The Physical Design Implementation of a 32-Bit 5-Stage Pipelined MIPS Processor using SCL 180nm Technology

M. S. Gowtham, S. Syed Jamaesha, E. Veera Boopathy, M. Anandapriya

Abstract: The proposed work describes the physical design implementation of a 32-bit 5-stage pipelined MIPS processor. The various blocks of this processor include the data-path, control logic, data and program memories. Hazard detection and data forwarding units have been included for efficient implementation of the pipeline. Modified architecture is proposed that leads to significant area reduction by exploiting most of the functional units in a single clock cycle. Also, by increasing the instruction throughput, the overall performance is increased. The simulation of Verilog design for this project is done in Cadence NCLauch followed by synthesis using Cadence Genus. The RTL to GDSII implementation is carried out in Cadence Innovus using SCL 180nm Technology. Physical verification is performed in Cadence Virtuoso using Calibre tool.

Keywords: MIPS Processor, Pipeline, RTL to GDSII, SCL 180nm.

I. INTRODUCTION

The process of developing an IC by packing millions of transistors into a single chip, with smaller die size, is known as Very-Large Scale Integration or VLSI [13]. Nowadays due to the development of technology, a very complex system design is able to be converted into a very small IC and this IC designing process involves Electronic Design Automation (EDA) and their corresponding tools. EDA tools provide automatic design approaches for IC design which can minimize the required Turnaround Time (TAT) [15]. For instance, the design approaches are auto place and route, gate-level optimization, STA, CTS and etc.

The VLSI design flow can be used as the guide for the project. The design flow of the project is shown in Figure 1.1. RTL stands for Register Transfer Level. In this stage, the behavior and the function of the design are described in Hardware Description Language (HDL) format. For logic synthesis, it is a process of converting the RTL source codes into optimized gate level netlist [13]. For physical design, it is the process of converting the optimized gate-level netlist into geometrical representation of the design which is known as layout [13], GDSII file, a binary file, consists of cell references and the geometry parameters of the cells [16].

Fig. 1 Design Flow for the project [13]
The Physical Design Implementation of a 32-Bit 5-Stage Pipelined MIPS Processor using SCL 180nm Technology

II. SYSTEM DESCRIPTION-MIPS PROCESSOR

MIPS processor design is based on the RISC design principle that emphasizes on a load/store architecture. Due to the difference in time taken to access a register as compared to a memory location, it is much faster to perform operations in on-chip registers rather than in memory.

To eliminate the latency of memory operations, MIPS processor uses the load/store architecture where the access to memory is only through load and store instructions. The processor has many registers and operations are performed in data present in those registers [1].

A. Architecture

The architecture of a 32-bit 5-stage pipelined MIPS Processor consists of the major blocks such as Memory and Register Blocks, Datapath, Control Logic, Data Forwarding Unit and Hazard Detection Unit.

The instruction set of MIPS consists of three types namely Register, Immediate and Jump. Patterson and Hennessy [2] gives the complete list of instructions belonging to these types.

The instruction set of the MIPS Processor is as follows:
1. ADD rd, rs, rt: Reg[rd] = Reg[rs] + Reg[rt].
2. BNE rs, rt, imm16: if (Reg[rs] ≠ Reg[rt]) PC = PC + 4 + Sign_ext (Imm16) << 2 else PC = PC + 4.
3. J target: PC = {PC[31:28], target, 00).
4. JR rs: PC = Reg[rs].
5. LW rt, imm16(rs): Reg[rt] = Mem[Reg[rs] + Sign_ext (Imm16)].
6. SLT rd, rs, rt: If (Reg[rs] < Reg[rt]) Reg[rd] = 00000001 else Reg[rd] = 00000000.
7. SUB rd, rs, rt: Reg[rd] = Reg[rs] − Reg[rt].
8. SW rt, imm16(rs): Mem[Reg[rs] + Sign_ext (Imm16)] = Reg[rt].
9. XORI rt, rs, imm16: Reg[rt] = Reg[rs] XOR Zero_ext (Imm16).

The complete data path for the architecture of a 32-bit 5-stage pipelined MIPS Processor after adding Pipelined Registers, Forwarding Unit, Stall Control Unit, and Flush Control Unit to the single-cycle data path.

Pipelining is the process of accumulating instruction from the processor through a pipeline. It allows storing and executing instructions in an orderly process. It is also known as pipeline processing.

Pipelining is a technique where multiple instructions are overlapped during execution. Pipeline is divided into stages and these stages are connected with one another to form a pipe like structure.

The pipelined architecture of the processor consists of the stages such as Instruction Fetch, Instruction Decode, Execution, Memory and Write Back, where the different stages of the pipeline are separated by pipeline registers.

![Fig. 3 5-stage pipelined MIPS Datapath](image-url)
III. DESIGN METHODOLOGY

VLSI design flow is a series of steps that is used to translate and synthesize the specifications of a system into a chip [5]. The design flow consists of system specification, functional design and verification, logic design and verification, circuit design and verification, physical design and verification and finally the fabrication and testing. This VLSI design flow is known as top-down VLSI design flow [5].

A. Digital Design Using Verilog HDL and its Simulation

The first step is specifications of the design which includes parameters such as speed, power and area are specified. The next phase is functional design. In this phase, the design is mostly specified in the structural model and then translated into RTL [5]. After the system specification is done, the next phase is functional verification. In this phase, the design is checked in terms of functionality [5]. Further verification for RTL is done in order to check the syntax and functionality of the RTL [5]. If the functionality of the design does not meet the requirements, the designer will perform the functional design again in order to achieve the design requirements [5].

The proposed design is implemented using Verilog HDL. Verilog HDL of the corresponding design is compiled and elaborated in Cadence NCPlace followed by the simulation carried out in Cadence NCVerif. NCPlace is a graphical user interface that gives us a unified view of the files and libraries in your design. The tool provides us with an easy and consistent way to configure and launch the Cadence simulation tools.

B. Logic Synthesis

Logic synthesis is a process of translating the high-level description of the design in RTL into an optimized gate-level netlist [5]. Logic synthesis is a set of techniques that is used for VLSI design industry [6]. Recently, CAD approach is popular for electronic system design [5]. Of course, logic synthesis can be done by using CAD approach. The Verilog RTL file is given as the input file along with the design constraint files into the synthesis tool to get a synthesized netlist.

To begin with, the RTL description of the design like RTL codes in Verilog HDL format is translated into an un-optimized internal representation. Basically, the un-optimized internal representation is a type of data structure that is used to store the information of the design [5]. Next, the logic optimization is performed by the tools based on the Karnaugh map optimization approach [5]. Before technology mapping and optimization, the design constraints and technology library are applied to the tools. Design constraints are known as design goals such as operating condition, chip area, clock frequency, clock uncertainty, load and others [5]. Technology library is also known as standard cell library. Standard cell library is a type of database library which consists of logic gate and macro cells like flip-flop, adder and multiplexer [5]. After that, technology mapping and optimization are performed by the tools. Generally, technology mapping is a process of connecting the selected elements from the standard cell library in such a way as to achieve the predefined synthesis goals and functionality of the design [7]. On the other hand, optimization is a process of applying the logic transformations to the design based on the design constraints [7]. Eventually, the optimized gate-level netlist is generated by the tool and the tool used here is Genus Synthesis solution. Once the circuit design and verification are completed, the next step is physical design.

The popular synthesis tool is Genus Synthesis solution from Cadence. The ultimate goal of the Cadence Genus Synthesis Solution is very simple: deliver the best possible productivity during register-transfer-level (RTL) design and the highest quality of results (QoR) in final implementation. The Genus synthesis solution provides up to 5X faster synthesis turnaround times and scales linearly beyond 10M instances. In addition, a new physically aware context-generation capability reduces iterations between unit- and chip-level synthesis by 2X or more.

C. Physical Design

Physical design is a process of converting gate-level netlist into geometric representation which is known as layout [5]. The layout consists of standard cells, routings, clock trees, power nets and other information [5]. The synthesized output netlist is taken as an input file into the physical design step. The Physical design implementation also requires constraint file from synthesized output, technology library files and the IO files according to the corresponding technology. With these files and by including the following design steps, the layout design of RTL is created and finally GDSII is streamed out. The physical design flow starts from partitioning followed up by floor planning, placement, Clock Tree Synthesis, Static Timing Analysis, routing, chip finishing and ends up in physical verification.

1) Partitioning, Floor-planning, Placement:

A VLSI chip consists of thousands of logic gates. Due to the complexity of the design, the layout of the entire design cannot be handled at the same time. Generally, decomposing the design into a set of smaller sub circuits or blocks is known as partitioning [9]. After decomposition, each sub circuit or block can be designed simultaneously and independently. An interface specification that is used for connecting all the sub circuits or blocks is generated [9]. The interconnects that is used to connect the blocks or subsystems are required. The collection of those interconnects is known as netlist [9]. There are a few factors that need to be considered for partitioning. Namely, size of the blocks or sub circuits, the number of blocks or sub circuits and the number of interconnects between the sub circuits or blocks.

In the floor-planning stage, the chip is planned in such a way to accommodate all design components like standard cells, macros, memory units and its interconnects within a minimum area [5]. The area of the chip that is used for accommodate those design
components can be calculated [9].

The interconnect of the chip and the location of the standard cells need to be determined in order to minimized of the chip area [9]. In the floor-planning stage, core utilization ratio is set to the appropriate value in order to give margins for routability. The utilization ratio is the ratio between height of the core area and the length of the core area. The extra space requirements for power networks and clock tree routing need to be taken into consideration during the setup of core utilization ratio. Usually, little iteration is required for the process.

After the floor-planning stage, the area accommodated by each component of the chip and the number of terminals are defined [9]. The process of arranging the components of the chip on the specified area is known as placement [9]. The steps for placement include determining the position of each components on the chip, determining the minimum area arrangement for the components on the chip which can allows the completion of routing without any violations and lastly place those components on the chip according to defined positions [9]. After the placement, all of the components of the design like standard cells, macros and IPs are placed on to the defined core area accordingly within minimum area [9].

2) Clock Tree Synthesis, Static Timing Analysis:

The clock tree is synthesized by design synthesis close to an ideal clock for a fully synchronous design. During CTS, the tools will buffer the clock tree networks to a sufficient level of transition like fast rise and fall edges. Meantime, the tool tries to balance the clock branches so that clock skew is under controlled in order to prevent any timing violations. Clock skew control is not so important in clock tree buffers generation.

STA is performed on fully synchronous designs to validate the timing performance of the design. The speed and timing constraints must be provided to the tool as an input for STA. For instance, the design constraints can be clock period, input delay, output delay, load capacitance, wire capacitance, wire load model, clock uncertainty and clock transition. STA checks all possible timing violation paths under worst case operating condition. The STA that is provided by the tool is quite efficient and accurate.

3) Routing and Chip Finishing:

Routing is a process of determining the geometrical representation which is known as layout of all the nets of the chip [9]. This process is used for interconnecting the components on the chip based on the netlist [9]. The area which is not accommodated by the components is partitioned into rectangular regions known as channel and switchbox [9]. The channel and switchbox are utilized by the routers in order to complete the routing between the all components on the chip [9]. Further optimization can be done based on the constraints [9].

Chip finishing is the additional steps required for the chip tape out preparation. The antenna check can be done by the tool based on the antenna rules that provided by foundry. The antenna violations are caused by the longer length of the interconnects with the same metal layers [12]. This will cause damage to the chip due to the collected charges on the particular metal layers during fabrication. The fill cells are added into the remaining unused area in standard cell rows. The purpose of this process is to meet the continuity of the N-well and P-well and power buses. Additional metals and polys fill layers are added to the un-routed areas in order to establish the density requirements that set by the foundry. IO fill cells are added to the remaining empty space corresponding to the IO cells in order to complete the IO power ring of the chip which can provides ESD protection. Lastly, redundant vias are added wherever there are spaces to prevent the possibility of manufacturing issues. Some vias might become too narrow and might not be functioning properly.

D. Physical Verification

Physical verification is a process of verifying the IC design for manufacturability and electrical connectivity rules [10]. The three major steps in physical verification are DRC, LVS checks and parasitic extraction [14]. The physical verification can be done by automation [10]. The scripts required by the EDA tools to perform physical verification are generated by the algorithms when developing the layout and run set files of DRC and LVS checks [10]. These files are then executed in the terminal command line in order to automate the physical verification process [10]. The result also is generated by automation [10]. Design Rule Checker (DRC) is a process of analysing the design by a given set of design rules [5]. The inputs for Physical verification include the GDSII file, post layout netlist file, DRC run set files, LVS run set files, PEX extraction run set files. Once the LVS and DRC are cleared, the PEX extraction is done and the Spef netlist is generated as an PEX output. This PEX parasitic extraction is called as the Post layout RC extraction which is done after the verification steps.

E. Timing Analysis/Closure

The Timing Analysis/Closure is the process by which a logic design consisting of primitive elements such as combinatorial logic gates (and, or, not, nand, nor, etc.) and sequential logic gates (flip flops, latches, memories) is modified to meet its timing requirements. This timing closure is done after the post layout verification as the final timing analysis. This includes the input files as the spef netlist generated after the PEX extraction and also the constraint files. This will generate an optimized output file ignoring all the timing violations in layout. The output created after the timing closure will be a sdf file, which will be taken as input for post layout simulation.

F. Post Layout Simulation

The Post Layout Simulation is done to check whether the implemented design is working properly and eligible to send the Physical Design to the Fabrication process. The output file from the Timing closure along with the constraint file after physical design is taken as input, and the simulation is performed. The pre and post layout simulation results are compared and verified. The post layout simulation is performed in Cadence NC Launch.
G. Tools for Physical Design

The EDA tools provide the design automations such as auto placement, auto routing, physical verification and others. The TAT for physical design can be reduced by using EDA tools [12]. Cadence Innovus Implementation System can be used for hierarchical chip-level design planning known as floor-planning, placement, CTS, routing and physical verification. Innovus system takes an optimized gate-level netlist with design constraints file as the input and produces layout as an output.

H. Tools for Physical Verification

The Physical verification of the implemented design is done in a combined platform of Cadence Virtuoso and Mentor Graphics Calibre. As the full custom IC layout suite of the industry-leading Cadence Virtuoso platform, the Virtuoso Layout Suite supports custom analog, digital, and mixed-signal designs at the device, cell, block, and chip levels. Calibre nmDRC and Calibre nmLVS are the market share leaders in physical verification. Calibre also leads the market with innovative features such as incremental DRC, which ensures you can complete your design rule checking quickly and efficiently, and equation-based design rules, which let designers define continuous, three-dimensional functions that accurately and precisely reflect the complex physical interactions of today's nanometer designs.

I. Tools for Timing Closure

The timing closure of the design is carried out in the Synopsys' PrimeTime tool. Synopsys' PrimeTime static timing analysis tool provides a single, golden, trusted signoff solution for timing, signal integrity, power and variation-aware analysis. It delivers HSPICE accurate signoff analysis that helps pinpoint problems prior to tape out thereby reducing risk, ensuring design integrity, and lowering the cost of design.

### Table I: Results and Analysis

| Area     | Core-90% | Routing-10% |
|----------|----------|-------------|
| Power    | Consumption-445.38mW | Leakage-14.3uW |
| Slack    | 7612ps |
| Frequency| 40MHz |

IV. CONCLUSIONS

The Physical Design implementation of the 32-bit 5 stage pipelined MIPS processor in SCL 180nm Technology is done with the help of EDA tools and the Physical Verification of final layout design is done to evaluate whether the implemented design is free of errors. After the final verification, the GDSII file of the implemented design is generated. The Physical Verification of the implemented physical design is done and the DRC, LVS are cleared. The PEX extraction of the design is carried out and Timing closure is done, the output of Timing closure is taken as input file to post layout simulation and the post layout simulation result is compared and verified with the RTL simulation. Total cell utilization is about 90% and the rest 10% is utilized for the routing purpose. Total power consumed by the design is 445.38mW and leakage power is 14.3uW. Timing slack is obtained as 7612ps after the detailed synthesis. The implemented design works at a frequency of 40MHz.

The Final Layout View is shown in Fig.4 and the Detailed View of the Merged Design and Dummy Layout is Shown in the Fig.5. The shows the analyzed parameters of the output design.

REFERENCES

1. Kang, S.M., Leblebici, Y. and Kim, C. 2015. CMOS Digital Integrated Circuits: Analysis & Design, 4th ed. Singapore: McGraw-Hill.
2. Ryota, K., Kennouke, F., Kazuo, T., Hitoshi, K. and Shoji, H. 1985. An Integrated Modular and Standard Cell VLSI Design Approach. IEEE Transactions on Electron Devices. 32(2), pp.487-492.
3. Steven, M.R. 1994. Computer Aids for VLSI Design. [Online] Available at: http://www.rulabinsky.com/cavd/text/chapc.html.
4. MIPS32 architecture for Programmers. Vol 1: Introduction to MIPS architecture.
5. D. A. Patterson and J. L. Hennessy, Computer Organization and Design, The hardware/Software Interface. Morgan Kaufmann, 2005.

6. Das, D. 2010. VLSI Design. New Delhi: Oxford University Press.

7. Bernard, V. and Dominique, B. 1989. Logic Synthesis for VLSI. In: Walter, E.P. and Hans, R. ed. VLSI and Computer Peripherals: 3rd Annual European Computer Conference, 8/9/10/11/12 May 1989, Hamburg, New Jersey: IEEE, pp.5/10-5/14.

8. Eugenio, V. and Pablo, S. 1995. CAD Tools for Synthesis. In: ISIE '95 Proceedings of the IEEE International Symposium on Industrial Electronics, 10/11/12/13/14 July 1995, Athens. New Jersey: IEEE, pp.27-32.

9. Jeffery, B., Arun, S. and Naveed, S. 1993. Physical Tradeoffs for ASIC Technologies. In: AISC Conference and Exhibit, 1993: Proceedings., Sixth Annual IEEE International, 27 September – 1 October 1993, New York, New Jersey: IEEE, pp.70-78.

10. Rajesh, M.A., Soujanya, R., Kalpashree, M.A. and Soumya, S. 2014. Automated Physical Verification of I/O Pads in Full-Custom Environment. In: 2014 Fifth International Symposium on Electronic System Design, 15/16/17 December 2014, Surathkal. New Jersey: IEEE, pp.203-205.

11. Gautham P, Parthasarathy R, Karthi Balasubramanian, “Low-Power Pipelined MIPS Processor Design”, Proceedings of the 2009 12th International Symposium on Integrated Circuits, IEEE, 11105421.

12. E Veera Boopathy et al, “Design and analysis of high speed low area-power semi-custom standard library cells using 90nm MOCMOS technology”, Journal of Physics: Conference series, vol.1362, 2019.

AUTHORS PROFILE

Dr.M.S.Gowtham is working as an Assistant Professor at Karpagam Institute of Technology, Coimbatore since 2012. He has completed Ph.D research in the Department of Electronics and Communications Engineering at Karpagam Academy of Higher Education, Coimbatore in 2019. He completed his bachelor as well as master degree in Electronics and Communication Engineering branch from Anna University, Chennai, India in 2010 and 2012 respectively. His fundamental research interests are MANET’s, Network Security, Cross Layer Optimization, and Wireless Networks.

Dr.S.Syed Jameesha received his BE Degree in 2007 from the Department of Electronics and communication Engineering, Anna University, Chennai, India and his ME Degree in 2010 from the Department of Electronics and communication Engineering, Anna University, Coimbatore, India. And also he completed his PhD in the department of ECE at karupag Academy of Higher Education, Coimbatore. He has been a faculty member in the Department of Electronics and communication Engineering, Karpagam Institute of Technology, Coimbatore. His research interests are mainly, Adhoc Network and Optical fiber communication Networks. He has been served as a academician in the department of ECE more than one decade.

E.Veera Boopathy is working as an Assistant Professor at Karpagam Institute of Computer Conference, Coimbatore since 2019. He had completed PG in VLSI Design from Anna University, Chennai in 2015. He completed his bachelor degree in Electronics and Communication Engineering branch from Vinayaka Mission’s University, Salem in 2011. His research areas are Low Power VLSI circuits, Microwave Engineering and MEMS

M.Anandapriya completed her bachelor as well as master degree in Computer Science and Engineering from Anna University, Chennai, India in 2013 and 2015 respectively. From July 2015 to May 2016, she was working as an Assistant Professor at Kamaraj College of Engineering and Technology, Vinudhunagar. From June 2016, she is working as an Assistant Professor in the Department of Computer Science and Engineering at Sri Krishna College of Engineering and Technology, Coimbatore. Her main research interests are Big Data and Analytics, Data Mining and it’s applications, Software Architecture, Database Management System and so on.