电子科技大学格拉斯哥学院
Glasgow College, UESTC

UESTC4019: Real-Time Computer Systems and Architecture

Lecture 4

Performance

How to Measure Performance (1 of 2)

- In evaluating processor hardware and setting requirements for new systems, performance is one of the key parameters to consider, along with cost, size, security, reliability, and, in some cases, power consumption
- It is difficult to make meaningful performance comparisons among different processors, even among processors in the same family.
- Raw speed is far less important than how a processor performs when executing a given application.

How to Measure Performance (2 of 2)

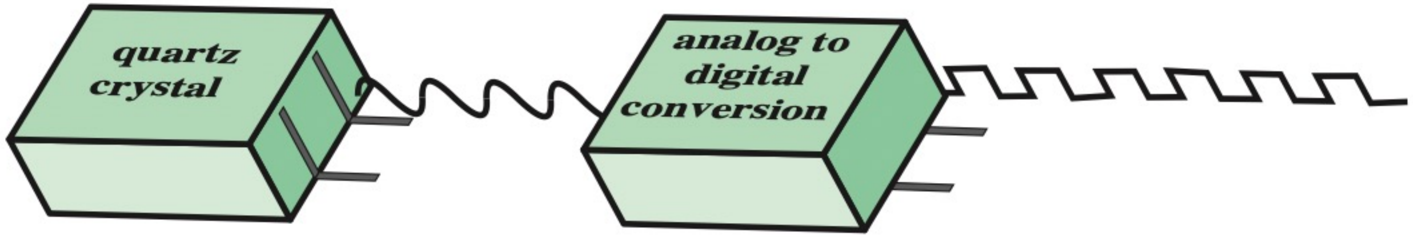
- Unfortunately, application performance depends not just on the raw speed of the processor but also on the instruction set, choice of implementation language, efficiency of the compiler, and skill of the programming done to implement the application
- In this lecture, we look at some traditional measures of processor speed
- We will examine benchmarking, which is the most common approach to assessing processor and computer system performance
- We will discusses how to average results from multiple tests

System Clock (1 of 6)

- Operations performed by a processor, such as fetching an instruction, decoding the instruction, performing an arithmetic operation, and so on, are governed by a system clock
- Typically, all operations begin with the pulse of the clock. Thus, at the most fundamental level, the speed of a processor is dictated by the pulse frequency produced by the clock, measured in cycles per second, or Hertz (Hz)

System Clock (2 of 6)

- Typically, clock signals are generated by a quartz crystal, which generates a constant sine wave while power is applied
- As shown in the Figure, this wave is converted into a digital voltage pulse stream that is provided in a constant flow to the processor circuitry
- For example, a 1-GHz processor receives 1 billion pulses per second



System Clock (3 of 6)

- The rate of pulses is known as the clock rate, or clock speed
- One increment, or pulse, of the clock is referred to as a clock cycle, or a clock tick
- The time between pulses is the cycle time
- The clock rate is not arbitrary but must be appropriate for the physical layout of the processor
- Actions in the processor require signals to be sent from one processor element to another

System Clock (4 of 6)

- When a signal is placed on a line inside the processor, it takes some finite amount of time for the voltage levels to settle down so that an accurate value (logical 1 or 0) is available
- Depending on the physical layout of the processor circuits, some signals may change more rapidly than others
- Thus, operations must be synchronized and paced so that the proper electrical signal (voltage) values are available for each operation

System Clock (5 of 6)

- The execution of an instruction involves a number of discrete steps, such as
  - fetching the instruction from memory
  - decoding the various portions of the instruction
  - loading and storing data
  - performing arithmetic and logical operations

System Clock (6 of 6)

- Most instructions on most processors require multiple clock cycles to complete
- Some instructions may take only a few cycles, while others require dozens
- When pipelining is used, multiple instructions are being executed simultaneously
- A straight comparison of clock speeds on different processors does not tell the whole story about performance

Cycles per Instruction (CPI) (1 of 7)

- A processor is driven by a clock with a constant frequency f or, equivalently, a constant cycle time τ, where

$$\tau = 1/f$$

- The instruction count, I_c, for a program is the number of machine instructions executed for that program until it runs to completion or for some defined time interval
- An important parameter is the average cycles per instruction (CPI) for a program

Cycles per Instruction (CPI) (2 of 7)

- If all instructions required the same number of clock cycles, then CPI would be a constant value for a processor
- However, on any given processor, the number of clock cycles required varies for different types of instructions, such as
  - load
  - store
  - branch

Cycles per Instruction (CPI) (3 of 7)

- Let CPI_i be the number of cycles required for instruction type i, and I_i be the number of executed instructions of type i for a given program. Then we can calculate an overall CPI as follows:

$$CPI = \frac{\sum_{i=1}^{n}(CPI_i \times I_i)}{I_c}$$

- The processor time T needed to execute a given program can be expressed as:

$$T = I_c \times CPI \times \tau$$

Cycles per Instruction (CPI) (4 of 7)

- The processor time *T* can be refined by recognizing that during the execution of an instruction, part of the work is done by the processor, and part of the time a word is being transferred to or from memory
- In this latter case, the time to transfer depends on the memory cycle time, which may be greater than the processor cycle time

Cycles per Instruction (CPI) (5 of 7)

- We can rewrite the processor time *T* equation as:

$$T = I_c \times [p + (m \times k)] \times \tau$$

where

p is the number of processor cycles needed to decode and execute the instruction

m is the number of memory references needed

k is the ratio between memory cycle time and processor cycle time

Cycles per Instruction (CPI) (6 of 7)

- The five performance factors in the preceding equation (I_c, p, m, k, τ) are influenced by four system attributes:
  - The design of the instruction set (known as instruction set architecture)
  - compiler technology (how effective the compiler is in producing an efficient machine language program from a high-level language program)
  - processor implementation
  - cache and memory hierarchy

Cycles per Instruction (CPI) (7 of 7)

- The five performance factors in the preceding equation (I_c, p, m, k, τ) are influenced by four system attributes:

	I_c	p	m	k	τ
Instruction set architecture	X	X			
Compiler technology	X	X	X		
Processor implementation		X			X
Cache and memory hierarchy				X	X

Millions of Instructions per Second (MIPS)

- A common measure of performance for a processor is the rate at which instructions are executed, expressed as millions of instructions per second (MIPS), referred to as the MIPS rate
- We can express the MIPS rate in terms of the clock rate and CPI as follows:

$$\text{MIPS rate} = \frac{I_c}{T \times 10^6} = \frac{f}{CPI \times 10^6}$$