GW ECE 3515 Lab Sheet

1 Branch Predictor

What is Static predictor?

What is Hybrid predictor?

What is OGEHL (Optimized Geometric History Length) predictor?

Table 1. Execution time (in msecs)

Predictor Type	Smatrix	Crafty	Mcf
Static			
Hybrid			
OGEHL			

Table 2. Branch Accuracy

Predictor Type	Smatrix	Crafty	Mcf
Static			
Hybrid			
OGEHL			

2 In-Order vs. Out-Order Execution

What is the difference between In-Order and Out-Order execution?

Table 3. Execution time (in msecs)

Execution Mode	Smatrix	Crafty	Mcf
In-Order			
Out-of-Order			

3 Superscalar Processor

Why superscalar processor has performance advantage over uniscalar processor?

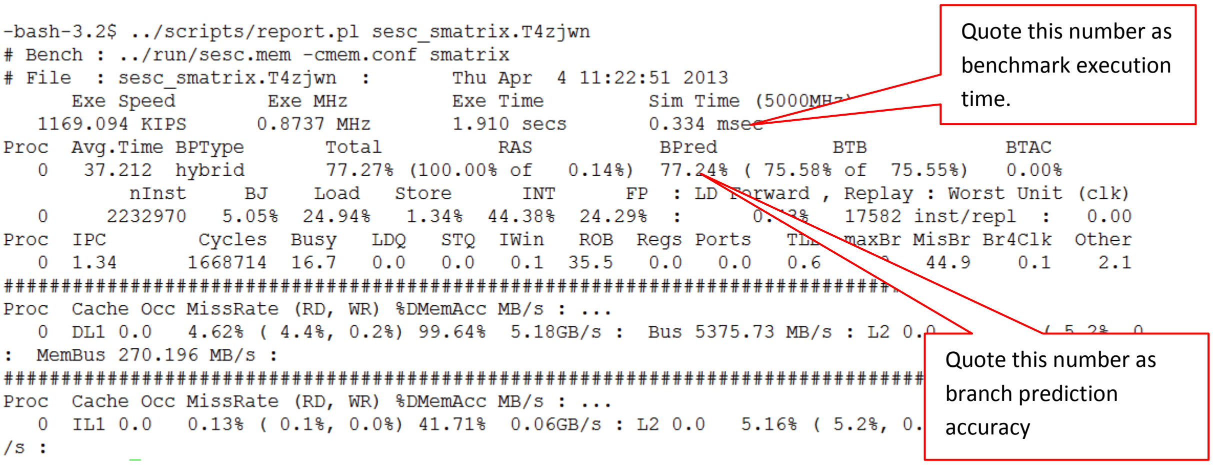
Table 4. Execution time (in msecs)

Width	Smatrix	Crafty	Mcf
1			
2			
4			
8			

LAB Instructions

Before starting your experiments, please make sure that you have SESC installed on your Linux shell server. Please refer to http://home.gwu.edu/~jiec/docs/sesc/sesc_intro.pdf for detailed information about how to install SESC and run benchmarks. Make sure you follow all the steps as listed in the sesc_intro.pdf.

In this lab, we will need to get the execution time of benchmarks and branch predictor accuracies. To get those numbers, please use the report.pl tool which is located in /sesc/scripts/report.pl (use report.pl –help for more usage). For example, if you have run the benchmark smatrix and got a report file sesc_smatrix.T4zjwn, you can type ../script/report.pl sesc_smatrix.T4zjwn in your tests directory, like this:



```
-bash-3.2$ ../scripts/report.pl sesc_smatrix.T4zjwn
# Bench : ../run/sesc.mem -cmem.conf smatrix
# File  : sesc_smatrix.T4zjwn   :        Thu Apr  4 11:22:51 2013
     Exe Speed        Exe MHz          Exe Time          Sim Time (5000MHz
   1169.094 KIPS     0.8737 MHz       1.910 secs        0.334 msec
Proc  Avg.Time BPType       Total          RAS           BPred          BTB            BTAC
   0   37.212  hybrid       77.27% (100.00% of   0.14%)  77.24% ( 75.58% of  75.55%)   0.00%
           nInst     BJ     Load    Store      INT      FP   : LD Forward , Replay : Worst Unit (clk)
   0     2232970   5.05%  24.94%   1.34%  44.38%  24.29%  :      0. 3%   17582 inst/repl  :   0.00
Proc  IPC          Cycles  Busy   LDQ   STQ  IWin   ROB  Regs Ports   TL    maxBr MisBr Br4Clk  Other
   0  1.34       1668714  16.7   0.0   0.0   0.1  35.5   0.0   0.0   0.6          44.9    0.1    2.1
#####################################################################
Proc  Cache Occ MissRate (RD, WR) %DMemAcc MB/s : ...
   0  DL1 0.0   4.62% ( 4.4%, 0.2%) 99.64%  5.18GB/s :  Bus 5375.73 MB/s : L2 0.0          ( 5.2%, 0
:  MemBus 270.196 MB/s :
#########################################################################
Proc  Cache Occ MissRate (RD, WR) %DMemAcc MB/s : ...
   0  IL1 0.0   0.13% ( 0.1%, 0.0%) 41.71%  0.06GB/s : L2 0.0   5.16% ( 5.2%, 0.
/s :
```

Quote this number as benchmark execution time.

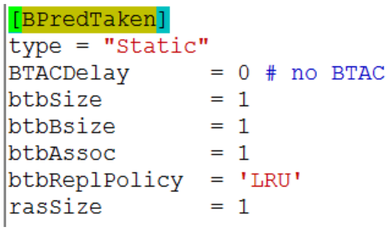
Quote this number as branch prediction accuracy

Attention: After every benchmark run, SESC will generate a new report file that has the format as “sesc_benchmarkname.randomletters”. Therefore, make sure that you know which file is newly generated such you that can correctly fill tables in the lab sheet.

Attention: Run benchmarks with SESC in the tests directory!

1 Branch Predictor

Step 1: In the tests directory, edit shared.conf. Go to line 340 and insert one line:
BTACDelay = 0 # no BTAC



```
[BPredTaken]
type = "Static"
BTACDelay       = 0 # no BTAC
btbSize         = 1
btbBsize        = 1
btbAssoc        = 1
btbReplPolicy   = 'LRU'
rasSize         = 1
```

Step 2: To use different branch predictors, static, hybrid, and ogehl, you need to edit the line 257 of the shared.conf one at a time, like this:

1. bpred = 'BPredIssueX' #this is to tell sesc to use hybrid branch predictor
2. bpred = ' BPredTaken' #this is to tell sesc to use static branch predictor
3. bpred = ' BestBPred' #this is to tell sesc to use OGEHL branch predictor

Attention: When you do the experiment, first use the default bpred configuration (BPredIssueX) and run the benchmarks one by one. After that, you change bpred’s value to BPredTaken and run the benchmarks again, and do so for BestPred.

2 In-Order vs. Out-Order Execution

Step 1: edit shared.conf in line 235, like this:

1. inorder = false # configure the processor to have in-order execution
2. inorder = true # configure the processor to have out-of-order execution

3 Superscalar Processor

Step 1: edit mem.conf in line 2, like this:

1. issue = 1 # uniscalar processor
2. issue = 2 # 2-issue superscalar processor
3. issue = 4 # 4-issue superscalar processor
4. issue = 8 # 8-issue superscalar processor