COMPUTER ARCHITECTURE

Home Assignment – 1

VISHNU KARTHIK RAVINDRAN

vravindr@syr.edu

SUID – 985286667

**Case Study for the IBM 360:**

The paper “Architecture of the IBM 360” explains the architecture, design problems incurred in the process to make the system design compatible, remedial measures for those design issues, and comprehensive details on the data format, storage assignments and I/O controls. This architecture had included ambitious concepts at that time in developing a generalized design to support the families of computers. The I/O devices were improvised assisting variety of systems and the floating-point word length was increased to 36 bits. Internal CPU functions were optimized to provide larger time for other applications to execute. This architecture had a greater performance, and, they achieved their goal of developing the system capable of supporting the innovations which come through the next few years.

The decisions which were taken on microcode, character bits, field specifications and increasing floating point word length during design phase put forth the architecture in the right path for compatibility. The architect’s primary objective was to create a common design to be applied in various fields like scientific computing, communications and various businesses.

Some of the features of this architecture still in existence are the microcode algorithms for machine independent codes, 32-bit ISA used with 16/64-bit ISA’s. One of the most interesting fact is that codes written in the IBM 360 could still execute in the latest mainframes with minor variations.

**Case study related to Reduced Instruction Set Computer:**

In the paper “The Case for the Reduced Instruction Set Computer”, Patterson and Ditzel proposes that RISC can be cost effective than CISC architecture and argues that CISC produces more harm than their counterpart. The authors claim that RISC can be effectively implemented in the VLSI computers. They argue that the adoption of complex instruction in the early days were because of the following reasons: speed gap in memory and CPU, advancements in integrated circuit memories leading to cost effective approach in designing complex architecture for microprogrammed control, expensive memories shrinking code size, marketing gimmick to use the latest technology in products to lure customers, upward compatibility, support for high level languages. In CISC, when a multiprogramming concept is applied, it saves its state in the shadow registers and the microcode complexity increases. This issue is not found on a system without CISC.

The writers opine that the compiler writers mostly do not utilize the complex instructions embedded in the architecture set. There are various consequences that are being covered in the paper on the implementation of CISC on machines. Firstly, the speed of the memory(cache) have become equivalent to the CPU speed. Secondly, the execution time of the instructions in RISC is far better than the CISC instruction set. Thirdly, the design time and design errors of processors are higher for CISC.

The authors also state that RISC architecture supports VLSI computers. In the electronics industry, the design time is very critical, and RISC can be easily implemented compared to CISC. As we had seen in the earlier paragraphs, the RISC speed is better compared to the CISC. They also claim that the chip area utilized by RISC is lower leading to implementation of additional functionalities in the VLSI computers. The author puts forward that execution of certain instructions are little worse in RISC architecture for the compilers than CISC which he defends it by saying that it can also be rectified in design in the future.

Nowadays, most of the processors are pretty much RISC based. In the current CISC architecture, still the Performance issue listed above persists. The popularity of the RISC instruction set will still grow as the RISC -V, the latest ISA, has become open instruction set architecture and hosted in GitHub. This will lead to further innovations in the domain with the contribution from experts.

**“B) [10] [Max 1 page] Visit the Intel on‐line microprocessor museum1 and determine the rate of increase in transistor counts and clock frequencies in the 70's, 80's, 90's, 00’s, and this decade. Also, create a plot of the number of transistors versus technology feature size using an MS Excel spreadsheet. “**

|  |  |  |
| --- | --- | --- |
| **Processor** | **Feature Size** | **Transistors** |
| **Intel 4004** | 10000 | 2300 |
| **Intel 8008** | 10000 | 3500 |
| **Intel 8080** | 6000 | 4500 |
| **Intel 8086** | 3000 | 29000 |
| **Intel 286** | 1500 | 134000 |
| **Intel 386** | 1500 | 275000 |
| **Intel 486** | 1000 | 1200000 |
| **Intel Pentium** | 800 | 3100000 |
| **Intel Pentium Pro** | 350 | 5500000 |
| **Intel Pentium II** | 250 | 7500000 |
| **Intel Celeron** | 250 | 7500000 |
| **Intel Pentium III** | 250 | 9500000 |
| **Intel Pentium 4** | 180 | 42000000 |
| **Intel Pentium M** | 90 | 55000000 |
| **Intel Core 2 Duo - 2006** | 65 | 291000000 |
| **Intel Core 2 Duo -2008** | 45 | 410000000 |
| **Intel Atom Processor** | 45 | 470000000 |
| **2nd gen** | 32 | 1160000000 |
| **3rd gen** | 22 | 1400000000 |

|  |  |  |
| --- | --- | --- |
| **Years** | **Transistor Count** | **Frequency** |
| **Max(70s)** | 29000 | 5 x 106 |
| **Max(80s)** | 1200000 | 25 x 106 |
| **Max(90s)** | 9500000 | 600 x 106 |
| **Max(00s)** | 410000000 | 2.4 x 109 |
| **Max(10s)** | 1400000000 | 3.8 x 109 |

Growth in transistors:

. 70s to 80s = ((1200000-29000)/29000) \*100 = 4038%

• 80s to 90s = ((9.5\*106-1.2\*106)/ 1.2\*106) \*100 = 692.6%

• 90s to 00s = ((410\*106-9.5\*106)/ 95\*106) \*100 = 4216%

• 00s to 10s = ((1.4\*109-410\*106)/ 95\*106) \*100 = 241%

Growth in frequency:

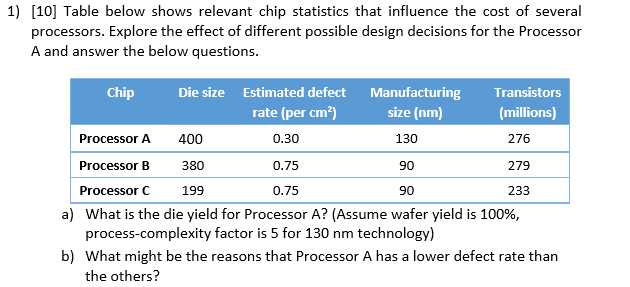
• 70s to 80s = ((25\*106-5\*106)/ 5\*106) \*100 = 400%

• 80s to 90s = ((600\*106-25\*106)/ 25\*106) \*100 = 2300%

• 90s to 00s = ((2.4\*109-600\*106)/ 600\*106) \*100 = 300%

• 10s to 00s = ((3.8\*109-2.4\*109)/ 2.4\*109) \*100 = 58%

**QUESTION:**



**SOLUTION:**

1. **Die Yield = Wafer yield x (1 + (Defects per unit area x Die Area))-a**

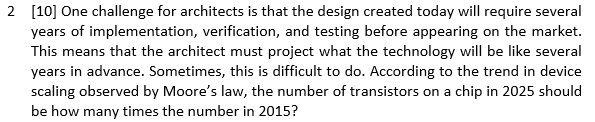
Note: The unit of Die size is not given, so considered unit as mm2

1 x (1+ (0.30 x 4))-5

**Ans: 0.0191**

1. From the table, we could infer that the manufacturing size for processor A is greater than processors B and C. When the manufacturing size is higher, the fabrication complexity is lower in that processor. In turn, the defect rate will be lower in those processors with lower fabrication complexities. This might be one of the reason for the processor A having lower defect rate compared to processor B and C.

**QUESTION:**



**SOLUTION:**

According to Moore’s law, the transistor doubles every 2 years. If the number of transistors in 2015 is X, then the number of transistors in 2025 is (2^5) X .

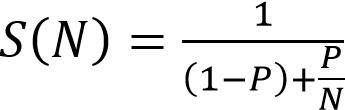
**Ans: 32 times**

**QUESTION:**

“3 [10] When parallelizing an application, the ideal speedup is speeding up by the number of processors. This is limited by two things: percentage of the application that can be parallelized and the cost of communication. Amdahl’s law considers the former but not the latter. What is the speedup with N processors if 50% of the application is parallelizable, ignoring the cost of communication? What will be the speedup for a system with 1000 processors? 4”

**SOLUTION:**

**Amdahl’s law =**

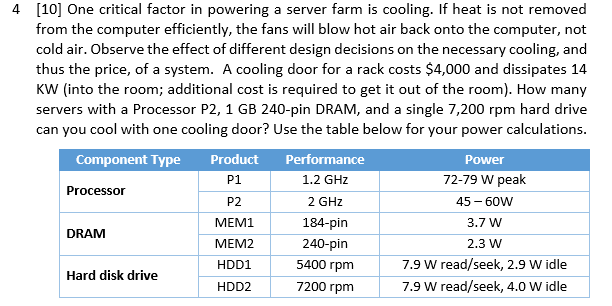


**Given: P = 0.5**

Speedup for N processors = 1/ ((1-0.5) +0.5/N) = **N / (0.5N + 0.5)**

Speedup for 1000 processors = 1/ ((1-0.5) +0.5/N) = 1000 / (0.5 x 1000 + 0.5) = **1.998**

**QUESTION**:

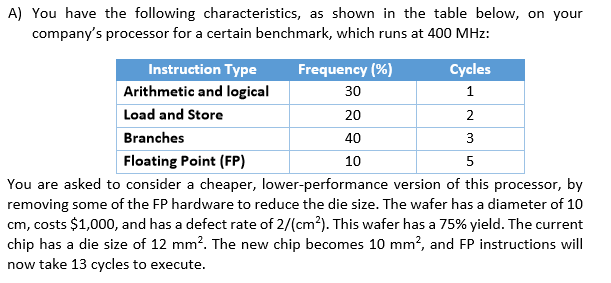


**SOLUTION:**

Peak power consumption = 60 + 2.3 + 7.9 = 70.2 W

Number of servers = 14000W/70.2W= 199.4 ~ **199 systems**

**QUESTIONS:**



1. “[10] What are the old and new CPI (Cycles Per Instructions) and MIPS (Million Instructions Per Second) ratings running this benchmark? “
2. “[10] What are the old and new die yields? What are the old and new costs per (working) processor? Please comment on the overall effect of the proposed hardware change on the cost and the performance of the processor. (Assume process-complexity factor is 4) “
3. “[10] What would be the theoretical limit of the best possible overall speedup that we could ever get by only improving the FP unit, and what would be the CPI and MIPS ratings of this new processor?”

**SOLUTION:**

1. CPI for old processor = (30 % x 1) + (20% x 2) + (40% x 3) + (10% x 5)

= **2.4 cycles/instruction**

CPI for new processor = (30 % x 1) + (20% x 2) + (40% x 3) + (10% x 13)

= **3.2 cycles/instruction**

MIPS for old processor = 400/ (2.4)

**= 166.67 MIPS**

MIPS for new processor = 400/ (3.2)

**= 125 MIPS**

1. **Old processor:**

**Dies per wafer = (pi x (Wafer diameter/ 2) ^2)/ Die Area- ((pi x Wafer diameter)/ sqrt (2 x Die Area))**

= (3.14 x 5^2)/0.12 – (3.14 x 10) / sqrt (2 x0.12) = **590**

**Die Yield = Wafer yield x (1 + (Defects per unit area x Die Area))-a**

0.75 x (1+ (2 x 0.12))-4  = **0.317**

**Cost of die = cost of wafer/ (dies per wafer x die yield)**

= (1000/ (590 x 0.317)) = $ **5.35**

**New processor:**

**Dies per wafer = (pi x (Wafer diameter/ 2) ^2)/ Die Area- ((pi x Wafer diameter)/ sqrt (2 x Die Area))**

(3.14 x 5^2)/0.10 – (3.14 x 10) /sqrt (2 x0.10) = **715**

**Die Yield = Wafer yield x (1 + (Defects per unit area x Die Area))-a**

0.75 x (1+ (2 x 0.10)/4)-4  = **0.361**

**Cost of die = cost of wafer/ (dies per wafer x die yield)**

= (1000/ (715 x 0.361)) = $ **3.87**

Note: Die size is given in mm2 so converted to cm2 => 12 mm2 to 0.12 cm2

From the above expressions, I could infer that the cost of the new processor decreased when the die size was decreased from 12mm2  to 10mm2. Hence Die size is proportional to the cost of the die. As the hardware trimmed down, the MIPS and CPI lowered down eventually reducing the performance.

1. **CPI for new processor = Sum (frequency\*clock cycle)**

= (30 % x 1) + (20% x 2) + (40% x 3) + (10% x 1)

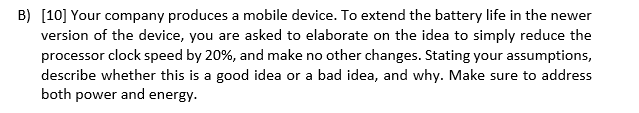
= **2 cycles/instruction**

**Fraction Enhanced = CPI of FP/ CPI of Processor** = 5/(1+2+3+5) = 5/11

**Overall Speedup = 1/((1-fenhanced) +(fenhanced/speedupenhanced))**

=1/((1-(5/11)) + (5/11)/5) = **1.57 times**

**MIPS for new processor = 400/ 2 X 106 = 200 MIPS**



I would say that the decrease in clock speed to increase the mobile battery life will not be a wise decision. I agree that the CPU operating at a higher clock rate dissipates more power, but the performance of the device will also be affected. In our case, the 20% reduction in clock speed reduces the speed by 20%. We can also say that the CPU power is directly proportional to the clock speed. Users will only opt for performance centric products in the market. Therefore, it is not good to undermine the clock speed. There are many examples[a] in the current world market where the manufacturers throttle the clock speed, resulting in the decline of mobile performance for the users. We also know that the power is defined as the energy consumed per unit time. So, power dissipation will depend on the reduced speed and energy consumption. If time taken by the processor to complete the task surpasses the energy consumption, then there won’t be any encouraging effect on the power consumption. I would also suggest the company for various sensible solutions like “overclocking” and “undervolting” to increase the battery life without changing the processor.

**REFERENCES:**

1. <https://en.wikipedia.org/wiki/List_of_Intel_Core_i3_microprocessors>
2. <https://www.intel.com/content/www/us/en/history/museum-transistors-to-transformations-brochure.html>
3. <https://www.androidpit.com/how-to-overclock-android>
4. <https://www.addictivetips.com/windows-tips/how-to-undervolt-a-cpu/>
5. <http://bgr.com/2017/12/11/iphone-slow-battery-replacement-fix/>
6. <http://www.intel.com/content/www/us/en/history/historic-timeline.html>
7. <http://www.intel.com/content/www/us/en/history/museum-transistors-to-transformations-brochure.html>
8. <http://www.intel.com/content/www/us/en/company-overview/intel-museum.html>