

**Faculty of engineering - Shoubra**

**Benha University**

**Research Article / Research Project / Literature Review**

in fulfillment of the requirements of

|  |  |
| --- | --- |
| **Department** | **Engineering Mathematics and physics** |
| **Division** | **------------** |
| **Academic Year** | **2019-2020preparatory** |
| **Course name** | **Computer** |
| **Course code** | **ECE001** |

**Title: -**

**Computer Architecture.**

By:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Name | Edu mail | B.N |
| 1 | محمد عبدالنبى عبدالعظيم محمود | muhammed195849@feng.bu.edu.eg | 760 |

**Approved by:**

|  |  |
| --- | --- |
| Examiners committee | Signature |
| Dr.Ahmed Bayoumy |  |
| Dr.Shady Elmashad |  |
| Dr. Abdelhamid Attaby |  |

TheWEBSITE

<html>

<body>

<h1>computerarchitecture main page </h1>

<h2>links</h2>

<ul>

<li><a href="THE main page .html ">computer mainpage </a></li>

<li><a href="computer architecture definition .html ">computer architecture definition </a>

</li>

<li><a href="computer architecture introduction.html ">computer architecture introduction </a>

</li>

<li><a href="computer architecture history.html" >computer architecture history </a>

</li>

</ul>

History:

The first documented computer architecture was in the correspondence between Charles Babbage and Ada Lovelace,

describing the analytical engine.When building the computer Z1 in 1936, Konrad Zuse described in two patent applications

for his future projects that machine instructions could be stored in the same storage used for data, i.e., the stored-programconcept. Two other early and important examples are:

John von Neumann's 1945 paper, First Draft of a Report on the EDVAC, which described an organization of logical elements; and

Alan Turing's more detailed Proposed Electronic Calculator for the Automatic Computing Engine, also 1945 and which cited John von Neumann's paper.

Definition:

Computer architecture is concerned with balancing the performance, efficiency, cost,

and reliability of a computer system. The case of instruction set architecture can be used to illustrate the balance of these competing factors

More complex instruction sets enable programmers to write more space efficient programs,since a single instruction can encode some higher-level abstraction (such as the x86 Loop instruction).

However, longer and more complex instructions take longer for the

processor to decode and can be more costly to implement effectively.

The increased complexity from a large instruction set also creates more room for unreliability when instructions interact in unexpected ways.

<img src="computerarchitecture..gif " alt="computerarchitecture ">

</body>

<html>

computerarchitecture main page

links

computer mainpage

computer architecture definition

computer architecture introduction

computer architecture history

History: The first documented computer architecture was in the correspondence between Charles Babbage and Ada Lovelace, describing the analytical engine.When building the computer Z1 in 1936, Konrad Zuse described in two patent applications for his future projects that machine instructions could be stored in the same storage used for data, i.e., the stored-programconcept.[3][4] Two other early and important examples are: John von Neumann's 1945 paper, First Draft of a Report on the EDVAC, which described an organization of logical elements;[5] and Alan Turing's more detailed Proposed Electronic Calculator for the Automatic Computing Engine, also 1945 and which cited John von Neumann's paper.[6] Definition: Computer architecture is concerned with balancing the performance, efficiency, cost, and reliability of a computer system. The case of instruction set architecture can be used to illustrate the balance of these competing factors More complex instruction sets enable programmers to write more space efficient programs,since a single instruction can encode some higher-level abstraction (such as the x86 Loop instruction). However, longer and more complex instructions take longer for the processor to decode and can be more costly to implement effectively. The increased complexity from a large instruction set also creates more room for unreliability when instructions interact in unexpected ways. computerarchitecture





**Research objectives**

**1-knowing introduction to Computer Architecture.**

**2-knowing the historyof the Computer Architecture**

**3-knowing the definition of computer engineering .**

**4-knowing the definition of Computer Architecture.**

**5-knowing the Implementation of Computer Architecture.**

Computer

**Abstract**

**1-The computer is one of the most important means of technology in this age.**

**2-Computer architecture is concerned with balancing the performance, efficiency, cost, and reliability of a computer system**

**3-There is no company or institution or house that does not have a computer.**

**4-An instruction set architecture (ISA) is an abstract model of a computer. It is also referred to as architecture or computer architecture. A realization of an ISA, such as a central processing unit (CPU), is called an implementation.**

**5-Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development. There is some overlap with the disciplines of systems analysis, systems architecture and systems engineering.**

**6-All people in all ages are in need of using the computer in their daily lives.**

**7-Companies and institutions do not spare the computer in keeping employees’ data and keeping files and information about work and other functions.**

**Table of contents**

|  |  |
| --- | --- |
| **Subject / section** | **Page** |
| **Research objectives** |  |
| **Abstract** |  |
| **Introduction** |  |
| **Literature Review** |  |
| **Results and discussion** |  |
| **Conclusions-References** |  |

**Computer Architecture.**

**Introduction**

In computer engineering, computer architecture is a set of rules and methods that describe the functionality, organization, and implementation of computer systems. Some definitions of architecture define it as describing the capabilities and programming model of a computer but not a particular implementation.[1] In other definitions computer architecture involves instruction set architecture design, microarchitecture design, logic design, and implementation.[2]



**Literature Review**

What is Computer Architecture?

Computer architecture can mean different things to different people. For this blog we take a “big tent” approach with a broad definition of computer architecture. Fundamentally, it is the art and science of designing computer systems (including hardware and software) to satisfy society’s insatiable appetite for computational resources. I like to use the tagline “from sub-atomic particles to data centers” to capture the many orders of magnitude in scale (~20) that influence (and are influenced by) computer architecture. Emerging technologies can create computational and storage elements with sub-nanometer dimensions, cloud computing requires reasoning about systems with hundreds of thousands of computational resources in a data center, and individual chips contain almost 10 billion transistors. Exploiting the opportunities and/or addressing challenges within and across these scales is an important aspect of computer architecture. Furthermore, application and system software (broadly defined) are intimately tied to computer system design in a symbiotic relationship. New applications/software inspires new hardware and new hardware enables new applications or systems software capabilities. However, the choice between maintaining or breaking existing abstractions creates tension in computer system design. Computer architecture guides the decision process by considering technology advances/limitations and software/application demands.



History:

The first documented computer architecture was in the correspondence between Charles Babbage and Ada Lovelace, describing the analytical engine. When building the computer Z1 in 1936, Konrad Zuse described in two patent applications for his future projects that machine instructions could be stored in the same storage used for data, i.e., the stored-program concept.[3][4] Two other early and important examples are:

John von Neumann's 1945 paper, First Draft of a Report on the EDVAC, which described an organization of logical elements;[5] and

Alan Turing's more detailed Proposed Electronic Calculator for the Automatic Computing Engine, also 1945 and which cited John von Neumann's paper.[6]

The term “architecture” in computer literature can be traced to the work of Lyle R. Johnson and Frederick P. Brooks, Jr., members of the Machine Organization department in IBM's main research center in 1959. Johnson had the opportunity to write a proprietary research communication about the Stretch, an IBM-developed supercomputer for Los Alamos National Laboratory (at the time known as Los Alamos Scientific Laboratory). To describe the level of detail for discussing the luxuriously embellished computer, he noted that his description of formats, instruction types, hardware parameters, and speed enhancements were at the level of “system architecture”, a term that seemed more useful than “machine organization”.[7]

Subsequently, Brooks, a Stretch designer, opened Chapter 2 of a book called Planning a Computer System: Project Stretch by stating, "Computer architecture, like other architecture, is the art of determining the needs of the user of a structure and then designing to meet those needs as effectively as possible within economic and technological constraints.

Brooks went on to help develop the IBM System/360 (now called the IBM zSeries) line of computers, in which “architecture” became a noun defining “what the user needs to know”.Later, computer users came to use the term in many less explicit ways.[8]

The earliest computer architectures were designed on paper and then directly built into the final hardware form. Later, computer architecture prototypes were physically built in the form of a transistor–transistor logic (TTL) computer—such as the prototypes of the 6800 and the PA-RISC—tested, and tweaked, before committing to the final hardware form. As of the 1990s, new computer architectures are typically "built", tested, and tweaked—inside some other computer architecture in a computer architecture simulator; or inside a FPGA as a soft microprocessor; or both—before committing to the final hardware form.[9]

Computer architecture is concerned with balancing the performance, efficiency, cost, and reliability of a computer system. The case of instruction set architecture can be used to illustrate the balance of these competing factors. More complex instruction sets enable programmers to write more space efficient programs, since a single instruction can encode some higher-level abstraction (such as the x86 Loop instruction).However, longer and more complex instructions take longer for the processor to decode and can be more costly to implement effectively. The increased complexity from a large instruction set also creates more room for unreliability when instructions interact in unexpected ways.

The implementation involves integrated circuit design, packaging, power, and cooling. Optimization of the design requires familiarity with compilers, operating systems to logic design, and packaging.[10]

Implementation:

Once an instruction set and micro-architecture have been designed, a practical machine must be developed. This design process is called the implementation. Implementation is usually not considered architectural design, but rather hardware design engineering. Implementation can be further broken down into several steps:

Logic implementation designs the circuits required at a logic-gate level

Circuit implementation does transistor-level designs of basic elements (e.g., gates, multiplexers, latches) as well as of some larger blocks (ALUs, caches etc.) that may be implemented at the log-gate level, or even at the physical level if the design calls for it.

Physical implementation draws physical circuits. The different circuit components are placed in a chip floorplan or on a board and the wires connecting them are created.

Design validation tests the computer as a whole to see if it works in all situations and all timings. Once the design validation process starts, the design at the logic level are tested using logic emulators. However, this is usually too slow to run realistic test. So, after making corrections based on the first test, prototypes are constructed using Field-Programmable Gate-Arrays (FPGAs). Most hobby projects stop at this stage. The final step is to test prototype integrated circuits, which may require several redesigns

For CPUs, the entire implementation process is organized differently and is often referred to as CPU design.



figure(2):Block diagram of a basic computer with uniprocessor CPU. Black lines indicate data flow, whereas red lines indicate control flow. Arrows indicate the direction of flow.

**Results and discussion**

after showing these details ,we must ask ourselves some questions and give us some answers, the first questionwhat is the importance of the Computer engineering?

Computer engineering (CE) is a branch of engineering that integrates several fields of computer science and electronic engineering required to develop computer hardware and software. Computer engineers usually have training in electronic engineering (or electrical engineering), software design, and hardware-software integration instead of only software engineering or electronic engineering. Computer engineers are involved in many hardware and software aspects of computing, from the design of individual microcontrollers, microprocessors, personal computers, and supercomputers, to circuit design. This field of engineering not only focuses on how computer systems themselves work but also how they integrate into the larger picture

Usual tasks involving computer engineers include writing software and firmware for embedded microcontrollers, designing VLSI chips, designing analog sensors, designing mixed signal circuit boards, and designing operating systems. Computer engineers are also suited for robotics research, which relies heavily on using digital systems to control and monitor electrical systems like motors, communications, and sensors.

In many institutions of higher learning, computer engineering students are allowed to choose areas of in-depth study in their junior and senior year because the full breadth of knowledge used in the design and application of computers is beyond the scope of an undergraduate degree. Other institutions may require engineering students to complete one or two years of general engineering before declaring computer engineering as their primary focus.[11]-[12]-[13]-[14]

**Conclusions**

1-computer architecture is a set of rules and methods that describe the functionality, organization, and implementation of computer systems.

2-The first documented computer architecture was in the correspondence between Charles Babbage and Ada Lovelace, describing the analytical engine.

**3-Modern computer performance is often described in instructions per cycle (IPC), which measures the efficiency of the architecture at any clock frequency; a faster IPC rate means the computer is faster.**

**4-There are two main types of speed: latency and throughput. Latency is the time between the start of a process and its completion. Throughput is the amount of work done per unit time**

**5-Companies and institutions do not spare the computer in keeping employees’ data and keeping files and information about work and other functions.**

**6-There is no company or institution or house that does not have a computer.**

**7-The computer has made life easier for us in general.**

**No one does not use the computer.**

**8-The computer is used by students in the study,Where the student uses the computer to do research and to search for information useful in the study.**

**References**

**[1]-[2]-Hennessy, John; Patterson, David. Computer Architecture: A Quantitative Approach (Fifth ed.). p. 11. This task has many aspects, including instruction set design, functional organization, logic design,and implementation.**

**[3]-"Electronic Digital Computers", Nature, 162: 487, 25 September 1948, doi:10.1038/162487a0, archived from the original on 6 April 2009, retrieved 2009-04-10**

**[4]-Susanne Faber, "Konrad Zuses Bemuehungen um die Patentanmeldung der Z3", 2000**

**[5]-Neumann, John (1945). First Draft of a Report on the EDVAC. p. 9.**

**[6]-Reproduced in B. J. Copeland (Ed.), "Alan Turing's Automatic Computing Engine", Oxford University Press, 2005, pp. 369-454.**

**[7]-Buchholz, Werner (1962). Planning a Computer System. p. 5.**

**[8]-"System 360, From Computers to Computer Systems". IBM100. Retrieved 11 May 2017.**

**[9]-Schmalz. "Organization of Computer Systems". UF CISE. Retrieved 11 May 2017**

**[10]-Martin, Milo. "What is computer architecture?" (PDF). UPENN. Retrieved 11 May 2017.**

**[11]-[12]-[13]-[14]-Trinity College Dublin. "What is Computer System Engineering". Retrieved April 21, 2006.,**

**"Changing Majors @ Clemson". Clemson University. Retrieved September 20, 2011.**

**"Declaring a College of Engineering Major". University of Arkansas. Archived from the original on October 12, 2014. Retrieved September 20, 2011.**

**"Degree Requirements". Carnegie Mellon University. Retrieved September 20, 2011.**

**"Programas de Materias" (in Spanish). Universidad Católica Argentina.**