УДК 004.23.25.4
Anastasiia Beinyk[1], Nataliia Zhukova[2]
[1]student of group CST-120, National University «Zaporizhzhia Polytechnic»
[2]PhD (Philology), assistant prof. National University «Zaporizhzhia Polytechnic»

IS ARM CPU GOING TO REPLACE X86?

The ARM architecture is nothing new because mobile phones and embedded devices have used mostly it almost since the start of the smartphone era. However, since Apple has decided to twist its equipment and to replace the x86 architecture with ARM, nowadays it is on the lips of all hardware enthusiasts. Seeing that Apple is the one who always sets trends, people are wondering if ARM CPUs going to replace x86.

Someone could question this because it sounds as if mobile CPUs could replace PCs. Will ARM architecture be good enough to replace x86? Apple Silicon M1 processor answer this question itself, which runs faster than desktop Intel CPUs and nearly as fast as AMD's Ryzen 5000 series, the current performance crown. It is custom silicon, which makes Apple Macbooks the current fastest laptops in the world.

Firstly, it is highly interesting if the ARM can do the same task as x86. Actually, the answer is 'yes' as ARM could do the same assignments, but it has some very important nuances that involve the "in what way", and especially the "in how long" (in terms of performance) as the x86 architecture is massive and has more than a thousand instructions, some of which are very complex. This approach is called Complex Instruction Set Computing (CISC). While ARM's RISC (Reduced Instruction Set Computing) philosophy is much simpler, and intentionally so. The design goal here is to build simple designs that are easy to manage, with a focus on power efficiency, too. However, today this comparison is becoming pointless as both design approaches copy and use the best parts of each other.

As a result, an ordinary user, who uses a laptop for web surfing, photo and video editing and works with a huge amount of files, would not notice any difference. Nevertheless, there is one for those people who use old applications that were meant for x86 because they need to be recompiled to run on ARM as well. Thus, it should be born in mind that all the software must be adapted or ported, and this is something that not all companies can do, nor all companies can afford. It is something that currently has no viability, not even in the medium term; in the long term, it is something that could happen, but we cannot expect it soon. However, even that can usually run through x86 emulation, which Windows is starting to support.

For developers, there are a lot of differences in how applications get compiled, but these days, most compilers do a good job of supporting the major instruction

sets, and you will not have to make any changes to get it compiling for multiple platforms.

However, the point of changing x86 to ARM architecture is mainly all about speed. ARM and x86 are both instruction sets, which are a list of micro-code "programs" that the CPU supports. CPUs execute operations sequentially, like a machine given a list of tasks to do. Each instruction is known as an opcode, and architectures like x86 have a lot of opcodes. That is why x86 is known as a "Complex Instruction Set," or CISC.

These architectures generally take the design approach of packing a lot of stuff into a single instruction. While RISC basically does away with complex multi-part instructions. Each instruction mostly can execute in a single clock cycle. As CPUs need to load all their instructions from RAM and execute them as fast as possible. It turns out it is far easier to do that when you have many simple instructions versus a lot of complex ones. The CPU runs faster when the instruction buffer can be filled up, and that is a lot easier to do when the instructions are smaller and easier to process. Which makes ARM much faster in daily use.

To sum up, is ARM CPU going to replace x86? Mostly yes, especially on laptops. But this is not to say that x86 will die off anytime soon, it will still have its consumers, for instance, mainframe which does not need high speed. But we should not neglect the benefits of RISC architectures, which is the way of the future.