a

Qualifying Phase Report - JoSDC’24

Team Information

|  |  |  |
| --- | --- | --- |
| Team Name | | |
|  | | |
| Team Members | | |
| # | Name | Email |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |

\*\* name the file of your team name before submission, and submit as PDF file.

eg: “Qualification Phase Report - TeamX.pdf”

# Summary

|  |  |
| --- | --- |
| Total number of bugs found |  |
| Total number of bugs fixed |  |

# Corrected errors.

Using the table below, please document the errors you have discovered during this phase. Ensure that necessary fields are completed. Additionally, remember to annotate the code with comments that include the error ID and a brief explanation to maintain synchronization between the code and the documentation.

Table Bug 1 information

|  |  |  |  |
| --- | --- | --- | --- |
| Bug Title: |  | | |
| Bug ID : |  | Bug Type |  |
| Reported by: |  | Open Date |  |
| Assigned to: |  | Close date |  |
| Description |  | | |
| Steps to reproduce |  | | |
| Expected Behavior |  | | |
| Actual Behavior |  | | |
| Solution implemented |  | | |

\*\* feel free to copy and paste the table as much as needed in the next pages, filling one table for each bug you find, also, you may add a screen shot showing more details or fixing the bug.

#supplement pages for Bug report.

# Full Analysis

In this section, you are required to fully analyze the processor and prepare to debug and trace its execution process. This involves summarizing the key signals and their behavior for all nine instructions supported by the processor. Understanding these signals is critical for troubleshooting and verifying that each instruction functions as expected.

The table below should be filled with an analysis of the key signals for each instruction, including their expected values during execution.

Table Control Unit Analysis Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Instruction | RegDst | Branch | MemReadEn | MemToReg |
| ADD |  |  |  |  |
| ADDI |  |  |  |  |
| SUB |  |  |  |  |
| AND |  |  |  |  |
| OR |  |  |  |  |
| SLT |  |  |  |  |
| LW |  |  |  |  |
| SW |  |  |  |  |
| BEQ |  |  |  |  |
|  |  |  |  |  |
| Instruction | ALUOp | MemWriteEn | ALUSrc | RegWriteEn |
| ADD |  |  |  |  |
| ADDI |  |  |  |  |
| SUB |  |  |  |  |
| AND |  |  |  |  |
| OR |  |  |  |  |
| SLT |  |  |  |  |
| LW |  |  |  |  |
| SW |  |  |  |  |
| BEQ |  |  |  |  |

# Functional Testing

In functional testing part, instructions should be hand-assembled, converted to hexadecimal (machine code), and loaded into the instruction memory. Then make sure to fill out the required fields below and add the corresponding screen shots to show your testing process.

Table Functional Testing Benchmark 1

|  |  |  |  |
| --- | --- | --- | --- |
| # | Instruction | Hexadecimal (Machine Code) | Result |
| eg | addi $5, $0, 0xff | 0x20050000 | $5 = 0x000000ff |
| 1 | addi $6, $0, 0x55 |  |  |
| 2 | sub $7, $5, $6 |  |  |
| 3 | sw $7, 0x0($0) |  |  |
| 4 | lw $8, 0x0($20) |  |  |
| 5 | beq $6, $7, fin |  |  |
| 6 | or $9, $6, $7 |  |  |
| 7 | and $8, $6, $7 |  |  |
| 8 | add $0, $6, $7 |  |  |
| 9 | fin : slt $10, $0, $5 |  |  |

Next page/s, are your space to add screen shot or multiple as needed, to show your results on simulation side, make sure to show correct and clear results, where waveforms and target values should be highlighted and well organized.

#supplement page for simulation results.

# Performance Results (Optional)

In this section, you are required to configure your corrected code to run in the Quartus tool. Using the Timing Analyzer (as discussed during the training phase), you will need to create a clock and specify timing constraints suitable for your processor. You are free to apply any constraints you deem appropriate.

After completing the analysis, report the performance metrics and required data in the table below. Additionally, provide the necessary commands from the Synopsys Design Constraints (SDC) file.

Table Performance Data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | Metric | Value | Description |
|  | | Clock Frequency |  | clock frequency you configured in the Quartus tool. |
|  | | Design Size (LE) |  | Size of design in terms of logic elements |
| model | Slow 85C | Fmax |  | Fmax : The highest frequency at which the processor can operate reliably.  Setup Time : The time required to set up signals before the clock edge.  Hold Time : The minimum time signals must remain stable after the clock edge. |
| Setup Slack |
| Hold Slack |
| Slow 0C | Fmax |  |
| Setup Slack |
| Hold Slack |
| Fast 0C | Fmax |  |
| Setup Slack |
| Hold Slack |

Provide the commands from your **SDC file** that define the clock and timing constraints.

#

# Free Space

This section is provided as a free space where you can add or present any additional information, notes, or observations that you think are important. You can also use this area to offer feedback on the phase, such as challenges faced, strategies you applied, or recommendations for future improvements.

Feel free to include:

* Additional **documentation** or **clarifications** on specific parts of the project.
* **Notes** on debugging strategies, testing approaches, or unique modifications you applied.
* **Feedback** on the tools used, process, or guidance provided during the phase.

This space is for any information you believe would add value to the overall documentation or help improve future phases.

#supplement page for free space section.