Performance

Hung-Wei Tseng

Announcement

- Homework #1 due next Monday before class
- Reading quizzes 4.1-4.4 due next Tuesday
- Office hour ThF 11a-12p @ CSE 3217
- Slides on course webpage
  - Pre-release slides: published before we start new topics, not including clicker questions. Just for note-taking
  - Slides: published after class, everything in the class
- Midterm
  - Similar to homework questions
  - Similar to clicker question, but not multiple choices
  - Short answer questions

Outline

- What is performance?
- What is the performance equation?
- What affects performance

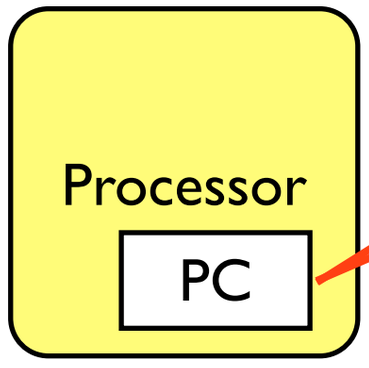
Performance!

What do you want in a computer?

- Frame rate
- Responsiveness
- Real-time
- Throughput
- Cost
- Volume
- Weight
- Battery life
- Low power/low temperature
- Reliability
- Latency/Execution time

Execution Time

- The simplest kind of performance
- Shorter execution time means better performance
- Usually measured in seconds



instruction memory

```
120007a30:   0f00bb27   ldah   gp,15(t12)
120007a34:   509cbd23   lda    gp,-25520(gp)
120007a38:   00005d24   ldah   t1,0(gp)
120007a3c:   0000bd24   ldah   t4,0(gp)
120007a40:   2ca422a0   ldl    t0,-23508(t1)
120007a44:   130020e4   beq    t0,120007a94
120007a48:   00003d24   ldah   t0,0(gp)
120007a4c:   2ca4e2b3   stl    zero,-23508(t1)
120007a50:   0004ff47   clr    v0
120007a54:   28a4e5b3   stl    zero,-23512(t4)
120007a58:   20a421a4   ldq    t0,-23520(t0)
120007a5c:   0e0020e4   beq    t0,120007a98
120007a60:   0204e147   mov    t0,t1
120007a64:   0304ff47   clr    t2
120007a68:   0500e0c3   br     120007a80
```

How many of these?

Instruction Count!

How long is it take to execution each of these?

Cycles per instruction * cycle time

Performance equation!

Performance Equation

$$\text{Execution Time} = \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Cycles}}{\text{Instruction}} \times \frac{\text{Seconds}}{\text{Cycle}}$$

How many instruction executed?

How long is it to execute each instruction

- ET = IC * CPI * CT
- IC (Instruction Count)
- CPI (Cycles Per Instruction)
- CT (Seconds Per Cycle)
  - 1 Hz = 1 second per cycle; 1 GHz = 1 ns per cycle

Speedup

- Compare the relative performance of the baseline system and the improved system
- Definition

$$\text{Speedup} = \frac{\text{Execution time}_{\text{baseline}}}{\text{Execution time}_{\text{improved system}}}$$

What affects performance

How compiler affects performance?

- ET = IC * CPI * CT
- What can a compiler affect?

A. IC

B. IC & CPI

C. IC, CPI & CT

D. IC & CT

Demo: compiler & performance

- Compiler optimization can help reducing the instruction count
- Compiler optimization can improve CPI
  - Wise selection of instruction combinations
  - Use registers to eliminate loads and stores

Recap: Performance Equation

$$\text{Execution Time} = \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Cycles}}{\text{Instruction}} \times \frac{\text{Seconds}}{\text{Cycle}}$$

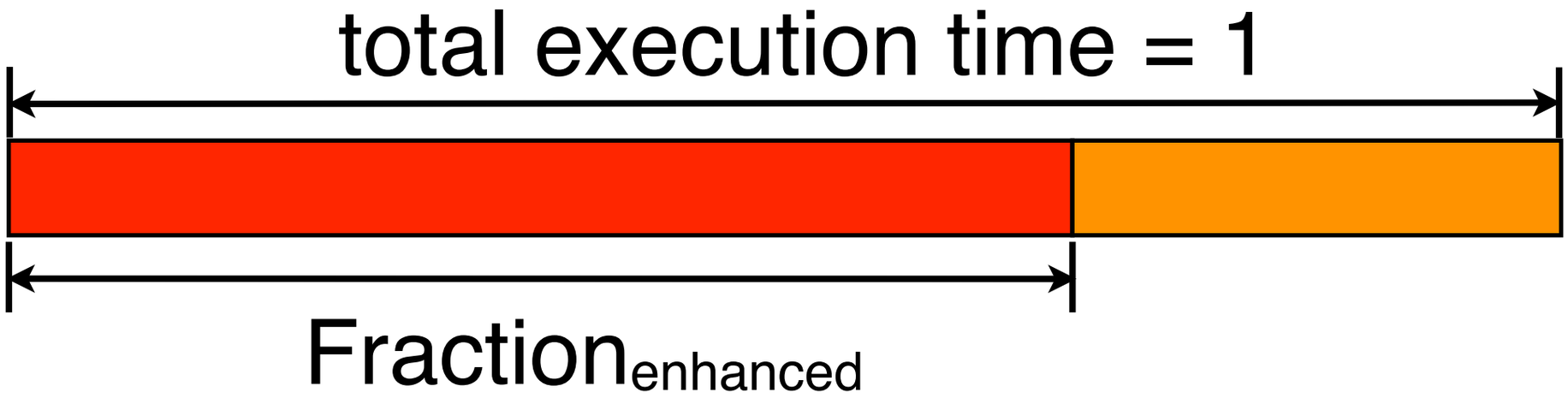
- ET = IC * CPI * Cycle Time
- IC (Instruction Count)
  - ISA, Compiler, algorithm, programming language
- CPI (Cycles Per Instruction)
  - Machine Implementation, microarchitecture, compiler, application, algorithm, programming language
- Cycle Time (Seconds Per Cycle)
  - Process Technology, microarchitecture

Amdahl's Law

Amdahl's Law

$$\text{Speedup} = \frac{1}{(1 - \text{Fraction}_{\text{enhanced}}) + \frac{\text{Fraction}_{\text{enhanced}}}{\text{Speedup}_{\text{enhanced}}}}$$

- Amdahl's Law can be used anywhere!
- The Fraction means the fraction of "time"



Amdahl's Law

- $$\text{Speedup} = \frac{1}{(1 - \text{Fraction}_{\text{enhanced}}) + \frac{\text{Fraction}_{\text{enhanced}}}{\text{Speedup}_{\text{enhanced}}}}$$

- Assume that we have an application composed with a total of 500000 instructions, in which 20% of them are the load/store instructions with an average CPI of 6 cycles, and the rest instructions are integer instructions with average CPI of 1 cycle.
  - If we double the clock rate to be 2GHz without improve the memory latency, the average CPI for load/store instruction will also be doubled to 12 cycles. What's the performance improvement after this change?

$$\text{Fraction}_{\text{enhanced}} = \frac{500000*(0.8*1)*1}{500000*(0.8*1+0.2*6)*1} = 0.4$$

$$\text{Speedup} = \frac{1}{(1 - 0.4) + \frac{0.4}{2}} = 1.25$$

Amdahl's Law and Multi-core Processor

- Assume that we have an application, in which 50% of the application can be fully parallelized with 2 processors. What's the speedup if we use a dual-core processor instead of a single-core processor?

$$\text{Speedup} = \frac{1}{(1 - \text{Fraction}_{\text{enhanced}}) + \frac{\text{Fraction}_{\text{enhanced}}}{\text{Speedup}_{\text{enhanced}}}}$$

$$\text{Speedup}_{\text{dual}} = \frac{1}{(1 - 0.5) + \frac{0.5}{2}} = 1.33$$

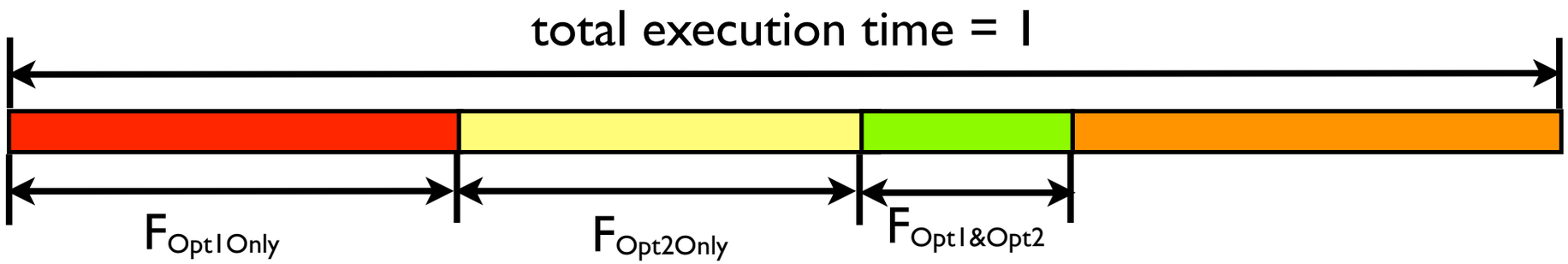
Multiple optimizations

- We can apply Amdahl's law for multiple optimizations
- These optimizations must be dis-joint!
  - If optimization #1 and optimization #2 are dis-joint:

$$\text{Speedup} = \frac{1}{(1 - F_{Opt1} - F_{Opt2}) + \frac{F_{Opt1}}{Speedup_{Opt1}} + \frac{F_{Opt2}}{Speedup_{Opt2}}}$$

  - If optimization #1 and optimization #2 are not dis-joint:

$$S = \frac{1}{(1 - F_{Opt1Only} - F_{Opt2Only} - F_{Opt1\&Opt2}) + \frac{F_{Opt1}}{Speedup_{Opt1Only}} + \frac{F_{Opt2}}{Speedup_{Opt2Only}} + \frac{F_{Opt1\&Opt2}}{Speedup_{Opt1\&Opt2}}}$$



Amdahl's Law for quad-core processor

- Assume that we have an application, in which 50% of the application can be fully parallelized with 2 processors. Assuming 50% of the parallelized part can be further parallelized with 4 processors, what's the speed up of the application running on a 4-core processor?

Code can be optimized for 2-core = 50%*50% = 25%

Code can be optimized for 4-core = 50%*50% = 25%

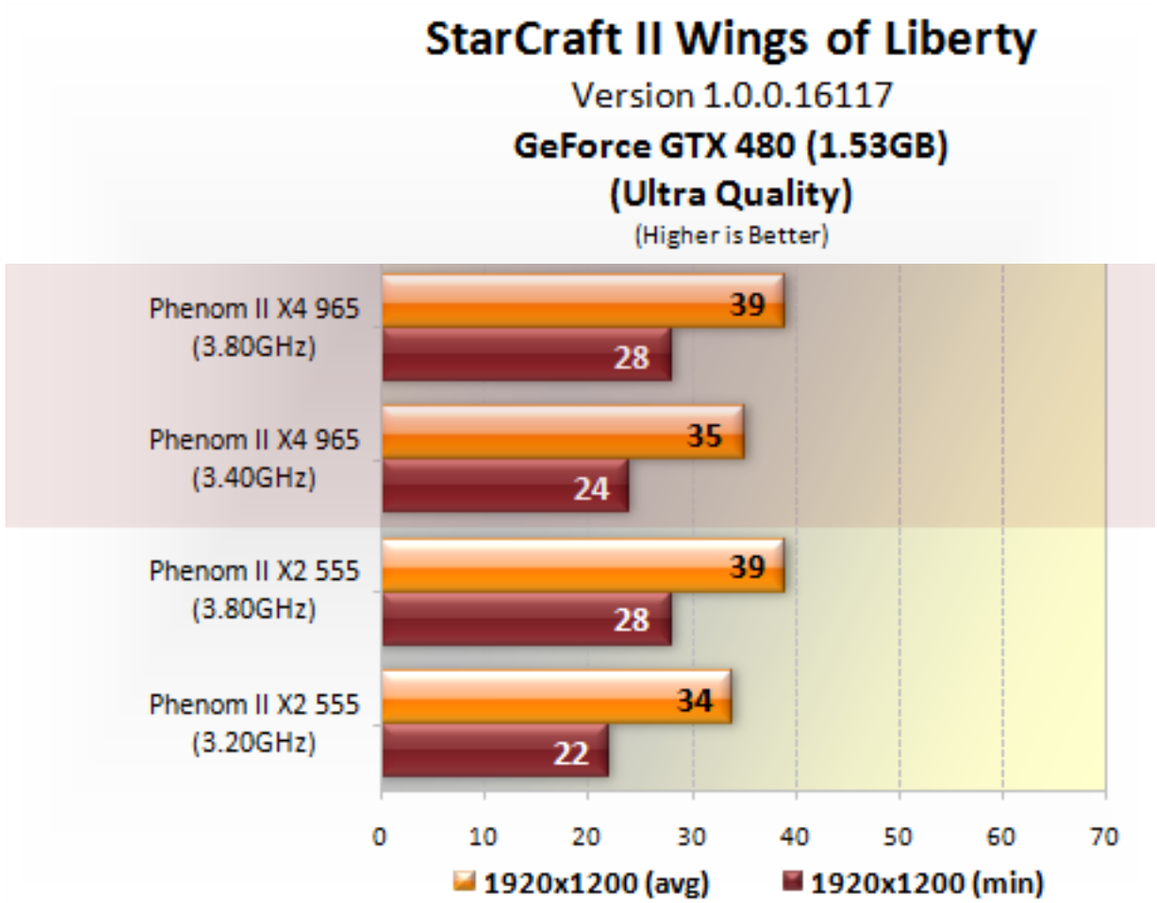
$$\text{Speedup}_{\text{quad}} = \frac{1}{(1- 0.5) + \frac{0.25}{2} + \frac{0.25}{4}} = 1.45$$

Lessons Learned from Amdahl's Law

$$\text{Speedup} = \frac{1}{(1 - \text{Fraction}_{\text{enhanced}}) + \frac{\text{Fraction}_{\text{enhanced}}}{\text{Speedup}_{\text{enhanced}}}}$$

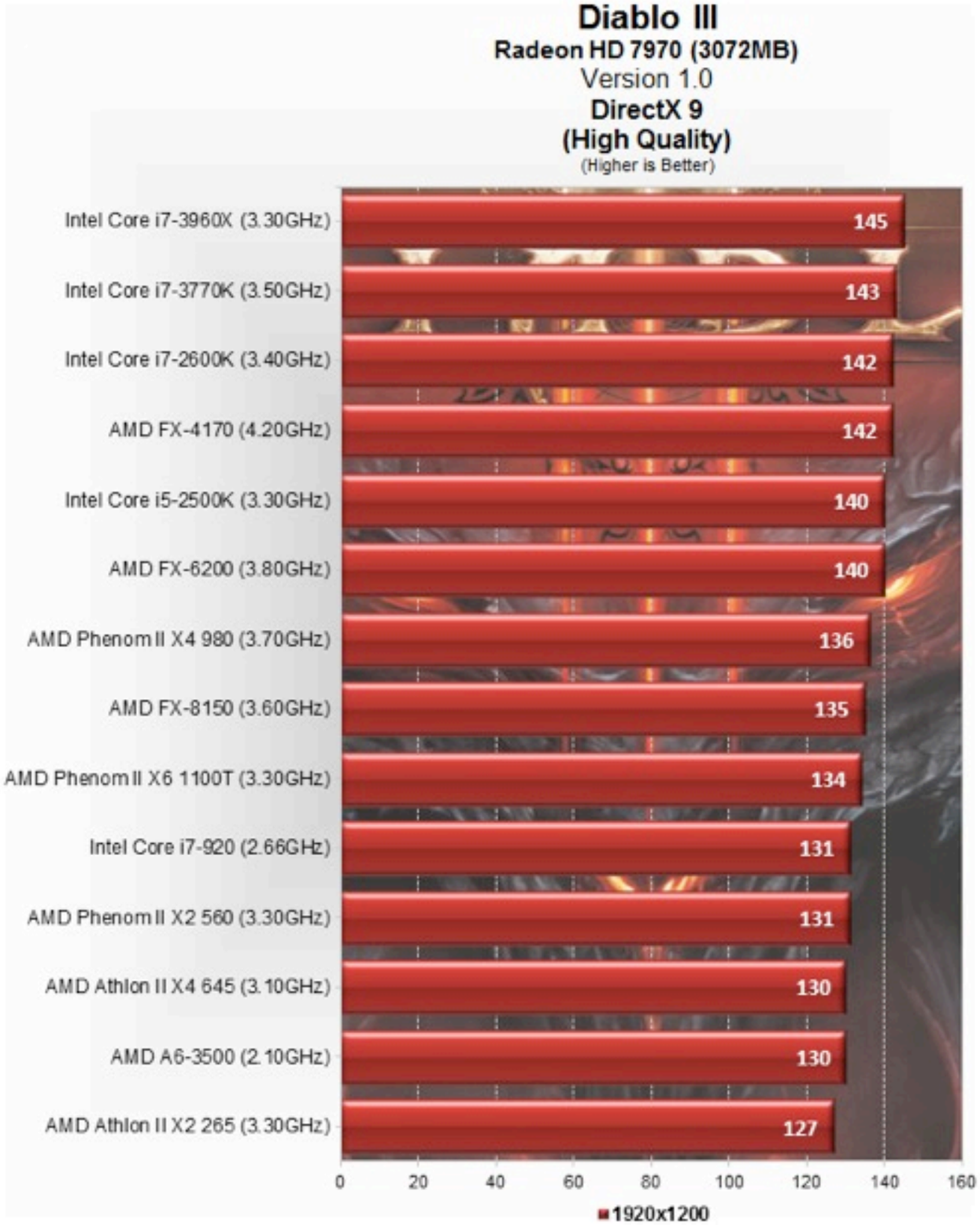
- Make the most "time-consuming" part fast

Case study: StarCraft II



- Adding cores does not always work
  - The application does not scale with the number of cores very well.
- Still help improving overall system performance if you have multiple tasks in the background (like web browsers, IMs...)

Case study: Diablo III



- The CPU is not the main performance bottleneck
  - GPU
  - network
  - storage (loading maps)

Power & Energy

Power

- P=aCV2f
  - a: switches per cycle
  - C: capacitance
  - V: voltage
  - f: frequency, usually linear with V
- Double the clock rate consumes more power than a quad-core processor!
- Packaging of the chip
- Heat dissipation cost

Energy

- Energy = P * ET
- Lower power does not necessary means better battery life if the processor slow down the application too much
- The electricity bill is related to energy!

Double Clock Rate or Double the Processors?

- Assume 60% of the application can be fully parallelized with 2-core or speedup linearly with clock rate. Should we double the clock rate or duplicate a core?

$$\text{Speedup}_{\text{2-core}} = \frac{1}{(1 - 0.6) + \frac{0.6}{2}} = 1.43$$

$$\text{Power}_{\text{2-core}} = 2x$$

$$\text{Energy}_{\text{2-core}} = 2 * [1/(1.43)] = 1.39$$

$$\text{Speedup}_{\text{2XClock}} = 2$$

$$\text{Power}_{\text{2XClock}} = 8x$$

$$\text{Energy}_{\text{2XClock}} = 8 / 2 = 4$$

Other important metrics

Bandwidth

- The amount of work (or data) during a period of time
  - Network/Disks: MB/sec, GB/sec, Gbps, Mbps
  - Game/Video: Frames per second
- Also called “throughput”
- “Work done” / “execution time”

Response time and BW trade-off

- Increase bandwidth can hurt the execution time of a single task
- If you want to transfer 2 Peta-Byte of data from UCLA
  - 125 miles (201.25 km) from UCSD
  - You can use an Internet 2 network with 100Gbps speed
  - 2 Peta-byte over 167772 seconds = 1.94 Days
  - 22.5TB in 30 minutes
  - Bandwidth: 100 Gbps

Or …

- Use a Toyota Prius!
  - 125 miles (201.25 km) from UCSD
  - 75 MPH on highway!
  - 50 MPG
  - Max load: 374 kg = 2,770 hard drives (1TB per drive)
  - 4 hours round-trip
  - Get nothing in first 30 minutes...
  - Bandwidth: 145 GB/sec

- Internet 2 network with 100Gbps speed
  - 2 Peta-byte over 167772 seconds = 1.94 Days
  - 22.5TB in 30 minutes
  - Bandwidth: 100 Gbps = 12.5 GB/sec

Reliability

- Mean time to failure (MTTF)
- Hardware can fail because of
  - Electromigration
  - Temperature
  - High-energy particle strikes

Metrics for marketing

MIPS
(Million Instructions per second)

$$\text{MIPS} = \frac{\text{Instruction Count}}{\text{Execution Time} \times 10^6}$$

$$= \frac{\text{IC}}{\text{IC} \times \text{CPI} \times \text{CycleTime} \times 10^6} = \frac{\text{Clock Rate}}{\text{CPI} \times 10^6}$$

- MIPS does not include instruction count!
  - Cannot compare different ISA/compiler
  - Different CPI of applications, for example, I/O bound or computation bound
  - If new architecture has more IC but also lower CPI?

MIPS
(Million Instructions per second)

	MIPS	clock rate
XBOX 360	19,200	3.2GHz
PS3	230,400	3.2GHz
Core i7	76,383	3.2GHz

MFLOPS (Million FLoating-point Operations Per Second)

	MFLOPS	clock rate
XBOX One	1,228,800	1.6 GHz
PS4	2,900,000	1.6 GHz
Core i7 EE 3970X + AMD Raedon 6990	5,099,000	3.5 GHz

MFLOPS (Million FLoating-point Operations Per Second)

- Share all limitations with MIPS
  - Cannot compare different ISA/compiler
  - Different CPI of applications, for example, I/O bound or computation bound
  - If new architecture has more IC but also lower CPI?
  - Does not make sense if the application is not floating point intensive