Name : Sajawal Khan

Department : BS(CS)

Semester : 4th

ID # : 14756

Assignment No : 2nd (Chapter# 02)

Subject : Computer Architecture

Submitted To :Muhammad Amin Sir

Q.1 Give answer to each of the following:

A. Discuss different desktop applications that require the great power of contemporary microprocessor-based systems?

Ans:

Different desktop applications that require the great power of contemporary microprocessor based systems are:

- ✓ Image processing
- ✓ Three-dimensional rendering
- ✓ Speech recognition
- ✓ Videoconferencing
- ✓ Multimedia authoring
- ✓ Voice and video annotation of files
- ✓ Simulation modelling

B. Discuss the techniques used in contemporary processors to increase speed?

Ans:

The techniques used in contemporary processors to increase speed are following:

- ★ **Pipelining:**
  Pipelining enables a processor to work simultaneously on multiple instructions by performing a different phase for each of the multiple instructions at the same time.
- ★ **Branch prediction:**
  Branch prediction potentially increases the amount of work available for the processor to execute.
- ★ **Superscalar execution:**
  This is the ability to issue more than one instruction in every processor clock cycle. In effect, multiple parallel pipelines are used.
- ★ **Data flow analysis:**
  The processor analyzes which instructions are dependent on each other's results, or data, to create an optimized schedule of instructions.
- ★ **Speculative execution:**
  This enables the processor to keep its execution engines as busy as possible by executing instructions that are likely to be needed.

C. Discuss the problems created due to increase in clock speed and logic density of the processor?

Ans:

Discuss the problems created due to increase in clock speed and logic density of the processor are:

- ♦ **Power:**
  As the density of logic and the clock speed on a chip increase, so the power density increases and also dissipated the heat.
- ♦ **RC delay:**
  The speed at which electrons can flow on a chip between transistors is limited by the resistance and capacitance of the metal wires connecting them; specifically, delay increases as the RC product increases.
- ♦ **Memory latency:**
  Memory access speed (latency) and transfer speed (throughput) lag processor speeds.

D. Discuss the speedup of a program using multiple processors compared to a single processor using Amdahl's Law?

Ans:

The speedup using a parallel processor with *N* processors that fully exploits the parallel portion of the program is as follows:

Speedup =Time to execute program on a single processor/Time to execute program on *N* parallel processors

$= T(1 - f) + Tf/T(1 - f) + Tf/N = 1/(1 - f) + f/N$

E. Discuss the multicore, MIC, and GPGPU in detail?

Ans:

Multicore:

- The use of multiple processors on the same chip provides the potential to increase performance without increasing the clock rate.
- Strategy is to use two simpler processors on the chip rather than one more complex processor.
- With two processors larger caches are justified.
- As caches became larger it made performance sense to create two and then three levels of cache on a chip.

MIC:

- Leap in performance as well as the challenges in developing software to exploit such a large number of cores.
- The multicore and MIC strategy involves a homogeneous collection of general purpose processors on a single chip.

GPUs:

- Core designed to perform parallel operations on graphics data.
- Traditionally found on a plug-in graphics card, it is used to encode and render 2D and 3D graphics as well as process video.
- Used as vector processors for a variety of applications that require repetitive computations.

Q.2 Solve each of the following:

A. A benchmark program is run on a 60 MHz processor. The execute program consists of 104,000 instruction executions, with the instruction mix and clock cycle count given below. Determine the effective CPI, MIPS rate, and execution time for this program.

Instruction Type	Instruction Count	Cycles per Instruction
Integer arithmetic	46,000	1
Data transfer	33,000	2
Floating point	16,000	2
Control transfer	9000	2

Ans:

Effective CPI:

CPI = (1*46000) + (2*33000) + (2*16000) + (2*9000) / 100

CPI = 162000 / 100

CPI = 1620

MIPS Rate:

MIPS rate = 60 MHz / 1620 * 10_6

MIPS rate = 60 * 10_6 / 1620 * 10_6

MIPS rate = 60 / 1620

MIPS rate = 0.037

Execution Time:

T = Ic / (MIPS * 10^6)

T = 104000 / (0.037 * 10^6)

T = 104000 / 37 * 10^3

T = 2811 * 10^{-3}

T = 2.811 sec

B. Consider two different machines, with two different instruction sets, both of which have a clock rate of 200 MHz. The following measurements are recorded on the two machines running a given set of benchmark programs:

Instruction Type	Instruction Count (millions)	Cycles Per Instruction
Machine A		
Arithmetic and logic	8	1
Load and store	4	3
Branch	2	4
Others	4	3
Machine A		
Arithmetic and logic	10	1
Load and store	8	2
Branch	2	4
Others	4	3

Determine the effective CPI, MIPS rate, and execution time for each machine.

Ans:

For Machine A:

CPI = (1*8 + 3*4 + 4*2 + 3*4) * 10^6 / (8+4+2+4) * 10^6

CPI = 40 * 10^6 / 18 * 10^6

CPI = 2.22

MIPS rate = 200MHz / 2.22 * 10^6

MIPS rate = 200 * 10^6 / 2.22 * 10^6

MIPS rate = 90

T = Ic / (MIPS * 10^6)

T = 18 * 10^6 / 90 * 10^6

T = 0.2 sec

For Machine B:

CPI = (1*10 + 2*8 + 4*2 + 3*4) * 10_6 / (10+8+2+4) * 10_6

CPI = 46 / 24

CPI = 1.92

MIPS rate = 200MHz / 1.92 * 10_6

MIPS rate = 200 * 10_6 / 1.92 * 10_6

MIPS rate = 104

T = Ic / (MIPS * 10_6)

T = 24 * 10_6 / 104 * 10_6

T = 0.23 sec

C. Early examples of CISC and RISC design are the VAX 11/780 and the IBM RS/6000, respectively. Using a typical benchmark program, the following machine characteristics result:

Processor	Clock Frequency (MHz)	Performance (MIPS)	CPU Time (seconds)
VAX 11/780	5	1	12 *x*
IBM RS/6000	25	18	*x*

The final column shows that the VAX required 12 times longer than the IBM measured in CPU time.

a. **What is the relative size of the instruction count of the machine code for this benchmark program running on the two machines?**

Ans:

The MIPS rate could be computed as the following:
MIPS rate = Ic / T * 10_6

Ic = MIPS rate * T * 10_6

Now by computing the ratio of the instruction count of the IBM RS/6000 to the VAX 11/780 which is:

18 * 1x * 10_6 / 1 *12 x * 10_6

= 18 / 12

= **1.5**

b. **What are the CPI values for the two machines?**

Ans:

Regarding to the VAX 11/780, the **CPI** = (5 MHz) / (1 * 10_6) = 5 * 10_6 / 1 * 10_6

= 5 / 1 = **5**

Regarding to the IBM RS/6000, the **CPI** = (25 MHz) / (18 * 10_6) = 25 * 10_6 / 18 * 10_6

= 25 / 18 = **1.4**

D. Consider the example in Section 2.5 for the calculation of average *CPI* and MIPS rate, which yielded the result of *CPI* = 2.24 and MIPS rate = 178. Now assume that the program can be executed in eight parallel tasks or threads with roughly equal number of instructions executed in each task. Execution is on an 8-core system with each core (processor) having the same performance as the single processor originally used. Coordination and synchronization between the parts adds an extra 25,000 instruction executions to each task. Assume the same instruction mix as in the example for each task, but increase the *CPI* for memory reference with cache miss to 12 cycles due to contention for memory.

a. Determine the average *CPI*.

b. Determine the corresponding MIPS rate.

c. Calculate the speedup factor.

d. Compare the actual speedup factor with the theoretical speedup factor determined by Amdahl's law.

Ans:

a. Since we have the same instruction mix, that means the additional instructions for each task could be allocated appropriately between the instruction types. Therefore, the following table be gotten:

Instruction Type	CPI	Instruction Mix
Arithmetic and logic	1	60%
Load/store with cache hit	2	18%
Branch	4	12%
Memory reference with cache miss	12	10%

The **average CPI** = (1× 0.6) + (2 × 0.18) + (4 × 0.12) + (12 × 0.1) = 2.64. Therefore, the CPI has been increased since the time for memory access is also increased.

b. **MIPS** = 400/2.64 = 152. There is a corresponding drop in the MIPS rate.

c. The speedup factor equals to the ratio of the execution times. The execution time is calculated as the following: T = Ic / (MIPS * 10_6).

For the one processor, T1 = (2 * 10_6) / (178 * 10_6) = **11 ms**.

For the 8 processors, each processor executes 1/8 of the 2 million instructions plus the 25,000

T8 = 2 * 10_6 ÷ 8 + 0.025 * 10_6 / 152 * 10_6

T8 = **1.8 ms**

Therefore we have

Speedup = Time to execute program on a single processor /

Time to execute program on N parallel processor

Speedup = 11 / 1.8

Speedup = 6.11

d. By depending on the information given, it is not obvious how to quantify this effect in Amdahl's equation. Therefore, if it is supposed that the fraction of code ,which is parallelizable, is f = 1, then Amdahl's law decreases to Speedup = N = 8. Therefore, the actual speedup is only about 75% of the theoretical speedup.