Sanjeev Penupala

CS 3340.502

1/24/19

HW #1

1. List and briefly describe the 7 "layers" involved in using computers to solve problems.

   Problem – Something you want to do on the computer (i.e. surfing the web)

   Algorithm – These exist to convert the problem into something the program can execute

   Program – Executes the problem

   ISA (Instruction Set Arch) – The language where the microprocessor can understand the problem (in 0's and 1's). It is the interface the takes what the software wants and converts into what hardware can understand and execute

   Microarchitecture – Where microprocessors are implemented, giving you "chips". Has designed structures to it

   Circuits – Implements the structures from microarchitecture

   Electrons – Produced by the circuits, giving voltage differences going from one potential to another to solve what the problem is asking for

2. Briefly describe microprocessor evolution from 1971 to today in terms of number of transistors and processor speed.

   According to Moore's Law, transistors in a chip will double every 2-3 years. Looking at 1971, we had a Intel 4004 chip with 2300 transistors on it, with a processing speed of 106 KHz. As the years went by, transistors and processing speed kept on increasing and as recorded in 2013, we have over 5 billion transistors, with a processing speed of above 5

GHz, which is an insane amount for our current technology. If the trend continues, we will continue to get advanced technology like this, with better processing speeds by the month.

3. As time went on, did more and more transistors go towards processing or memory?

   It went towards the on-chip memory

4. What significant change in processor design caused a more balanced use of transistors between cache and processing?

   People invent a design called “Core Dual”, which gives you a second processor on the chip. And you can keep doubling the “core” over time, to give chips more processing power and keeping it balanced

5. What is cache?

   On-chip memory

6. Now that we have multi-core processors, have we learned how to program them? Why or why not?

   No because after the use of a lot of cores, it is difficult to program them because there is too many processors on the chip that we don’t know how to program, on top of high energy consumption. Processors we see contain identical processors.

Sanjeev Penupala

CS 3340.502

1/24/19

7. Describe Professor Patt's predictions for the next phase (Phase 3) of microprocessor design.

   He believes that we will sooner or later have "variable processors", which are heterogeneous in their application. Some of the parts of the chip would be reconfigurable, meaning depending the application, the use of it changes. Given our problem of high-energy consumption, some parts will be powered off during application run-time.

8. What is Step 1 of Professor Patt's vision of the future of microprocessors?

   Break the Layers – Meaning there must be awareness between the 7 layers and how they are working. By doing so, it can make one layer's job easier to do

9. What good things does Professor Patt believe will follow step 1?

   - More than one interface
   - "Organic" run-time systems
   - ILP Cores

10. Programs can be written on a platform that is easy to use but may not have great performance, or they can be written in a more challenging way for optimal performance. Discuss the trade-offs of each approach.

    For the first one, since you are trading off performance for easy of use, it can get your job done faster, but won't understand what is going down in the layers below them, which requires them to need a software layer to bridge the multi-core.

Sanjeev Penupala

CS 3340.502

1/24/19

For the second one, since you are trading off ease of use for utmost performance, you will understand what happens at the CPU/GPU level, but trade-off the amount of time and challenge of learning how the lower layers work.