EE557 Project3

Project name	SA3		
Document ref			
Version			
Release date			
Author	Fei		
Classification	[Document classification]		
Distribution List	[Distribution list]		
Approved by	Name	Signature	Date

Fei Wu

6897429283

wufei@usc.edu

Ming Hsieh Department of Electrical Engineering
University of Southern California, Los Angeles, CA 90089
Fall 2022

EE 557

1 Section1

1.1 Constrains

1.1.1 Resource

Transistor count: 200 million
Area: 20 mm^2

1.1.2 Parameters

Dynamic Branch Predictor1: 2lev/bimod
Branch Target Buffer: 256 = 2*128
Size of Return Address Stack: 4
Machine Width (issue/decode/commit per cycle) already get in SA2: 2/2/4
Instruction Fetch Queue Size already get in SA2: 2
Register Update Unit Size2,3 (**must be equal or larger than 32-entry**): 32/64/128
Load/Store Queue Size: 16/32/64
Number of Integer ALUs and Multiplier/Divider Units: line
Number of Floating-point ALUs and Multiplier/Divider Units
Number of Memory Ports
Caches (Size, Associativity, Replacement Algorithm, Block Size) 2,4

1.1.3 Reminder

Change RUU > modifying the read and write ports
RUU size must equal or larger than 32entry
-max:isnt 50000000
-fastfwd 5000000

1.2 Design process

1. Based on the simulation assignment 2, we decide the best decision of the issue/decode/commit per cycle and fetch queue and the design is started with the size 2/2/2/2.
2. First, I will check the affect of the single parameter and the process will be like SA2. And the result will be represented as the result of the MIPS rates and cache misses. Also, the area and transistor will be the constrain so I will find a tradeoff between them and the performance.
3. The first parameter I modify and test will be the number of entries in the RUU, it requires 32 entries at least and I will check between the 32entries/64entries/128entries.
4. Then I will optimize the choice of predictor mode between 2lev/bimod/comb.
5. Then I will optimize the memory access capability, which means I will change the load and store queue size to avoid the stall because the full access of the LS queue, I will

check the different between 16/32/64/128 entries.

6. Then I am going to improve the execution performance, I am going to add some integer and floating-point operation units. Double the number of the integer units respectively and then double the number of both of them.

7. Then check the effects of port number of memory, check the result between 2/4 and compare the performance.

8. Check the affect of cache size and configuration.

9. Do some combine test to check the affect of different related combination parameters' affects.

Simulation cycle reference:
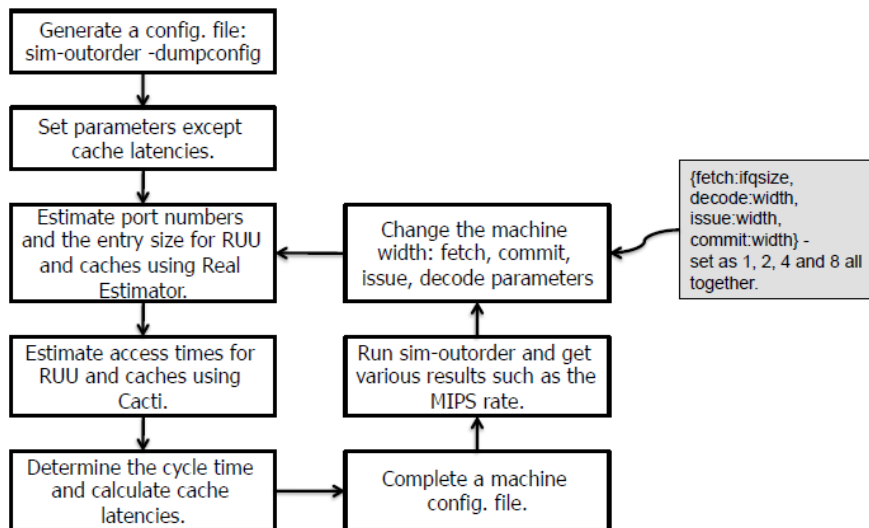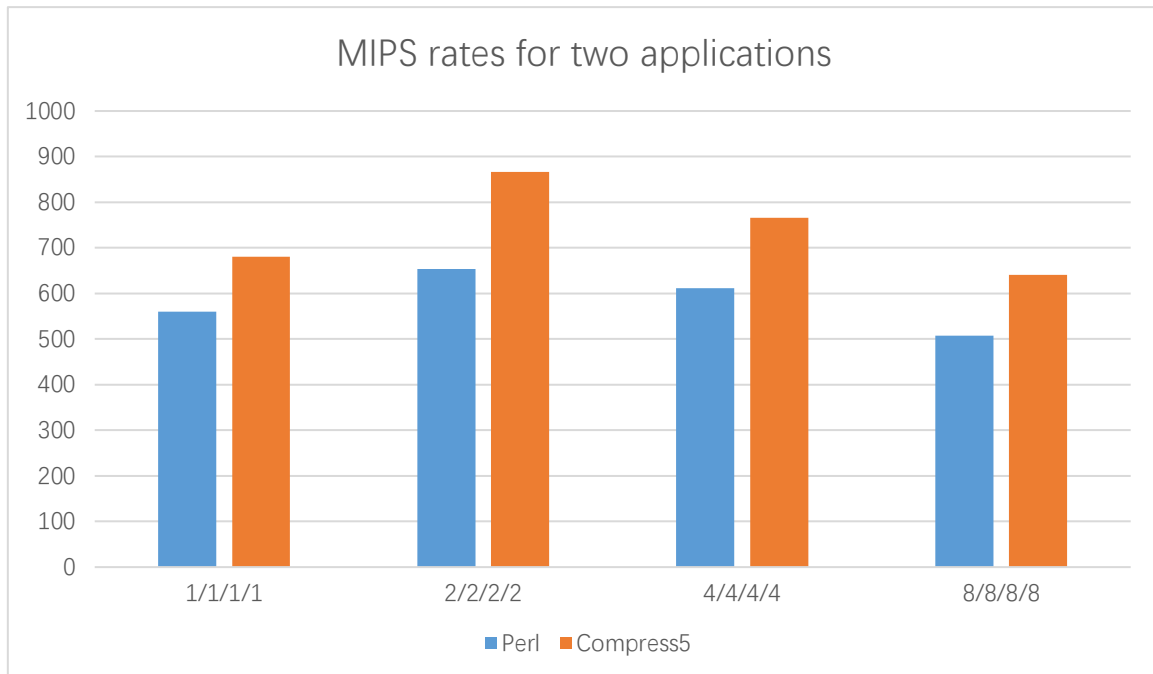

The iteration process:
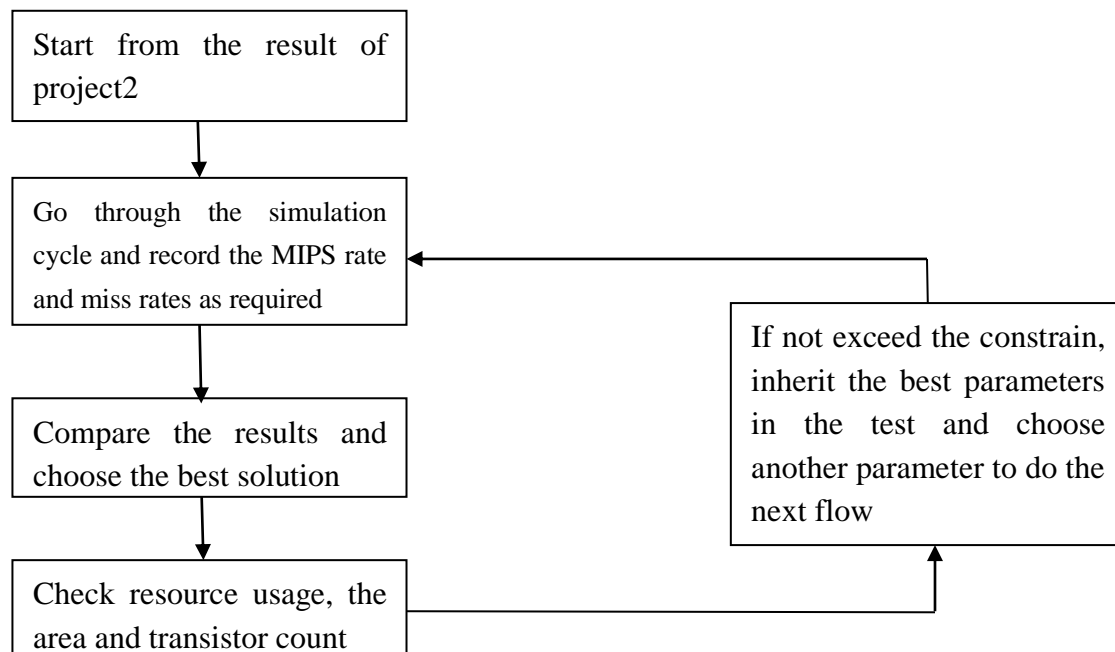
At the end of the signal parameter experiment, do some combined related parameters test.

2 Intermediate Results

2.1 Fetch/Issue/Decode

First step is to use the result of the simulation2 and set the fetch queue size to 2 and the decode and issue band width to 2.
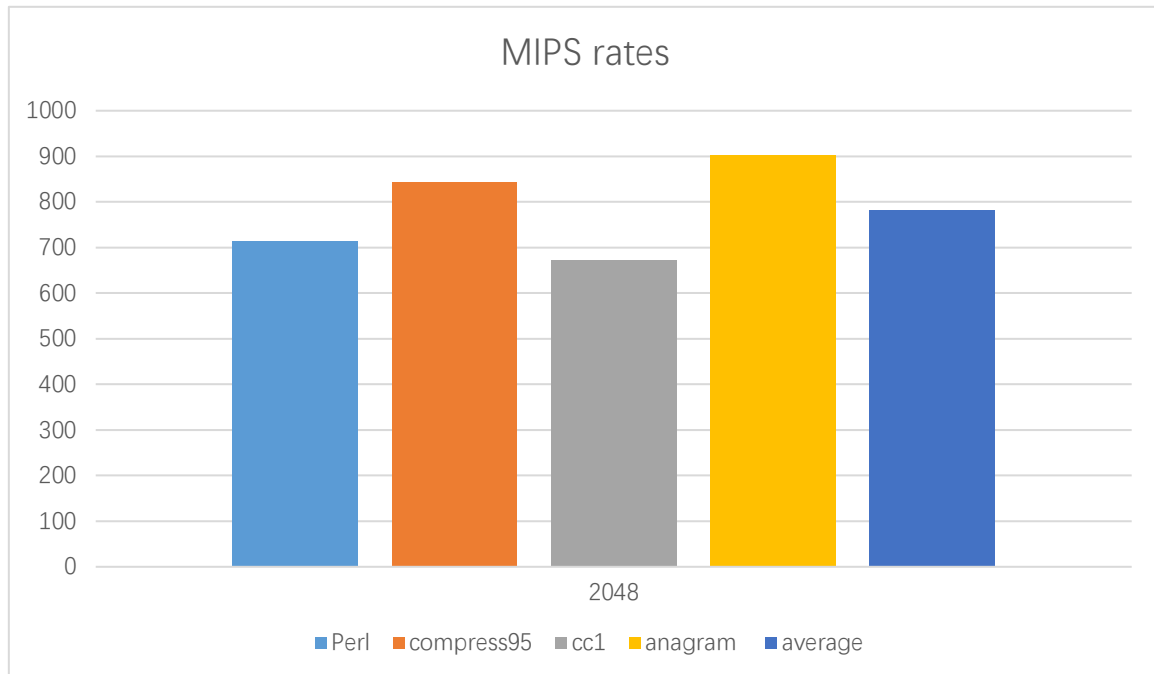

While I find the parameters given by simulation2 make the area exceed the requirement. The solution is trying to reduce the cache space.

2.2 Change the memory size

Reduce the set number of sets number of L2D to 2048

	L1-I	L1_D	L2
caches access times from Cacti (ns)	1.20055	1.67861	2.69151

RUU access times from Cacti (ns): 1.10936

MIPS rates

Transistors count 175504486

Area 79540.11 m^2

	Perl.ss	Compress95.ss	Cc1.ss	anagram	AVG
MIPS rates	713	842	672	902	782
Il1 misses rate	0.0315	0.0001	0.0329	0.0000	
Dl1 misses rate	0.0048	0.0305	0.0070	0.0016	
Ul2 misses rate	0.0097	0.0041	0.0046	0.2532	

2.3 RUU Entries

Reduce the set number of sets number of L2D to 2048

caches access times from Cacti (ns)	L1-I	L1_D	L2	RUU
32	1.20055	1.67861	2.69151	1.10936
64	1.20055	1.67861	2.69151	1.17978
128	1.20055	1.67861	2.69151	1.30264

MIPS rates

RUU	Perl.ss	Compress95.ss	Cc1.ss	anagram	AVG
32	694	864	667	931	789
64	680	794	640	794	728
128	628	719	580	719	662
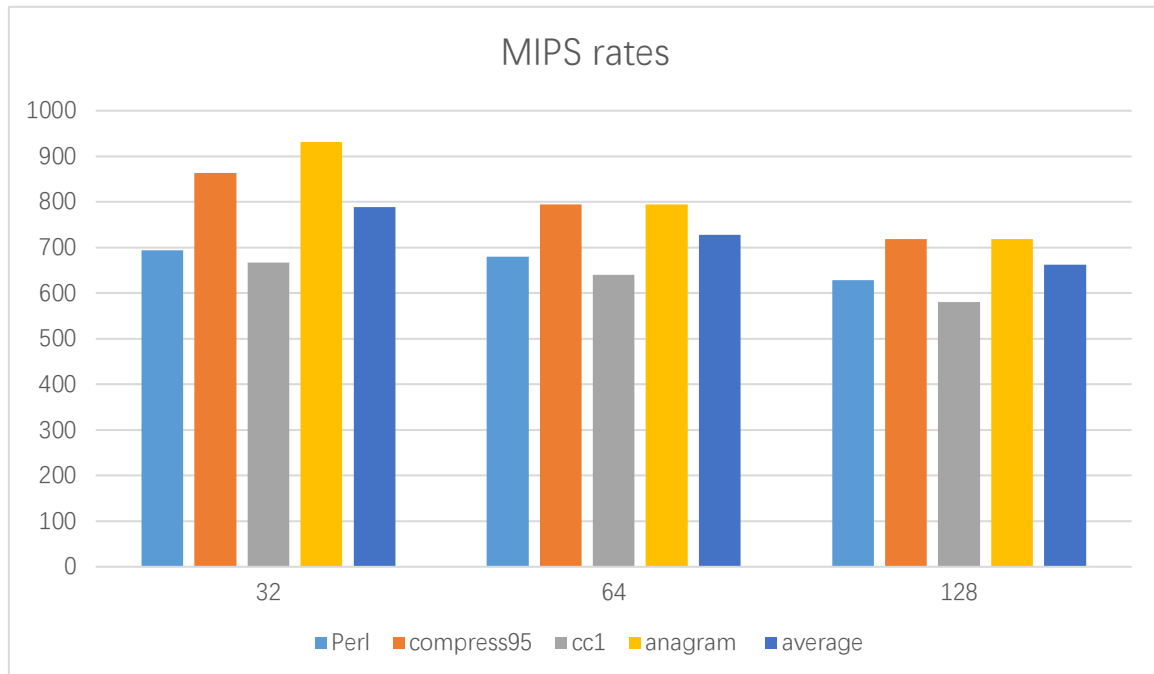
Il1 misses rate

RUU	Perl.ss	Compress95.ss	Cc1.ss	anagram
32	0.0357	0.0001	0.0395	0.0000
64	0.0298	0.0001	0.0314	0.0000
128	0.0297	0.0001	0.0312	0.0000

Dl1 misses rate

RUU	Perl.ss	Compress95.ss	Cc1.ss	anagram
32	0.0048	0.0307	0.0070	0.0016
64	0.0048	0.0308	0.0070	0.0016
128	0.0048	0.0308	0.0070	0.0016

Ul2 misses rate

RUU	Perl.ss	Compress95.ss	Cc1.ss	anagram
32	0.0108	0.0040	0.0049	0.2531
64	0.0094	0.0041	0.0045	0.2531
128	0.0093	0.0042	0.0045	0.2531

Still keep the entries number of RUU is 32.

Transistors count 175504486

Area 79540.11 m^2

2.4 Load/Store Queue Size
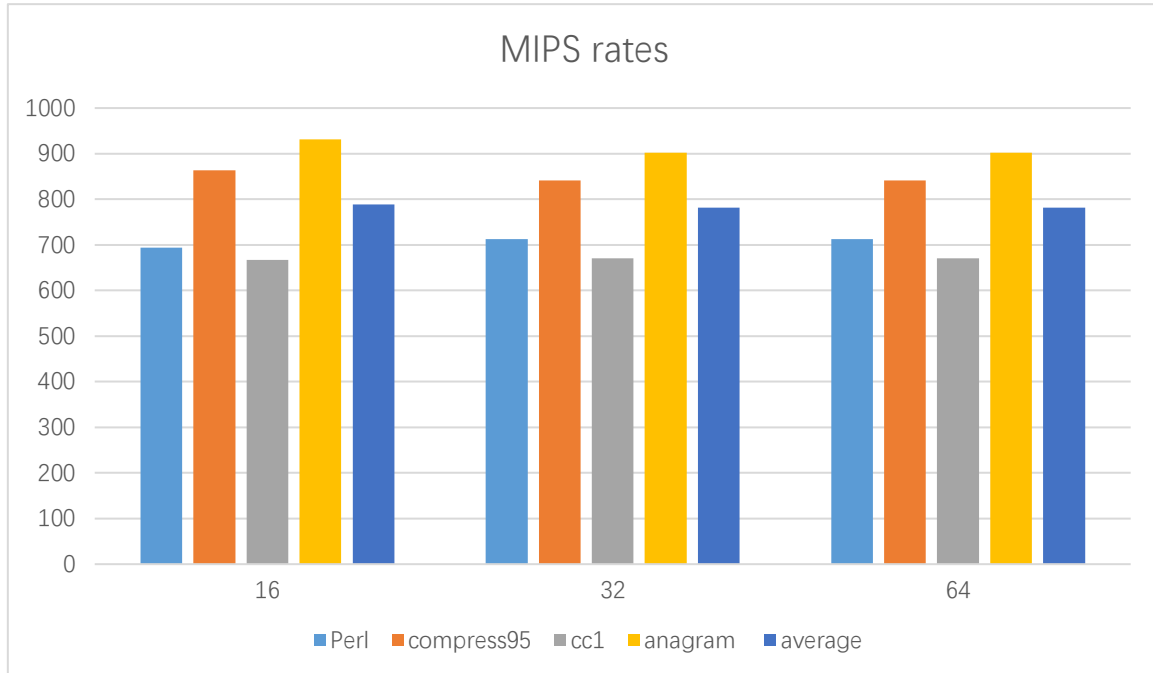
caches access times from Cacti (ns)

L/S	L1-I	L1_D	L2	RUU
16	1.20055	1.67861	2.69151	1.10936
32	1.20055	1.67861	2.69151	1.10936
64	1.20055	1.67861	2.69151	1.10936

MIPS rates

L/S	Perl.ss	Compress95.ss	Cc1.ss	anagram	AVG
16	694	864	667	931	789
32	713	841	671	902	782
64	713	841	671	902	782

Il1 misses rate

L/S	Perl.ss	Compress95.ss	Cc1.ss	anagram
16	0.0357	0.0001	0.0395	0.0000
32	0.0312	0.0001	0.0326	0.0000
64	0.0312	0.0001	0.0326	0.0000

Dl1 misses rate

L/S	Perl.ss	Compress95.ss	Cc1.ss	anagram
16	0.0048	0.0307	0.0070	0.0016
32	0.0048	0.0305	0.0070	0.0016
64	0.0048	0.0305	0.0070	0.0016

Ul2 misses rate

L/S	Perl.ss	Compress95.ss	Cc1.ss	anagram
16	0.0108	0.0040	0.0049	0.2531
32	0.0097	0.0041	0.0046	0.2532
64	0.0097	0.0041	0.0046	0.2532

Still keep the entries number of L/S queue is 16.

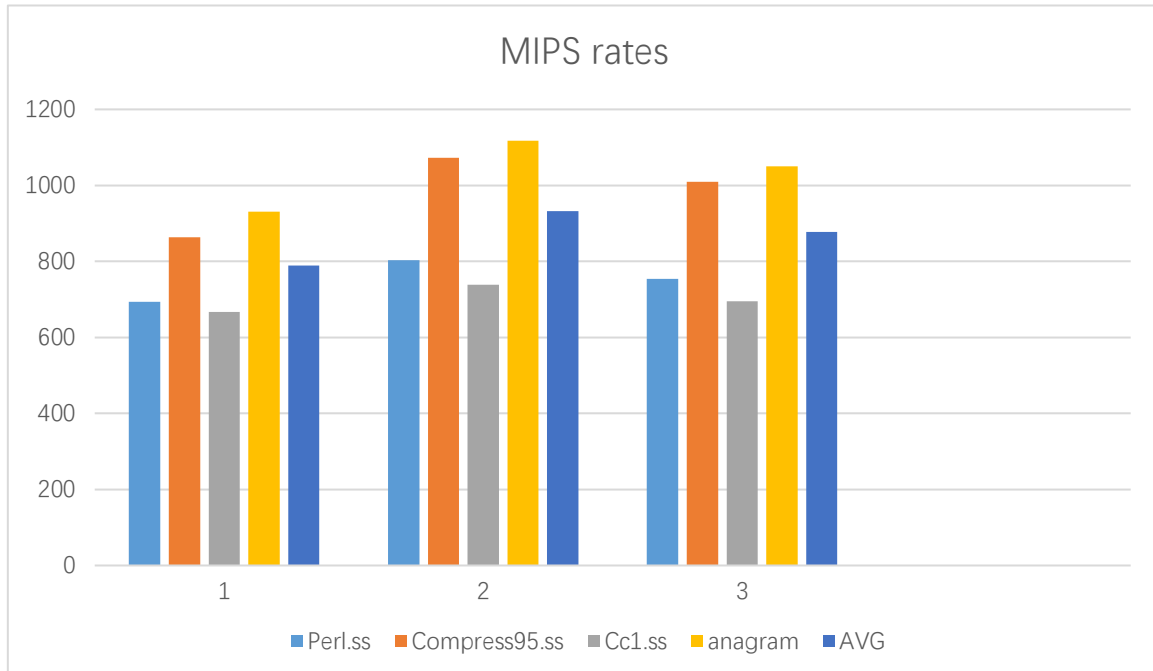Transistors count 175504486

Area 79540.11 m^2

2.5 Integer

caches access times from Cacti (ns)

Integer	L1-I	L1_D	L2	RUU

1	1.20055	1.67861	2.69151	1.10936
2	1.20055	1.67861	2.69151	1.19092
3	1.20055	1.67861	2.69151	1.26729
4	1.20055	1.67861	2.69151	1.33211

MIPS rates

L/S	Perl.ss	Compress95.ss	Cc1.ss	anagram	AVG
1	694	864	667	931	789
2	803	1073	739	1118	933
3	754	1009	695	1050	877

Il1 misses rate

Integer	Perl.ss	Compress95.ss	Cc1.ss	anagram
1	0.0357	0.0001	0.0395	0.0000
2	0.0322	0.0001	0.0340	0.0000
3	0.0330	0.0001	0.0351	0.0000

Dl1 misses rate

Integer	Perl.ss	Compress95.ss	Cc1.ss	anagram
1	0.0048	0.0307	0.0070	0.0016
2	0.0048	0.0304	0.0069	0.0016
3		0.0306	0.0069	0.0016

Ul2 misses rate

Integer	Perl.ss	Compress95.ss	Cc1.ss	anagram
1	0.0108	0.0040	0.0049	0.2531
2	0.0104	0.0041	0.0049	0.2531
3	0.0107	0.0041	0.0050	0.2529

Still keep the number of integer operation units is 2.

Transistors count 175724176

Area 80270.15 m^2

2.6 Floating-point

Redo the process like 2.6, while the IPC increase a little, while the access time increase, so the total MIPS decrease. So keep the units as 1;

2.7 Memory ports

Hypothetically, the resource will exceed the constrain if I double the memory ports. And try to set it as 4 and the transistor count can reach around 280M.
So, do not change the memory ports and keep it as 2.

2.8 Branch predictor

As seen, there is still so many missed when predicting and trying to optimize the predictor options.

caches access times from Cacti (ns)

Predictor	L1-I	L1_D	L2	RUU
2lev	1.20055	1.67861	2.69151	1.19092
Bimod 2048	1.20055	1.67861	2.69151	1.19092
comb	1.20055	1.67861	2.69151	1.19092

MIPS rates

	Perl.ss	Compress95.ss	Cc1.ss	anagram	AVG
2lev	803	1073	739	1118	933
Bimod 2048	802	1047	752	1241	960
comb	826	1084	772	1241	980

MIPS rates
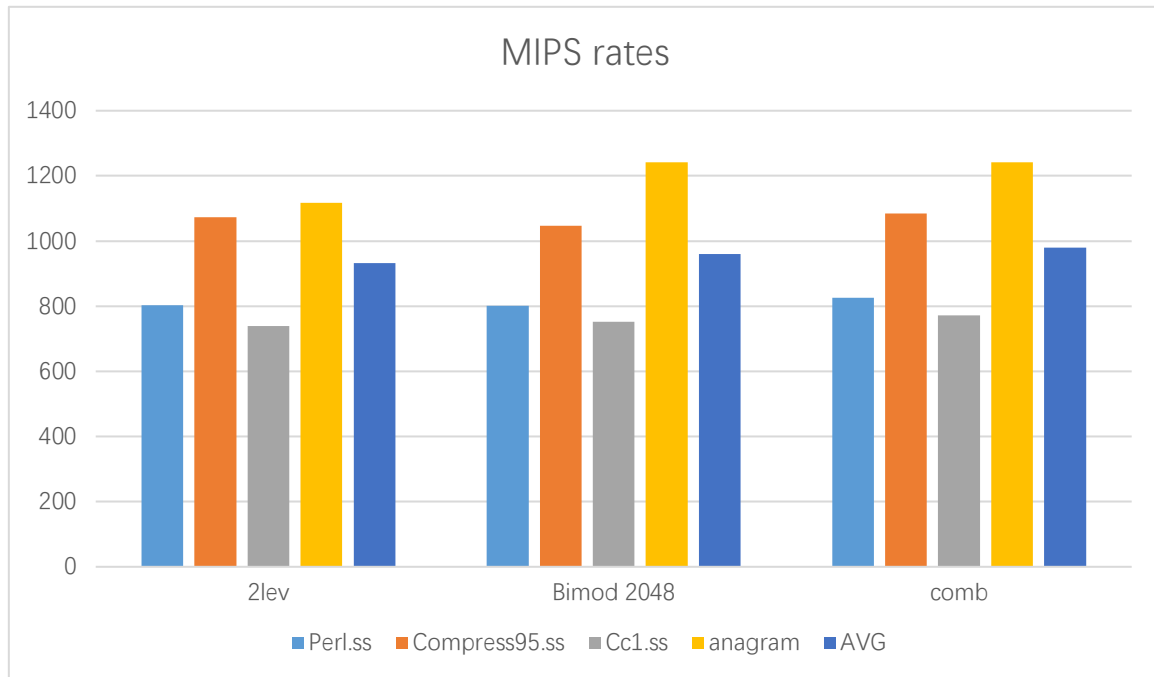
Il1 misses rate

	Perl.ss	Compress95.ss	Cc1.ss	anagram
Lev2	0.0330	0.0001	0.0351	0.0000
Bimod2048	0.0293	0.0001	0.0324	0.0000
comb	0.0314	0.0001	0.0340	0.0000

Dl1 misses rate

	Perl.ss	Compress95.ss	Cc1.ss	anagram
Lev2	0.0048	0.0306	0.0069	0.0016
Bimod2048	0.0048	0.0307	0.0069	0.0016
comb	0.0049	0.0308	0.0070	0.0016

Ul2 misses rate

	Perl.ss	Compress95.ss	Cc1.ss	anagram
Lev2	0.0107	0.0041	0.0050	0.2529
Bimod2048	0.0113	0.0040	0.0052	0.2530
comb	0.0113	0.0041	0.0052	0.2530

Still keep the number of integer operation units is 3.

Transistors count 175654715

Area 79913.35m^2

Then the comb will be use as the algorithm to predict the coming branch.

3 Section3:Final Design

a) MIPS rates: 980

b) Cycle time: 1.19092ns

c) Area from Estimator: 79913.35 m^2

d) Transistor count: 175654715

e) Cache latencies:

f) Cache miss rates:

	Perl.ss	Compress95.ss	Cc1.ss	anagram	Average	Latency
Il1	0.0314	0.0001	0.0340	0.0000	0.0164	1
Dl1	0.0049	0.0308	0.0070	0.0016	0.0111	2
Ul2	0.0113	0.0041	0.0052	0.2530	0.0684	2