Advanced Topics in Computer Architecture

Lecture 8
Practice with Benchmarking and Simulation

Marenglen Biba
Department of Computer Science
University of New York Tirana

Benchmarking software

- SiSoftware Sandra (the System ANalyser, Diagnostic and Reporting Assistant) is an information & diagnostic utility.

Practice 1: Benchmark the ALU and FP units

Benchmark Setup

- Options: multithreading, SMT

1. Aggregate Arithmetic Performance
2. Performance Vs. Speed
3. Performance Vs. Power
4. Performance VS. Price

Practice 2: Multi-media performance of processors.

- Multimedia integer
- Multimedia float
- Performance Vs. Price
- Performance Vs. Speed
- Performance Vs. Power

Practice 3: Multi-core efficiency

- Inter-core latency
- Inter-core bandwidth
- Performance Vs. Price
- Capacity (cache size) Vs. Price
- Capacity Vs. Power

Practice 4: Power Management Efficiency

- ALU Power Performance
- FPU Power Performance
- Capacity Vs. Price
- Performance Vs. Power
- Performance Vs. Speed

Practice 5: Cryptography

- Cryptographic bandwidth
- Hashing bandwidth
- Performance Vs. Price
- Performance Vs. Speed
- Performance Vs. Power

Practice 6: Memory Bandwidth

- Integer Memory Bandwidth
- Float Memory Bandwidth
- Performance Vs. Price
  - Aggregate Memory Performance
- Performance Vs. Speed
- Performance Vs. Power

Practice 7: Memory Latency

- Performance Vs. Price
- Performance Vs. Speed
- Performance Vs. Power

Practice 8: Cache and Memory

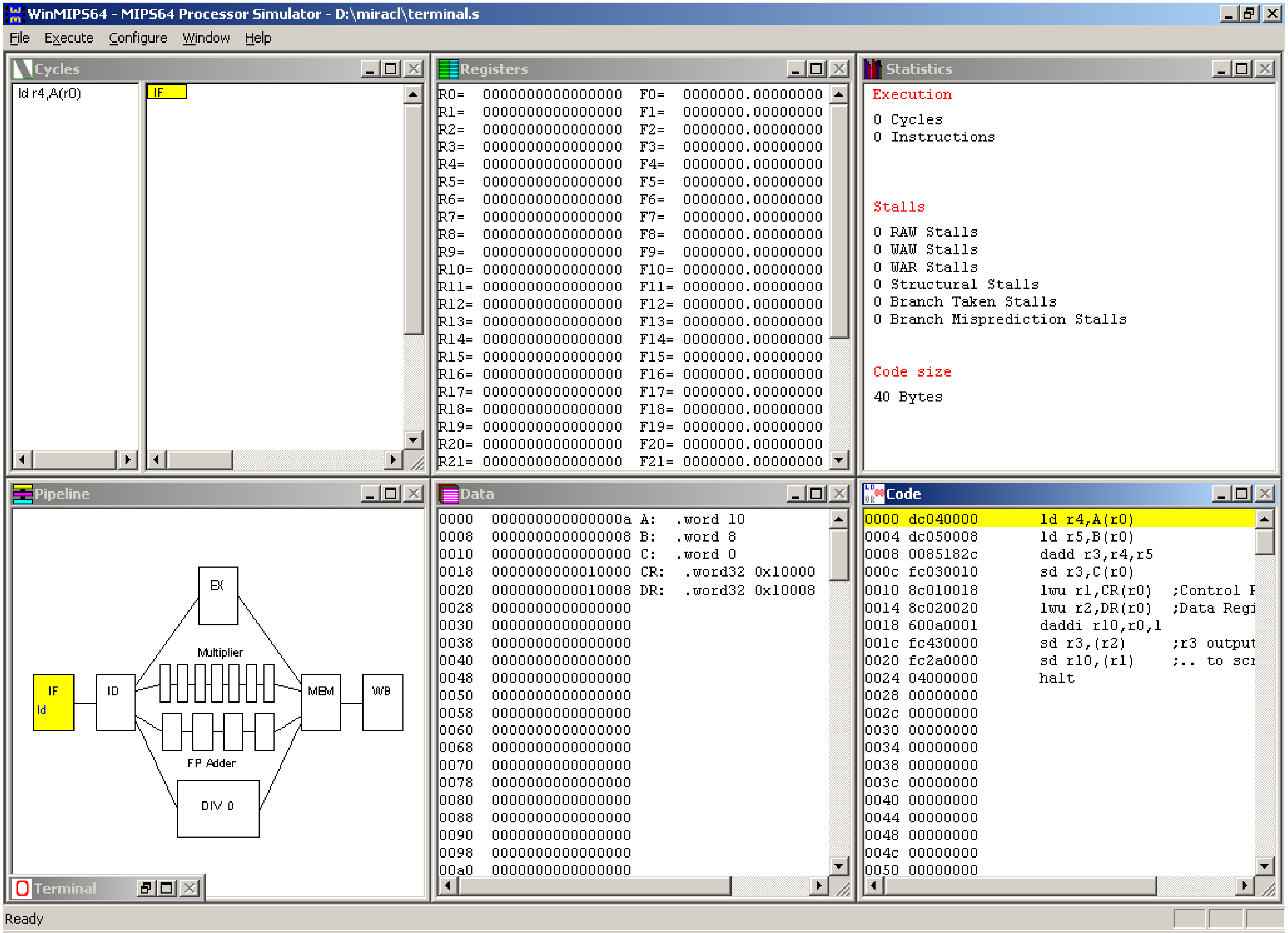
- Test Block Size Vs. Data Bandwidth
- Cache memory bandwidth
- Performance Vs. Price
- Performance Vs. Speed
- Performance Vs. Power

Other Practices

- .NET Arithmetic
- .NET Multi-media
- Java Arithmetic
- Java Multi-media

SIMULATION WITH WINMIPS64

WinMIPS64



```
R0=  0000000000000000  F0=  0000000.00000000
R1=  0000000000000000  F1=  0000000.00000000
R2=  0000000000000000  F2=  0000000.00000000
R3=  0000000000000000  F3=  0000000.00000000
R4=  0000000000000000  F4=  0000000.00000000
R5=  0000000000000000  F5=  0000000.00000000
R6=  0000000000000000  F6=  0000000.00000000
R7=  0000000000000000  F7=  0000000.00000000
R8=  0000000000000000  F8=  0000000.00000000
R9=  0000000000000000  F9=  0000000.00000000
R10= 0000000000000000  F10= 0000000.00000000
R11= 0000000000000000  F11= 0000000.00000000
R12= 0000000000000000  F12= 0000000.00000000
R13= 0000000000000000  F13= 0000000.00000000
R14= 0000000000000000  F14= 0000000.00000000
R15= 0000000000000000  F15= 0000000.00000000
R16= 0000000000000000  F16= 0000000.00000000
R17= 0000000000000000  F17= 0000000.00000000
R18= 0000000000000000  F18= 0000000.00000000
R19= 0000000000000000  F19= 0000000.00000000
R20= 0000000000000000  F20= 0000000.00000000
R21= 0000000000000000  F21= 0000000.00000000
```

```
Execution

0 Cycles
0 Instructions


Stalls

0 RAW Stalls
0 WAW Stalls
0 WAR Stalls
0 Structural Stalls
0 Branch Taken Stalls
0 Branch Misprediction Stalls


Code size

40 Bytes
```

```
0000  000000000000000a A:   .word 10
0008  0000000000000008 B:   .word 8
0010  0000000000000000 C:   .word 0
0018  0000000000010000 CR:   .word32 0x10000
0020  0000000000010008 DR:   .word32 0x10008
0028  0000000000000000
0030  0000000000000000
0038  0000000000000000
0040  0000000000000000
0048  0000000000000000
0050  0000000000000000
0058  0000000000000000
0060  0000000000000000
0068  0000000000000000
0070  0000000000000000
0078  0000000000000000
0080  0000000000000000
0088  0000000000000000
0090  0000000000000000
0098  0000000000000000
00a0  0000000000000000
```

```
0000 dc040000       ld r4,A(r0)
0004 dc050008       ld r5,B(r0)
0008 0085182c       dadd r3,r4,r5
000c fc030010       sd r3,C(r0)
0010 8c010018       lwu r1,CR(r0)   ;Control F
0014 8c020020       lwu r2,DR(r0)   ;Data Regi
0018 600a0001       daddi r10,r0,1
001c fc430000       sd r3,(r2)      ;r3 output
0020 fc2a0000       sd r10,(r1)     ;.. to scr
0024 04000000       halt
0028 00000000
002c 00000000
0030 00000000
0034 00000000
0038 00000000
003c 00000000
0040 00000000
0044 00000000
0048 00000000
004c 00000000
0050 00000000
```

Practice: Examples in WinMIPS64

- Sum of two numbers
- Multiplication of floating point numbers
  - Execute with only forwarding enabled
  - Enable branch target buffer and see the effect in CPI and misprediction
- InsertionSort
  - Raw stalls
  - Execute with only forwarding enabled
  - Enable branch target buffer and see the effect in CPI and misprediction
- Factorial
- Series
- Power

End of Lecture

- Readings
  - Sandra SiSoftware Documentation
  - WinMIPS64 documentation