

ECE 4100 PROJECT 2

Verification/Experiments

Yingyan Wang / ywang627

I. Specification

Optimal Result with Most Hardware		Aiming Result (95%)
File	Average Retire/Cycle	Average Retire/Cycle
gcc	2.42207	2.300967
gobmk	2.364457	2.24623
hmmer	2.266854	2.153511
mcf	2.42207	2.300967

Table 1. Optimal Results and 95% Optimal Results.

(Specification for most hardware involved: $k_0=2$, $k_1=2$, $k_2=2$, $R=6$, Fetch Rate = 8)

II. Fetch Rate Discussion

Fetch Rate = 4 vs. Fetch Rate = 8

In this project, Cache misses are not accounted. Although in reality, increasing fetch rate, can cause more cache misses, but as this is not the part of hardware considered in the scope of this project. An arbitrary value of Fetch Rate = 8 will be used throughout the rest of the experiment.

III. Average Retire/Cycle for different Number of CDB

$k_0, k_1, k_2=2, FR = 8$				
File	$R = 5$	$R = 4$	$R = 3$	$R=2$
Gcc	2.42207	2.411091	2.366752	1.925336
gobmk	2.364457	2.361721	2.304625	1.881857
hmmer	2.266906	2.262546	2.206385	1.834055
mcf	2.369444	2.369444	2.324554	1.878887

Table2. Average Retire/Cycle for different R size.

(Specification: $k_0=2$, $k_1=2$, $k_2=2$, Fetch Rate = 8)

From the experiments result with varied R values, it is observed that 3 is the least value one can achieve, for a 95% optimal average retire/cycle. With R values below 3, the result would be falling below the 95% threshold.

IV. Maximum R value with minimum FU sizes.

File	Average Retire/Cycle
gcc	1.288610
gobmk	1.316846
hmmer	1.272621
mcf	1.258732

Table4. Average Retire/Cycle with one of the FU reduced to size of 1.

(Specification: k0,k1,k2=1, R = 6, Fetch Rate = 8)

V. Further experiments with varied FU size, with R = 3

File	k0=1	k1=1	k2=1
gcc	1.727951	2.000800	2.048887
gobmk	1.867100	2.027740	2.020774
hmmer	1.962563	1.867379	1.924335
mcf	1.850687	1.892828	1.984954

Table4. Average Retire/Cycle with one of the FU reduced to size of 1.

VI. Summary

From the experiments, one can observe that with maximum FUs available, R = 3 is the smallest value for an optimal solution above 95% threshold.

File	Optimal Percentage
gcc	97.7%
gobmk	97.4%
hmmer	97.3%
Mcf	98.1%

Table 5. Optimal Percentage of Average Retire/Cycle with Limited Hardware (R = 3).

Also with the same total amount of hardware, but k0,k1,k2 = 1, R = 6, the results are much lower than the optimal results. This is because limited FUs would reduce the average fire per cycle rate, and stalling

further actions. Therefore, even there are more result buses, the more instructions are stalled before being issued.

With the reduced R value of 3, further experiments were conducted, for possible reduction of FU size.

For all four trace files, reducing the size of any three of the FU, will decrease the average retire per cycle below the 95% threshold.

This result is consistent with speculation. Larger size of FU would increase the average issues/fires per cycle. However, the number of result bus will still have a limiting effect on the average retirements per cycle, as the instructions ready for the state update stage, would be stalled if all the result buses are busy.

In conclusion, the minimized hardware for the best results would be **$k_0=2, k_1=2, k_2=2, R=3$**

The total hardware is **$k_0+k_1+k_2+R = 2+2+2+3 = 9$** .