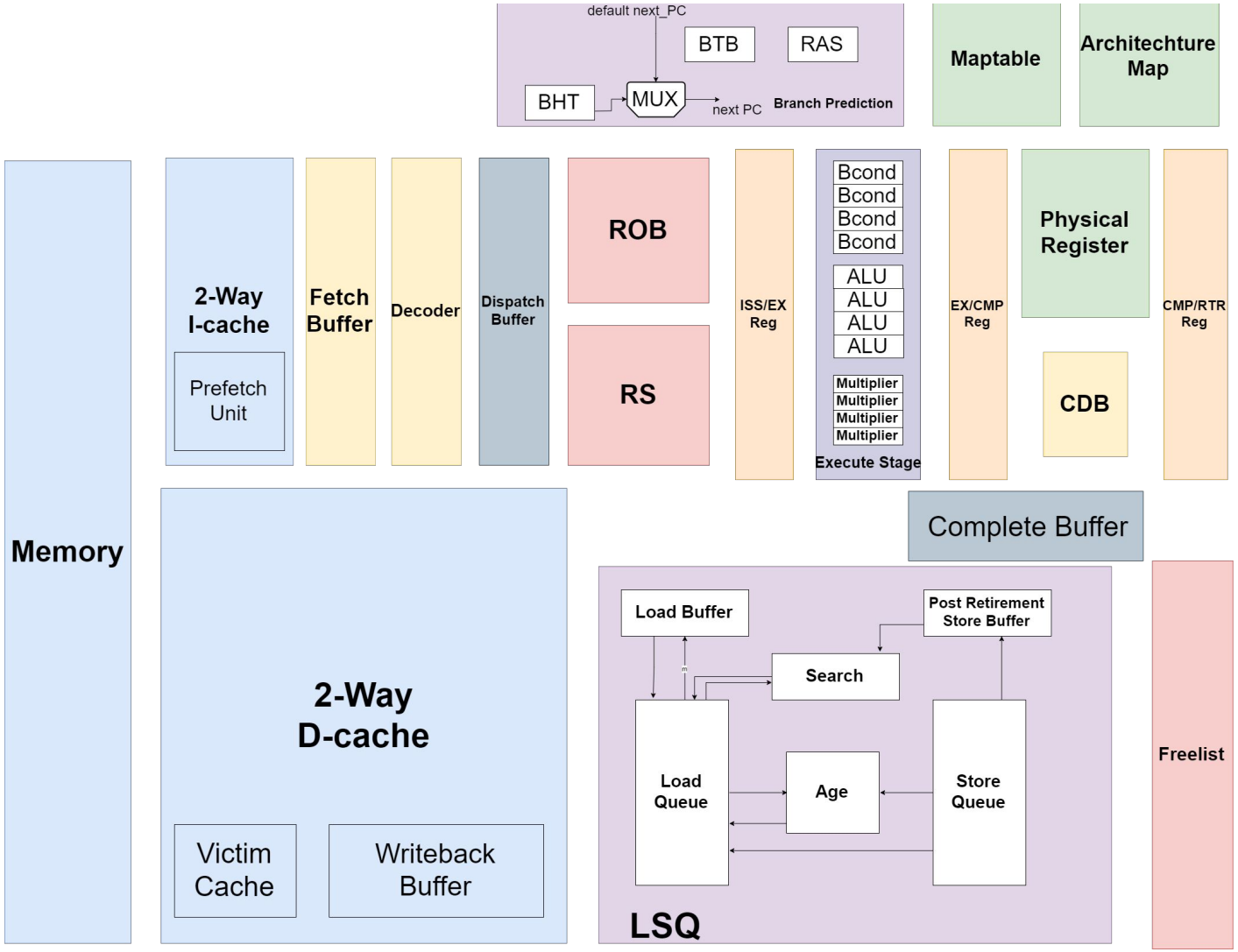
EECS 470 Computer Architecture
Project Presentation

Risc V 4-way Superscalar R10K OoO Processor

Group 8
OoO
Xiuneng Lu, Siyu Niu, Yunjie Pan, Runyu Zheng

Dec. 11th, 2019

High Level Design

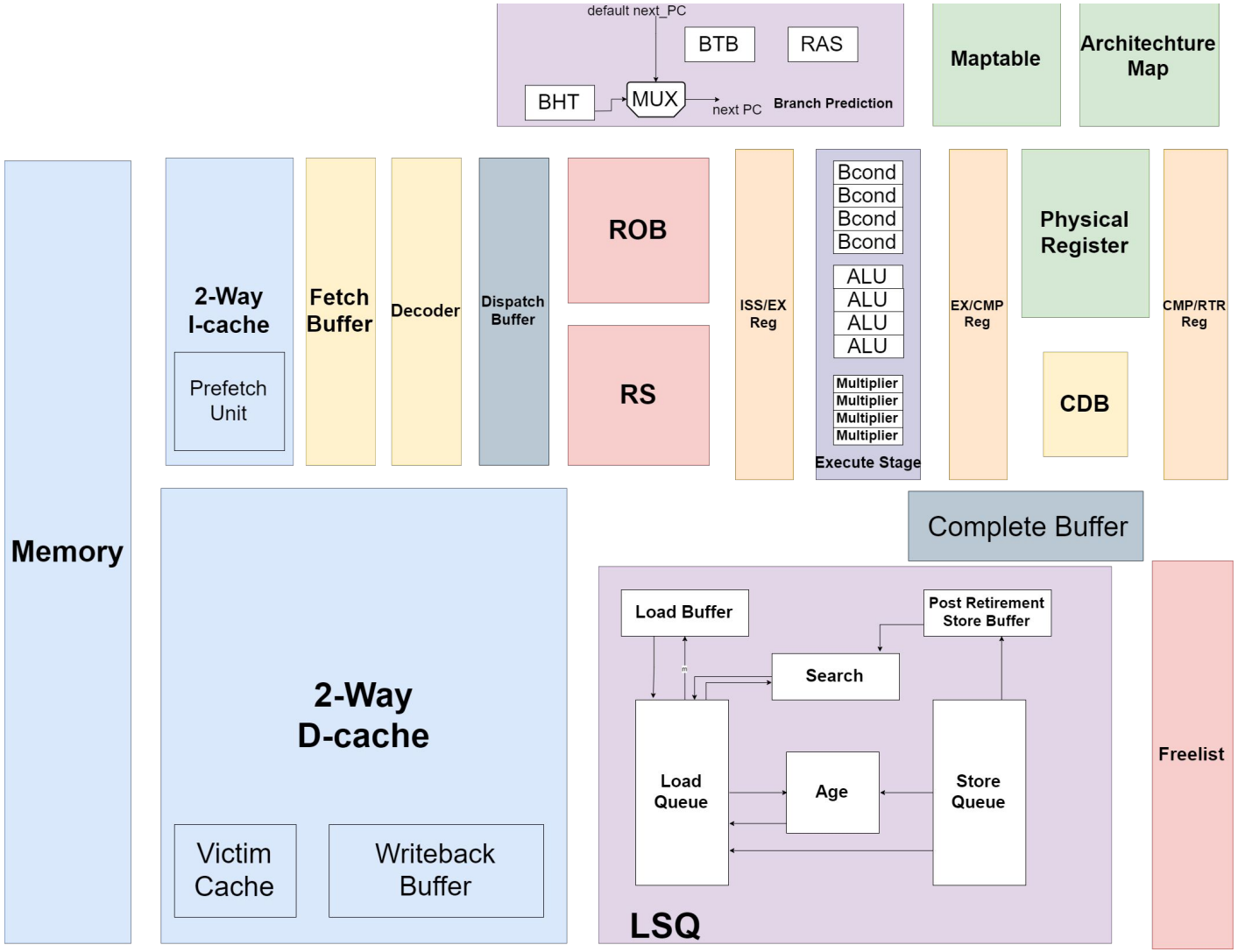


Features

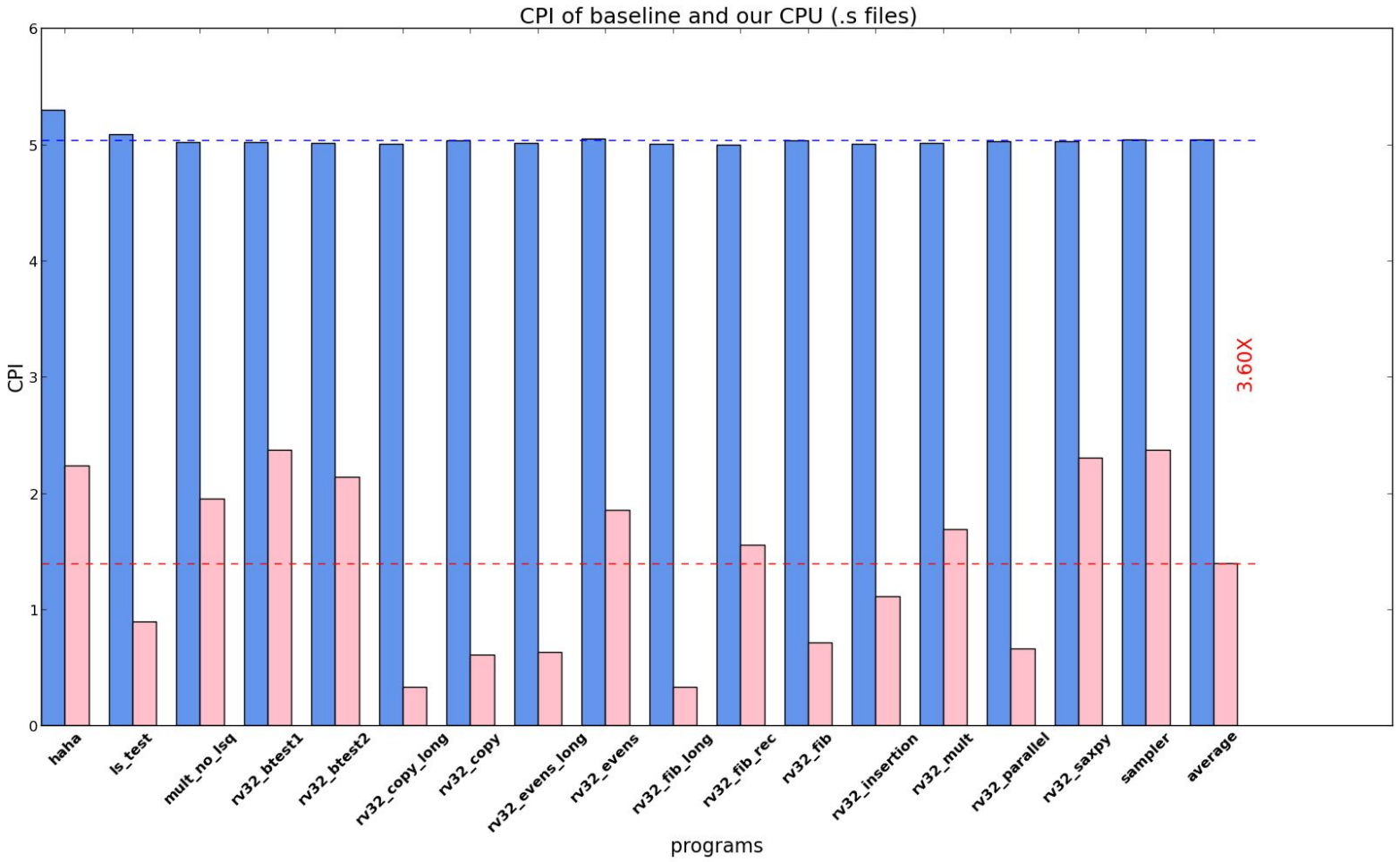
Features	Included	Comments
RISC V R10k OoO Processor	Yes	
Graphical debugging Tool	Yes	print contents of each buffer each cycle
Automated regression testing infrastructure	Yes	Run all tests and compare rst w/ P3
Superscalar	Yes	Superwidth = 4
Store-to-load forwarding in LSQ	Yes	We have, but only for SW
Loads issue out-of-order past pending stores (non-speculative)	Yes	As described
Post-retirement store buffer	Yes	16 entries
Multiple outstanding load misses	Yes	Hide miss lantancy
Next-line or stride prefetching for instructions and/or data	Yes	Prefetch next 16 instructions
Write-back data cache	Yes	write back & write allocate
Associativity > 1	Yes	2-Way
Victim cache	Yes	64 bit
Return address stack	Yes	16 entries
Loads speculatively issue past pending stores	No	Implemented. but still buggy
Load dependence predictor	No	Implemented, worked, but inefficient and influence critical path too much

Graphical Debugger

High Level Design



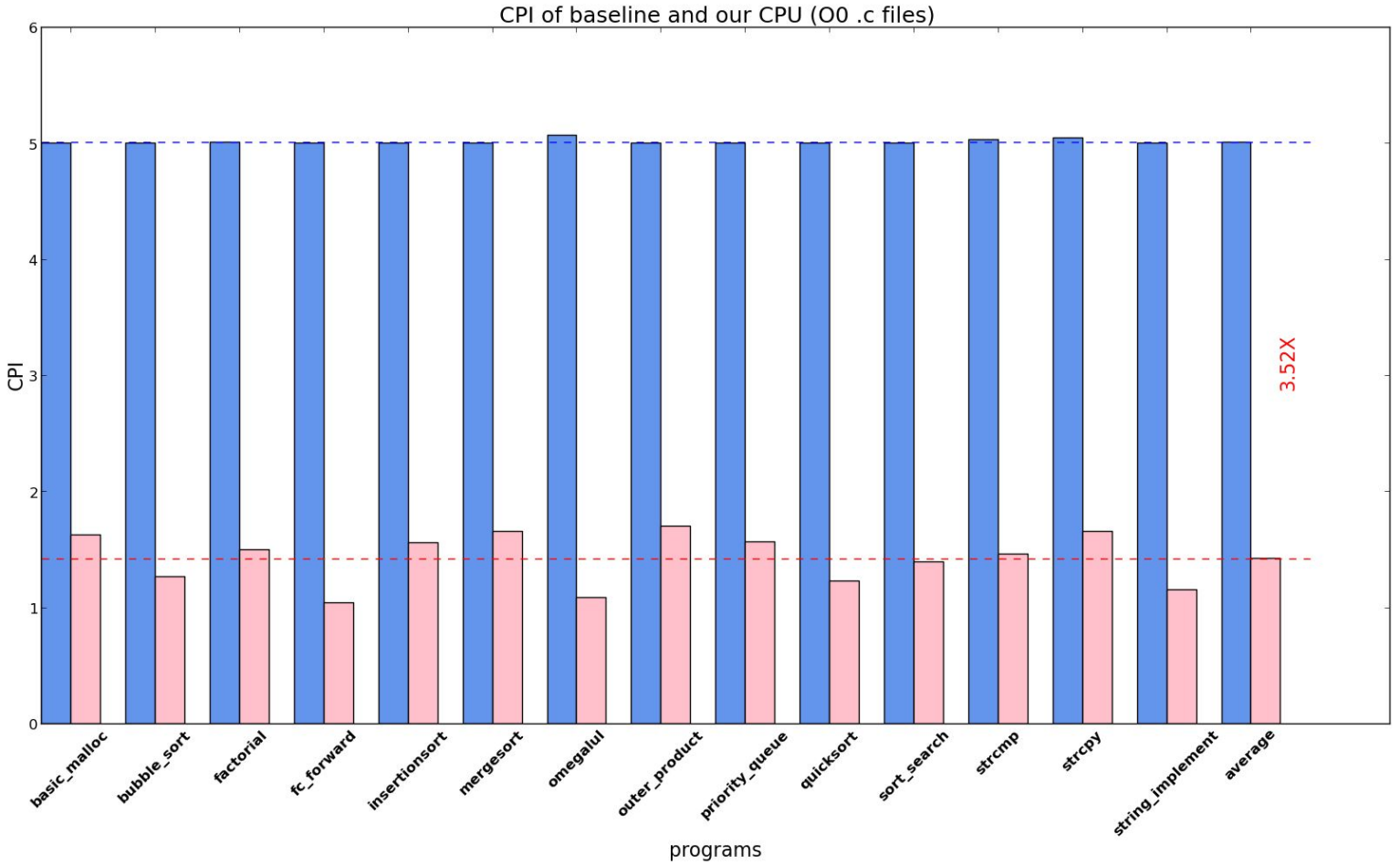
Results



for public testcase(.s files)
baseline average CPI=**5.04**
our average CPI = **1.40**
27.8% of baseline

clock cycle @Synthesis **17ns**

Results

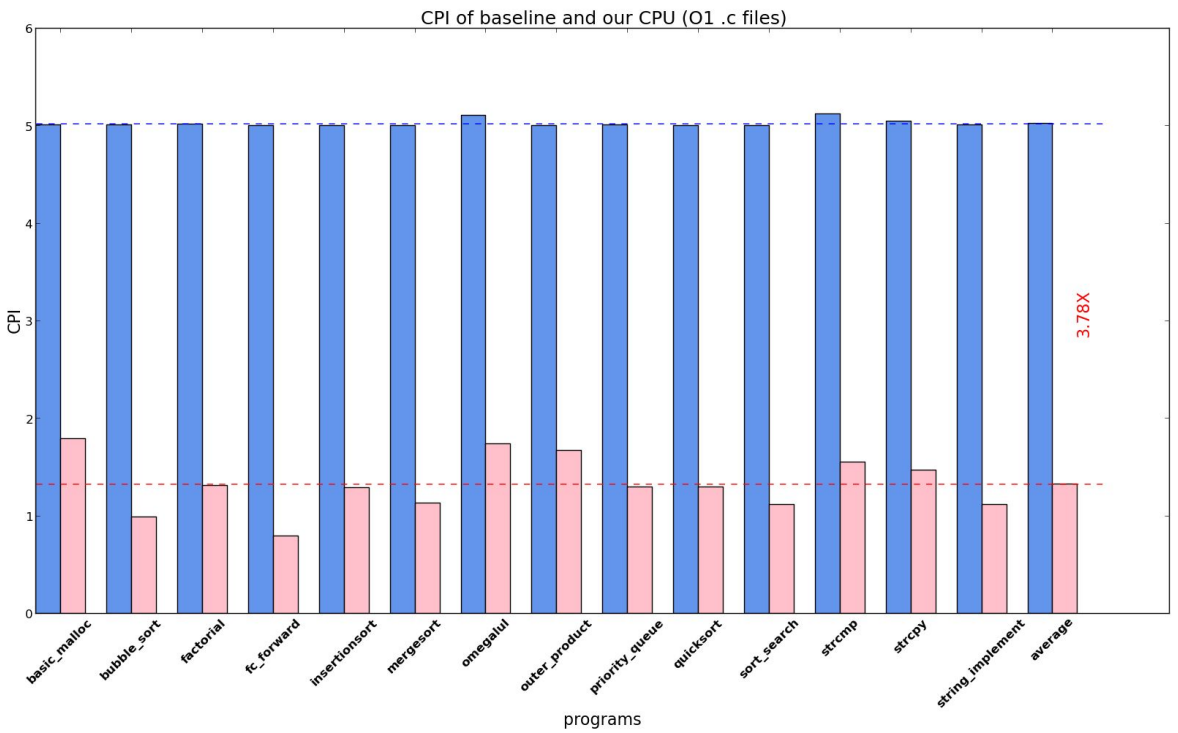


for public testcase(.c files) with O0 optimization

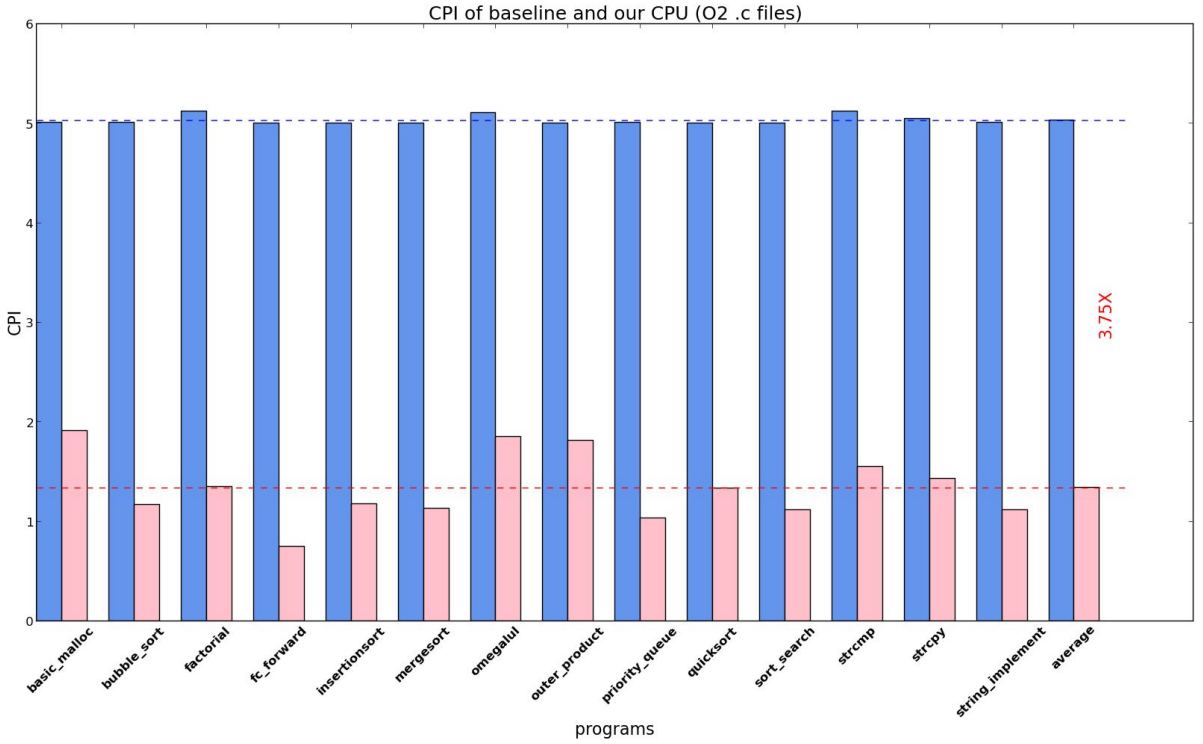
baseline average CPI=**5.01**
our average CPI = **1.42**
28.4% of baseline

clock cycle @Synthesis **17ns**

Results



-O1 average CPI = 1.32

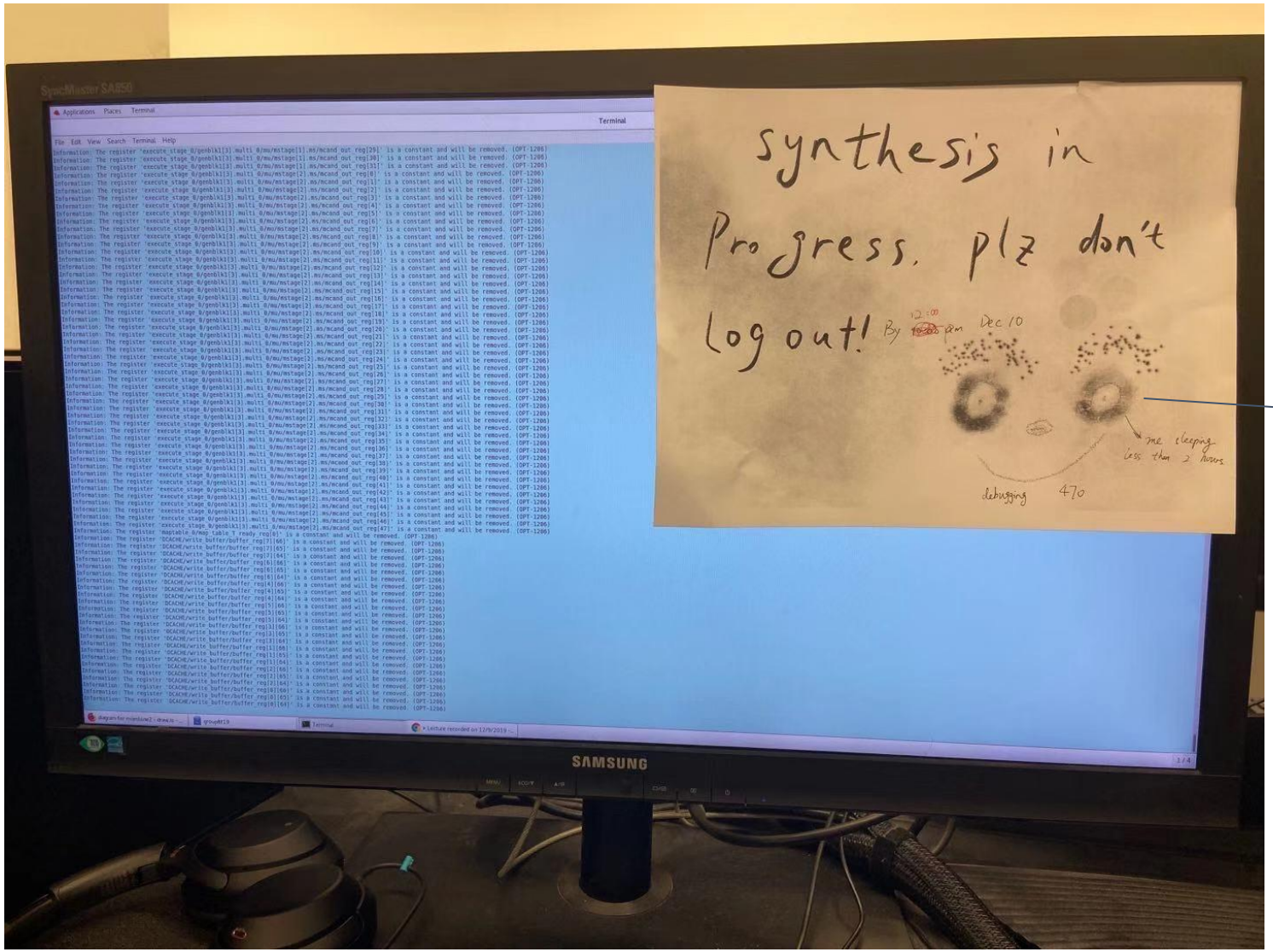


-O2 average CPI = 1.34

Starting it all over again...

- Leave more time for Optimization
  - We spent too much time on implementing the feature
- Always thinking the hardware cost before implementing large structures
- Always assign a initial value for a variable
  - simulate may assert xxx as 0 but synthesis may not
- Finish debugging speculative load and load dependence predictor

Q&A



EECS 470 makes me a good painter!