International Journal of Multidisciplinary and Current Educational Research (IJMCER)
ISSN: 2581-7027 ||Volume|| 3 ||Issue|| 3 ||Pages 59-64 ||2021||

Novel Approach for Computer Architecture Simulation

Mr. S. V. Pattalwar[1], Dr. V. M. Thakare[2]
[1] *Research Scholar, SGBAU, Amravati, Maharashtra, India.*
[2] *Professor & Head, Dept. of Computer Sc., SGBAU, Amravati, Maharashtra, India.*

ABSTRACT: In computer design and development novel simulation is a powerful way of acquiring and predicting system behaviors. While designing new simulation has been extensively utilized for system performance and evaluation, design space exploration, and system assessment, it may involve a high cost. Specifically, the complete simulation time is calculated by raw speed and how the instruction format is designed. The speed of the simulator may be improved by changing different parameters using simulators with hardware assistance also using simulators with high-level abstraction speed can be improved, further research also managed to decrease the total number of instructions to be simulated but this may also decrease accuracy in execution. It is also observed that most of the research is develop upon logical techniques. This research work discussed the new techniques and implications learned from these observations

KEYWORDS: Design simulation, architectural space exploration (ASE), regression, mathematical methods.

I. INTRODUCTION

In designing a processor, the planning must estimate the complete behavior of the execution and manufacturing and the system must also meet particular architectural aims decided by planer. As one of the most novel techniques for addressing this issue, simulation has been extremely useful since it offers planner a complete balance of all the design parameters [1]. Application scopes of architectural simulation include but are not limited to performance evaluation, functional validation, design space exploration (DSE), and assessment of architectural innovations. Now a day, all new system designer has designed great simulators to the design of processor. For example, Intel maintains simulation environment [2], which is designed for Intel systems, ranging from embedded system (e.g., Intel's 32-bit embedded [3]) to supercomputer. AMD designed CimNow simulator [4] that emulates CMB Athlon 64 and CMB microprocessor systems. IBM also uses TAsin [5] to calculate further system products. Design simulation may play a vital role in educational research of computer systems; since it enables validations of novel research ideas and the real chips are not designed specifically. One great evidence is that in most of the research papers published in important journals of system design used design simulators to validate and prove their concepts. And their importance has been widely accepted, the speed of architectural simulation is very slow. Particularly, the speed of a new simulator is typically between 2 KIPS (thousand instructions per seconds) and 2 MIPS (million instructions per second), and these are very slow as compared to advance simulators used in the research. To make the simulation faster new simulation techniques are proposed, which includes the high abstraction level. (e.g., Sniper [7], [8]), To increase the speed of simulation various techniques describe includes assistance of hardware to the simulator and use of graphite simulators and use of many processors in the single design. An overview of the above simulators can be found in [6] and [8]. Though there is increase in the speed of simulation but this speed is not yet acceptable as compared to demand of time and is not very useful for the large and multiple task to complete in less amount of time, e.g. to execute thousands of task to execute in the very less amount of time. This is because this time is calculated not only as a simulators speed but also by the complete set of instructions executed in the processor.

II. BACKGROUND

Many researchers studied the various parameters on design of simulators to design best system for the last many decades. Such design simulations are discussed below. In traditional design of simulators, initially traces (i.e., dynamic execution of time ordered stream of instructions) are collected from virtual systems. These instructions are considered as inputs to these simulators to determine the nature of target architecture, where logical simulation is differentiated from complete simulation of timing. As systems are not logically compiled in these simulators, it may produce a great benefit of speed as compared to the older simulators [1].These may combine the logical behavior and detailed space behavior together to produce the great simulated results. Many simulators considered compiled logical instructions set instead of a space as the important parameter, and the modern processors can be designed using this technique. The simulators are initially assembled, debugged and then documented before their release. Several processors are also being developed using the same technique which is considered as the most efficient technique for the development of modern processors.

Some simulators are very dynamic, superscalar in design of the simulation and implemented multithreading processing. The main concentration in the development of simulator is to increase speed simulator, make the instruction compliable and executable, and may provide many other facilities for designer. [6] employs the various tool to produce space information which can be feed the memory design for the manufacture of the memory system on different simulation. Since the systems are logically designed, this may increase the speed of processor as well as simulation.

The paper is organized as follows:
Section 1 Introduction. **Section 2** discusses Background. **Section 3** discusses previous work. **Section 4** discusses existing methodologies. **Section 5** discusses attributes and parameters and how these are affected on various architectures. **Section 6** isproposed method. **Section 7** *is outcome* and result. **Section 8** is conclusion. Finally**Section** 9 is future scope of this analytical paper.

III. PREVIOUS WORK DONE

In this research literature, many system designs and modeling technologies have been observed to provide improved parameters in the design of processor.The designed parameters are very broad, including, assessment of design innovations, logical validation, accuracy calculation, DSE, and tuning issues.

Design Space Exploration:DSE is extremely useful parameter in the increase the speed of simulation for the design of new and latest processors and to facilitate with great features and increase the use of the processor for the wide used. [2].This parameter is also useful in dealing with the various issues and resolve the complexity and the further improvements in the processor can be easily being adopted using the same parameter and reduces the amount of time of instruction compilation and execution and also improves the overall performance of the targeted system to be developed for the use and is very useful for the educational institutes and academia. This makes the system more accurate and flexible and portable.

Assessment of System Innovations: There are many challenges in the development of simulation of the processor and the system. This parameter is very useful to reduce these challenges and make system more convincing and reliable considering the network and bandwidth issues this parameter will help to produce a great simulation environment and is very important in the design of barrier free environment for the design and development of the processor and make the complete system more useful [2], These systems had been developed to calculate the effectiveness of such systems which can produce a great result.

Software Performance Debugging and Tuning: This parameter supports the various task related to the checking of the simulation system and various properties that designer wanted to be deployed in the production of the complete simulation system.

IV. EXISTING METHODOLOGIES

Since from many years the researchers from the domain of system and software for the processor designed are trying to increase the convenience of the system to facilitate the user with the many facilities such as multitasking and improving the overall speed of the system simulation by designing the great speed simulation, they are making use of various parameters as discussed and there is a great tradeoff between these parameters that may eventually be very significant in the development of such efficient system.For the sake of these issues the simulator can be classified in various categories depending on the environment in which they can be utilized.

Classification of Simulators based on Details of Environment of Simulation

Functional Simulators : These are implementation simulators and mainly focus on design architecture of the simulation process. They work exactly as emulator, their speed of working is usually very high, as a program runs on the simulator. Also they are unable to work on the minute architecture. While planning advance sets of commands, different simulators may be used for checking purposes. On the other hand these types of simulators will help in observing the design characteristic of a process during its execution**,** for example, the number of various kinds of commands in a process, memory storage space, etc.

Timing Simulators : Timing simulators, also known as performance simulators, simulate the minute designs of processors. They produce detailed mathematics of the timing/performance of main target architecture [2]. For example, in case of the simulation of a processor, this data might consist of issues like performance of a memory system instructions per cycle (IPC), program run time, and other details of minute architecture-related

mathematics. These simulators are categorized on the basis of level of exact simulation process**:** simulators driven by event, and simulator based on interval.

Cycle-level Simulators: These types of simulators are very fast as compared to other older versions of the simulators and they produce a high simulation quality. The main focus of these types of simulators is the cycle of the event. They act accurately and carefully on each cycle of the event and produce the better and faster simulation process for different types of the processor designs.

Event-driven Simulators: An event-driven simulator simulates main target based on characteristics instead of cycles. These simulators focus on saving the time of the simulation process precisely when it work on the schedule of the event. While working on the schedule of the event it directly jump to the start of the event rather than middle or end of an event depending on the queue of the event. In this way these simulators can significantly improves the time factor needed for the simulation process. The main issue to note here is often literature does not differ between cycle-level and characteristic-level simulators.

V. ANALYSIS AND DISCUSSION

Parallel sampled simulation is again a beautiful simulator which reduces the time required for simulation of each part of the simulation process. During this process check pointing is used during sampling of events environment. Also many samples of the event can be sample at the same time instance hence this process is known as parallel sampled simulation. [3]. On the other hand, it is very difficult to replace ubiquitous multiple core processors, it is very interesting to use multithreaded environment during the parallel sampled simulation.In case of simulating multiple core design of the processor the process may be divided into various threads, where single section can be mapped to important core.Speed and accuracy are again the two different dimensions of the simulation when the processor simulates the event in a multithreading environment. In this kind of simulation process the environment can be mapped for each single portion of the simulation and this can create a disturbance in the actual correctness achieved by using the parallel environment. To resolve this issue cycle-by-cycle mapping can be relaxed to more than single step as a tradeoff between simulation speed and correctness [3], [4].

Comparison of Recent X86 Simulators: Six simulators (gem5 [2], MARSSx86 [4], Multi2Sim [3], Sniper [6], PTLsim [5] and ZSim [7]) are selected for a comprehensive study due to following reasons:

- All of the above simulators may have different simulation designs, but all of them can be included into the same kind of timing simulators.
- These are more accurate simulators with very active design, except PTLsim. PTLsim is passive environment for development, but it is in use still.

All these simulators support x86 and other important designs.
It also has the ability to perform minute simulation on any area of system.

Architecture Modeling: Architecture designs in Sesame, which mainly operate on the transaction level [4], [5], simulate the various issues on accuracy of the computation and characteristics generated by an design of application.These simulators mainly work for the accuracy parameter and will not focus on the functional behavior of the simulation process. The important reason behind this is the features declared in the application design process and hence can be applicable to the various environment of the simulation process. Template models can be constructed and used during design and processing the cores, Interconnecting phases, and different kinds of storage. The structure of the design will specify which block to be applied when depending on situation. In other words design of the structure will work as the blue print for the simulation process and greatly help the complete process of simulation.

Mapping: In the process of mapping describes how the simulation process is carried out and also defined different processes according to their characteristics of the software used in the simulation. The multiple processes are mapped in a single design section. There is a great co-ordination between different process used in the simulation and will help the whole simulation process and reduces the time of simulation. The various linking of the logical processes and the different component are defined in the design of the simulation process.
Interconnection channels—i.e., the channels for temporary storage in the linking layer—are also defined on the design. For instance, single storage is used in shared memory, while the second buffer is used onto a point-to-point channel between various processorsThis process tries to achieve the deadlock free processing and mechanism for the simulation can be easily achieved as each process receives the dedicated resource in the simulation process. The mechanism also takes care that different instances of the resources can be managed

properly and focus is on the deadlock free execution of the processes and eventually reduces the time required for the simulation process. The process of simulation also ensures that the mechanism should be very secure for the execution. As long as interconnection characteristics cannot be sent, the virtual system blocks. This is surely possible because the mapping stage implements in the same environment as the model of design. Hence, both the stages and the design shared the same environment. This also implies that, each time a logical system sends an application event to the design, the logical processor is locked in simulated environment until the event's latency has been simulated for the design. When the details of the implementation are defined they can be refined for the simulation process. The refinement is done at multiple levels and can be link achieve the more accuracy in the simulation process. The dataflow graphs are greatly useful and indicate the flow of simulation.

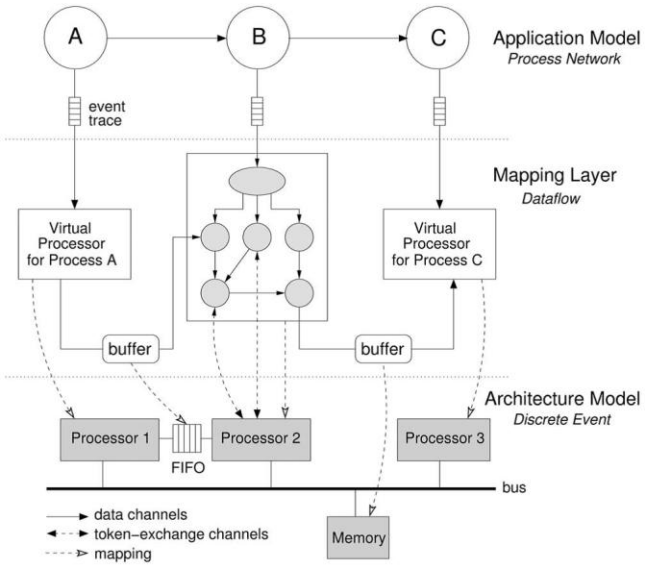

Fig.1.Sesame's Application Model Layer, Architecture Model Layer, and Mapping Layer which Interfaces between Application and Architecture Models[5].

VI. PROPOSED METHODOLOGY

In this research, the technique and methodology is described for architect of Computer system design to apply simulation environment to assess various design factors, check new research areas and analyze the accuracy of different design sections.

The Environment Simulation and Process : In the simulation procedure is the sequence of different stages that the architect must define to compile and execute the simulations. This technique divides the environment into the six stages shown in Fig. 1. Of these six stages, no older research focused specifically on improving the performance of implication a processor enhancement (Stage 2). Consequently, the following subsections focus on Stages up to 6 starting from 1.

This process is categorized in following stages.

Stage1: Design perspectives - Architecture under considerations

Stage 2: methodologies applicable

Stage 3: Modeling, parameter selection

Stage 4: Validation

Stage 5: performance evaluation

Stage 6: Accuracy

Fig. 2: Simulation Process

Use of PB Aspect : This research defines many aspects of the architectural behavior of the considered as the crucial in the study of simulation. Changes in these issues may affect the simulation and also affect the accuracy of the simulation process which are computed by the P&B matrix and the simulation is done considering various parameters, then some of the important parameters are measured by calculating and giving the specific ranks known as rank of effect. Eventually, the final simulation is achieved known as benchmark for simulation.The below section describes the correctness of evaluation of architectural model of previously describe fig 1 using the parameter PB and also describes when and how to apply subset as stated.

This model also describes the different characteristics related to the bottleneck issues which can easily measure by applying PB architecture. These issues are categorized and logical function is applied on the issues to calculate the rank of matrix. Thus rank of matrix is very useful in determining the benchmark simulation process.
Specifically defining, the recommendations may compromise:
a. choosing the values of parameters of issues,
b. selecting a model for defining a subset and
c. analyzing how the improvement affects the design and correctness.
Thus the several changes, in the different parameters significantly affect the simulation procedure and reduce the total number of simulations. One can easily calculate the design and correctness bottlenecks, and gives logical details of the great impact on design improvements.

OUTCOME AND RESULTS

Considering simulation issues:
Tc1 = time required for core 1 to Compiled event n1 with no. of actual Clocks & I1 no of instructions
MIPS = represents the MIPS speed per one no. of core
IPC = Instructions per clock (was set to 0.5)
F = processor speed in GHz
N = no of threads
HTR = host to target ratio
C1 = average cycles for N1threads for c1 no of cores
SPEEDUP = speedup gained by improving no of cores
S_{overal}= SPEEDUP considering all cores
Sn (due to core) = SPEEDUP due to nth core
Fe = SPEEDUP affected due to system Overheads
MJPS system = changes in the system

Design Improvement-I : After allotting starting from letter A and ending to letter Symbol G to above issues we got the below table.

Table 1. PB Design Parameters from A to G

A		B		C		D		E		F		G	
-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1
1	5	1	5	3	7	3	7	4	7	1	7	1	5

Design improvements - II

Table 2. PB Design Matrix with Foldover(X = 8) for Parameters A to G

A	B	C	D	E	F	G	Execution Time
+1	+1	+1	-1	+1	-1	-1	9
-1	+1	+1	+1	-1	+1	-1	11
-1	-1	+1	+1	+1	-1	+1	20
+1	-1	-1	+1	+1	+1	-1	10
-1	+1	-1	-1	+1	+1	+1	9
+1	-1	+1	-1	-1	+1	+1	74
+1	+1	-1	+1	-1	-1	+1	7
-1	-1	-1	-1	-1	-1	-1	112
-1	-1	-1	+1	-1	+1	+1	17
+1	-1	-1	-1	+1	-1	+1	76
+1	+1	-1	-1	-1	+1	-1	6
-1	+1	+1	-1	-1	-1	+1	31
+1	-1	+1	+1	-1	-1	-1	19
-1	+1	-1	+1	+1	-1	-1	33
-1	-1	+1	-1	+1	+1	-1	6
+1	+1	+1	+1	+1	+1	+1	4
-34	-224	-96	-202	-110	-170	32	

The above table represents how the changes in the different parameters change the values of the constraints and issues of the simulation process. Higher values indicate improvement in the simulation quality and lower values represents the reduction.PB and other values affect the various parameters on simulation environment and also the correctness of the simulation process..

VII. CONCLUSION

This research presented a great environment for the simulation process which provides the high quality architectural model and analysis of different methodologies and tools for design level correctness for the calculation of different embedded modes of computer system architecture. This research also provides features for selecting the basic design framework for simulation using PB architectural methodology by using logical modeling and categorization of multiple objectives.However, these basic designs can be simulated using many several simulation environments and using different procedure for simulation using different stages of architectural improvements. This is simulation process bridges the gap of abstraction between the software and simulation environment. This improves the overall simulation process and gives better quality results for the simulation.

VIII. FUTURE SCOPE

It is expected that the research in this area and continuous development will eventually result in a several of utility of the proposed design. These strategies will also greatly increase the effectiveness and efficiency of the previous designs.

REFERENCES

[1] Qi Guo, Tianshi Chen, Yunji Chen, and Franz Franchetti, "Accelerating Architectural Simulation Via Statistical Techniques: A Survey", IEEE Transaction on Computer-Aided design of Integrated Circuits and Systems, Vols. No.:35, Issue No:3, (March 2016), pp. 433-446.

[2] AyazAkram,LinaSawalha, "A Survey of Computer Architecture Simulation Techniques and Tools", IEEE Transaction on Computer-aided Design of Integrated Circuits and Systems, Vols. No.:7, (May 20, 2019), pp. 78120-78145.

[3] Mohammad Alian, Daehoon Kim, and Nam Sung Kim, "pd-gem5: Simulation Infrastructure for Parallel/Distributed Computer Systems", IEEE Computer Architecture Letter, Vol: 15, Issue No: 1, (January-June 2016), pp. 41-44.

[4] Andy D. Pimentel, CagkanErbas, Simon Polstra, "A Systematic Approach to Exploring Embedded System Architectures at Multiple Abstraction Levels", IEEE Transactions on Computers, Vol.: 55 Issue No.:02, (February 2006), pp. 0385-1104.

[5] Jung Ho Ahn and William J. Dally, "Data Parallel Address Architecture:,Computer Systems Laboratory Stanford University, Stanford, California 94305, USA, Vols. No.:5, (2006).

[6] Joshua J. Yi and David J. Lilija, "Simulation of computer Architectures: Simulators, Benchmarks, Methodologies and Recommendations", IEEE Transactions on Computers, Vol. 55, NO:3, (March 2006), pp. 268-280.

[7] Myoungsoo Jung, Jie Zhang, Ahmed Abulila, Miryeong Kwon, NargesShahidi, John Shalf, Nam Sung Kim, and MahmutKandemir, " SimpleSSD: Modeling Solid State Drives for Holistic System Simulation" , IEEE Computer Architecture Letters, Vol:17 Issue No:1, (January-June 2018), pp. 37-41.

[8] Davide Patti, Andrea Spadaccini, Maurizio Palesi, FabrizopFazzino, Vinsenso Catania, "Supporting Undergraduate Computer Architecture Students using a Visual MIPS64 CPU Simulator", IEEE Transaction on Education, Vol.55, No.3, (August 2012).

[9] Doug Burger, Joel James, C. Hoe, Derek Chiou, ResitSendag, "The Future of Architectural Simulation", IEEE Computer Society, (June 2011).

[10] Junmin Wu, Xiaodong Zhu, Tao Li, and Xiufeng Sui, "WBSP: A Novel Synchronization Mechanism for Architecture Parallel Simulation", IEEE Transaction on Computers, Vol:65 Issue No: 3, (March 2016), pp. 992-1005.