94880120 |
| host_inst_rate | 139292 |
| host_mem_usage | 684908 |
| host_op_rate | 153640 |
| host_seconds | 0.94 |
| host_tick_rate | 100877489 |
| sim_freq | 1000000000000 |
| sim_insts | 130993 |
| sim_ops | 144503 |
| sim_seconds | 0.000095 |
| sim_ticks | 94880120 |

**Stats για εκτέλεση MinorCPU και μικρή Cache**

| | |
|---|---|
| final_tick | 46458260 |
| host_inst_rate | 1124431 |
| host_mem_usage | 680300 |
| host_op_rate | 1239737 |
| host_seconds | 0.12 |
| host_tick_rate | 401935721 |
| sim_freq | 1000000000000 |
| sim_insts | 129819 |
| sim_ops | 143276 |
| sim_seconds | 0.000046 |
| sim_ticks | 46458260 |

**Stats για εκτέλεση TimingSimpleCPU και μεγάλη Cache**

| | |
|---|---|
| final_tick | 79664260 |
| host_inst_rate | 1079570 |
| host_mem_usage | 680300 |
| host_op_rate | 1190336 |
| host_seconds | 0.12 |
| host_tick_rate | 661761580 |
| sim_freq | 1000000000000 |
| sim_insts | 129819 |
| sim_ops | 143276 |
| sim_seconds | 0.000080 |
| sim_ticks | 79664260 |

**Stats για εκτέλεση TimingSimpleCPU και μικρή Cache**