IJIRSET
International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY

e-ISSN: 2319-8753 | p-ISSN: 2347-6710

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 10, Issue 7, July 2021

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 7.569

9940 572 462 | 6381 907 438 | ijirset@gmail.com | www.ijirset.com

International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)

| e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 7.569|

|| Volume 10, Issue 7, July 2021 ||

|DOI:10.15680/IJIRSET.2021.100321 |

A Study on Von Neumann and Harvard Architectures

Koppula Geetha[1]

Assistant Professor, Department of ECE, SVS Group of Institutions, India[1]

ABSTRACT: An instruction set, or instruction set architecture (ISA), is the part of the computerarchitecture related to programming, including the native data types, instructions, registers,addressing modes, memory architecture, interrupt and exception handling, and external I/O. An ISA includes a specification of the set of opcodes (machine language), and the native commands implemented by a particular processor. To run the application, when power is first turned ON, the microprocessor addresses a predefined location and fetches, decodes, and executes the instruction one after the other. This paper provides a detailed study on von neumann and harvardarchitectures.

KEYWORDS: embedded systems, von Neumann architecture, Harvard architecture

I. ARCHITECTURES

Von Neumann Architecture

The Von Neumann architecture was first proposed by a computer scientist John von Neumann. In this architecture, one data path or bus exists for both instruction and data. As a result, the CPU does one operation at a time. It either fetches an instruction from memory, or performs read/write operation on data. So an instruction fetch and a data operation cannot occur simultaneously, sharing a common bus.

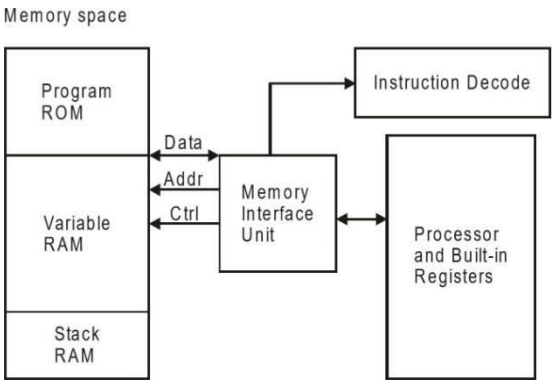

Figure 1: Von-Neumann Architecture

Von-Neumannarchitecturesupportssimplehardware.Itallowstheuseofasingle,sequential memory. Today's processing speeds vastly outpace memory access times, and we employ a very fast but small amount of memory (cache) local to theprocessor.

Harvard Architecture

The Harvard architecture offers separate storage and signal buses for instructions and data. This architecture has data storage entirely contained within the CPU, and there is no access to the instruction storage as data. Computers have separate memory areas for program instructions and data using internal data buses, allowing simultaneous access to both instructions and data.

Programs needed to be loaded by an operator; the processor could not boot itself. In a Harvard architecture, there is no need to make the two memories share properties.

| e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 7.569|

|| Volume 10, Issue 7, July 2021 ||

|DOI:10.15680/IJIRSET.2021.100321 |

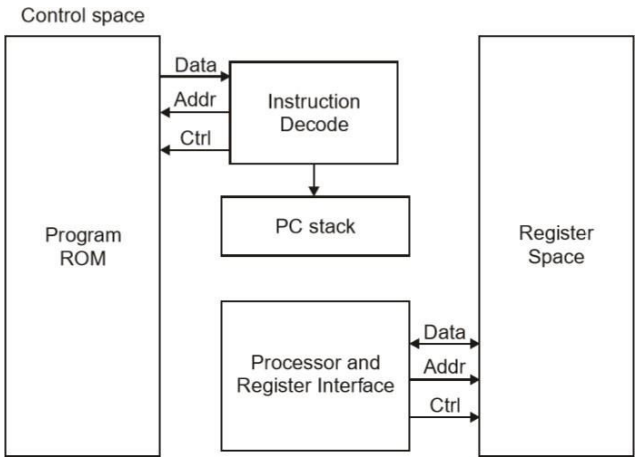

Figure 2: Harvard Architecture

II. PERFORMANCE ANALYSIS AND OPTIMIZATION

Performance or efficiency measures

Performance is a measure of the results achieved. Performance efficiency is the ratio between effort expended and results achieved. The difference between current performance and the theoretical performance limit is the performance improvement zone.

Another way to think of performance improvement is to see it as improvement in four potential areas. First, is the resource INPUT requirements (e.g., reduced working capital, material, replacement/reorder time, and set-up requirements). Second, is the THROUGHPUT requirements, often viewed as process efficiency; this is measured in terms of time, waste, and resource utilization. Third, OUTPUT requirements, often viewed from a cost/price, quality, functionality perspective. Fourth, OUTCOME requirements, did it end up making a difference.

Performance improvement at the operational or individual employee level usually involves processes such as statistical quality control. At the organizational level, performance improvement usually involves softer forms of measurement such as customer satisfaction surveys which are used to obtain qualitative information about performance from the viewpoint ofcustomers.

The difficulty is the significant variability one encounters when running the program.

What input data?

What hardware platform?

What compiler?

What compiler options>

We can focus on several major areas:

Complexity Time
Power consumption

Memory size

Cost

Weight
Other considerations include

Development time

| e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 7.569|

|| Volume 10, Issue 7, July 2021 ||

|DOI:10.15680/IJIRSET.2021.100321 |

Ease of maintenance

Extensibility

A system is a set of interacting or interdependent components forming an integrated whole[1] or a set of elements (often called 'components') and relationships which are different from relationships of the set or its elements to other elements or sets

Complexity analysis involves breaking down a user task into a set of constituent steps and then calculating a complexity metric for each step in the task relative to the type of user. Complexityanalysis can also be used to provide relative comparisons of complexity between releases of a product.

The complexity of an algorithm is a function describing the efficiency of the algorithm in terms of the amount of data the algorithm must process. Usually there are natural units for the domain and range of this function. There are two main complexity measures of the efficiency of an algorithm:

Time complexity is a function describing the amount of time an algorithm takes in terms of the amount of input to the algorithm. "Time" can mean thenumberof memory accesses performed, the number of comparisons between integers, the number of times some inner loop is executed, or some other natural unit related to the amount of real time the algorithm will take. We try to keep this idea of time separate from "wall clock" time, since many factors unrelated to the algorithm itself can affect the real time (like the language used, type of computing hardware, proficiency of the programmer, optimization in the compiler, etc.). It turns out that, if we chose the units wisely, all of the other stuff doesn't matter and we can get an independent measure of the efficiency of the algorithm. Space complexity is a function describing the amount of memory (space) an algorithm takes in terms of the amount of input to the algorithm. We often speak of "extra" memory needed, not counting the memory needed to store the input itself. Again, we use natural (but fixed-length) units to measure this. We can use bytes, but it's easier to use, say, number of integers used, number of fixed-sized structures, etc. In the end, the function we come up with will be independent of the actual number of bytes needed to represent the unit. Space complexity is sometimes ignored because the space used is minimal and/or obvious, but sometimes it becomes as important an issue astime.

III. PERFORMANCE OPTIMIZATION

In order to improve the performance, we must look at the common mistakes that are often made when assessing and trying to improve performance

Common mistakes

Expecting improvement in one aspect of the design to improve the overall performance proportional to improvement.

Using hardware independent metrics to predict performance. Using peak performance
Comparing performance based on a couple of metrics Using synthetic benchmarks.

High demand input or output operations, with tight timing constraints and high throughput.
Streaming applications including high speed audio and video. With such application, delays in the time domain translate directly to distortion in the frequency domain.

Caches and performance

A CPU cache is a cache used by the central processing unit of a computer to reduce the average time to access memory. The cache is a smaller, faster memory which stores copies of the data from frequently used main memory locations. As long as most memory accesses are cached memory locations, the average latency of memory accesses will be closer to the cache latency than to the latency of main memory.

The proportion of accesses that result in a cache hit is known as the hit rate, and can be a measure of the effectiveness of

| e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 7.569|

|| Volume 10, Issue 7, July 2021 ||

|DOI:10.15680/IJIRSET.2021.100321 |

the cache for a given program oralgorithm.

Read misses delay execution because they require data to be transferred from memory much more slowly than the cache itself. Write misses may occur without such penalty, since the processor can continue execution while data is copied to main memory in the background.

Instruction caches are similar to data caches, but the CPU only performs read accesses (instruction fetches) to the instruction cache. (With Harvard architecture and modified Harvard architecture CPUs, instruction and data caches can be separated for higher performance, but they can also be combined to reduce the hardware overhead.)

In computer science, a cache is a component that transparently stores data so that future requests for that data can be served faster. The data that is stored within a cache might be values that have been computed earlier or duplicates of original values that are stored elsewhere. If requested data is contained in the cache (cache hit), this request can be served by simply reading the cache, which is comparatively faster. Otherwise (cache misses), the data has to be recomputed or fetched from its original storage location, which is comparatively slower.Hence,the greater the number of requests that can be served from the cache, the faster the overall system performance becomes.

To be cost efficient and to enable an efficient use of data, caches are relatively small. Nevertheless, caches have proven themselves in many areas of computing because access patterns in typical computer applications have locality of reference. References exhibit temporal locality if data is requested again that has been recently requested already. References exhibit spatial locality if data is requested that is physically stored close to data that has been requestedalready.

Small memories on or close to the CPU can operate faster than the much larger main memory. Most CPUs since the 1980s have used one or more caches, and modern high-end embedded, desktop and server microprocessors may have as many as half a dozen, each specialized for a specific function. Examples of caches with a specific function are the D-cache and I-cache (data cache and instruction cache).

IV. CONCLUSION

Performance improvement is the concept of measuring the output of a particular process or procedure, then modifying the process or procedure to increase the output, increase efficiency, or increase the effectiveness of the process or procedure. The concept of performance improvement can be applied to either individual performance such as an athlete or organizational performance such as a racing team or a commercialenterprise.This paper provided a detailed study on von neumann and harvardarchitectures.

REFERENCES

1. ISO (2011) Robots and robotic devices - Safety requirements for industrial robots - Part 2: Robot systems andintegration.
2. Setchi RM, Lagos N (2004) Reconfigurability and reconfigurable manufacturing systems: state-of-the-art review.INDIN.
3. BrusselHV,WynsJ,PaulV,LucB,Patrick(1998)Referencearchitecture for holonic manufacturing systems: PROSA. Computers in Industry 37: 255-274.
4. Onori M, Akillioglu H, Hofmann A (2011) An evolvable robotic assemblycell.
5. Roopha Shree Kollolu Srinivasa, “DEVELOPMENTS IN WIRELESS NETWORKING TECHNOLOGY AND STUDY ON GERMAN RESEARCHER TEST 40 GBPS WIRELESS BROADBAND”, Wutan Huatan Jisuan Jishu, Volume XIV, Issue II, February2018.
6. Roopha Shree Kollolu Srinivasa, “REPRESENTATIONOFMAN-IN-MIDDLEATTACK AND WLAN SECURITYATTACKS”, “Science, Technology and Development “, Volume VIII Issue XII DECEMBER 2019.
7. Roopha Shree Kollolu Srinivasa, “HISTORY, DEPLOYMENT AND SERVICE MODELS TOWARDS THE EVOLUTION OF CLOUD COMPUTING”, Journal of Interdisciplinary Cycle Research, Volume XII, Issue III, March2020.
8. Roopha Shree Kollolu Srinivasa, “INFRASTRUCTURAL CONSTRAINTS OF CLOUDCOMPUTING”, International Journal of Management, Technology And Engineering, Volume X, Issue XII, DECEMBER2020.

| e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 7.569|

|| Volume 10, Issue 7, July 2021 ||

|DOI:10.15680/IJIRSET.2021.100321 |

9. Roopha Shree Kollolu Srinivasa, "A REVIEW ONWIDE VARIETY AND HETEROGENEITY OF IOT PLATFORMS", The International journal of analytical and experimental modal analysis, Volume XII, Issue I, January2020
10. Roopha Shree Kollolu Srinivasa, "RISKANALYSIS OF PUTTING ATTACKS INTO PERSPECTIVE AND CONDUCTING A VULNERABILITYASSESSMENT", "Science, Technology and Development ", Volume VIII Issue XII DECEMBER 2019
11. Roopha Shree Kollolu Srinivasa, "TECHNOLOGIES AND ISSUES OF CLOUD COMPUTING", Journal of Interdisciplinary Cycle Research, Volume XIII, Issue II, February2021
12. Roopha Shree Kollolu Srinivasa, "CHARACTERISTICS, APPLICATIONS AND USE CASES OF CLOUD COMPUTING", International Journal of Management, Technology And Engineering, Volume X, Issue VI, JUNE2020
13. Roopha Shree Kollolu Srinivasa, "A REVIEW ON THE ADVANTAGES AND TYPES OF WIRELESS NETWORKS", JAC : A Journal Of Composition Theory, Volume X, Issue II, 2017
14. Roopha Shree Kollolu Srinivasa, "CLASSIFICATIONS OF WIRELESS NETWORKING AND RADIOTRANSMISSION TECHNOLOGY", Wutan Huatan Jisuan Jishu, Volume XIV, Issue XI, November2018
15. Roopha Shree Kollolu Srinivasa, "A REVIEW ON THE COMPARISON OF CLOUD COMPUTING DEPLOYMENT MODELS", JASC: Journal of Applied Science and Computations, Volume VIII, Issue VII, July2021
16. Roopha Shree Kollolu Srinivasa, "RECENT RESEARCH DIRECTIONS TOWARDS INTERNET OF THINGS", Wutan Huatan Jisuan Jishu Journal, Volume XVI, Issue I, January2020
17. Roopha Shree Kollolu Srinivasa, "AN OVERVIEW ON THE IOT RESEARCH CHALLENGES", JASC: Journal of Applied Science and Computations, Volume VI, Issue VII, JULY2019
18. Roopha Shree Kollolu Srinivasa, "A STUDY ON THE DIFFERENCES BETWEEN IOT AND TRADITIONAL NETWORK", JASC: Journal of Applied Science and Computations, Volume VI, Issue II, February2019
19. Roopha Shree Kollolu Srinivasa, "WLAN TOTOLOGY AND COMPARISON BETWEEN WIRED AND WIRELESS NETWORK", Parishodh Journal, Volume VI, Issue V, May2017
20. Wang H, Zhang H (2010) A distributed and interactive system to integrated design and simulation for collaborative product development. Robotics and Computer-Integrated Manufacturing 26:778-789
21. Makrisa S, Michalosa G, Eytanb A, Chryssolourisa G (2012)Cooperating Robots for Reconfigurable Assembly Operations. Procedia CIRP 3: 346-351.
22. Hvilshoj M, Bogh S (2011) Little Helper-An autonomous industrial mobile manipulator concept. Internation Journal of Advanced Robotic Systems 8:80-90.

INNO SPACE
SJIF Scientific Journal Impact Factor

Impact Factor:
7.569

doi cross ref

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details