### **1st Page: Introduction**

Name: Sanchit Gupta
Roll No.: 23M1114
Branch: Electronic System (EE5)
Project Title: RTL to GDS Flow of MIPS Processor using Opensource Tools

#### **1. Introduction**

The MIPS (Microprocessor without Interlocked Pipeline Stages) architecture is a popular instruction set architecture (ISA) known for its simplicity and efficiency. This project focuses on the design and implementation of a Single-Cycle MIPS processor, following the RTL to GDS flow using open-source Electronic Design Automation (EDA) tools. The processor supports a variety of arithmetic, logical, and control operations, providing a robust foundation for embedded systems and educational purposes.

The project leverages open-source tools such as Yosys, OpenSTA, and OpenROAD, coupled with the FreePDK45 library, to translate the high-level RTL design into a physical layout ready for fabrication.

#### **2. Design Methodology**

##### **2.1 RTL Design**

The MIPS processor was designed in Verilog, following a Single-Cycle architecture. This architecture completes each instruction in a single clock cycle, allowing for straightforward control logic and efficient execution of instructions.

Key Features:
- Single-Cycle execution.
- Supports arithmetic operations (e.g., ADD, SUB).
- Logical operations (e.g., AND, OR).
- Control operations (e.g., JUMP, BRANCH).

The Instruction Set Architecture of this MIPS processor includes the following instructions.

1.ADD

2.SUB

3.AND

4.OR

5.SLT

6.JR

7.LW

8.SW

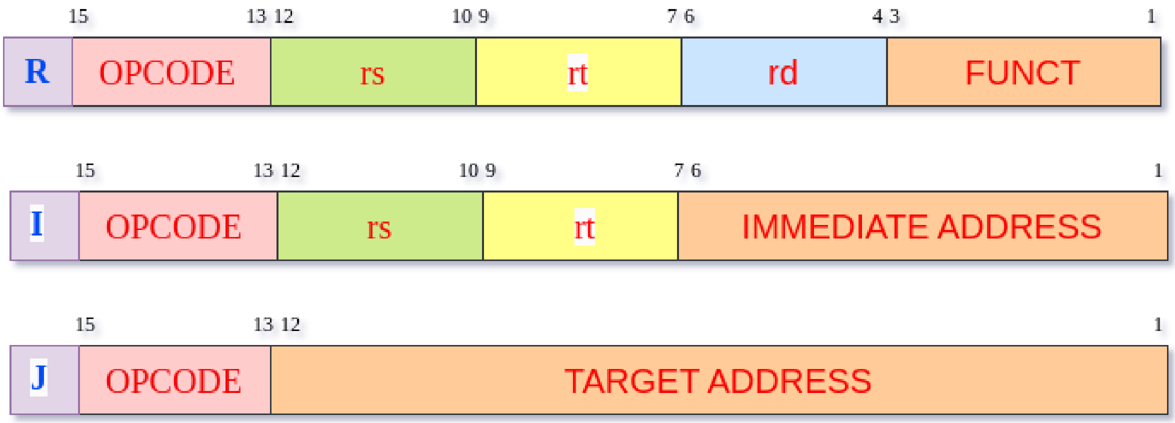
9.BEQ

10.ADDI

11.SLTI

12.J

13.JAL



Design Implementation:
The processor design includes modules such as the ALU (Arithmetic Logic Unit), Control Unit, and Register File, all implemented in Verilog. The design was thoroughly verified using testbenches to ensure correct functionality across a range of instructions.

##### **2.2 Synthesis and Timing Analysis**

The RTL code was synthesized using the Yosys tool, which generated a gate-level netlist compatible with the FreePDK45 library. Yosys, an open-source synthesis tool, efficiently maps the high-level Verilog code to standard cell libraries, creating an optimized gate-level representation.

Gate-Level Netlist Generation:
- Yosys synthesizes the Verilog code.
- The FreePDK45 library provides standard cells for synthesis.
- The synthesized netlist is then subjected to timing analysis.

Timing constraints were verified using the OpenSTA tool, ensuring that the design meets the required setup and hold times across all paths. Power consumption analysis was also conducted to evaluate the processor's efficiency in terms of energy usage.

##### **2.3 Physical Design**

The physical design was executed using the OpenROAD tool, which covers various stages from floor planning to clock tree synthesis (CTS) and routing.

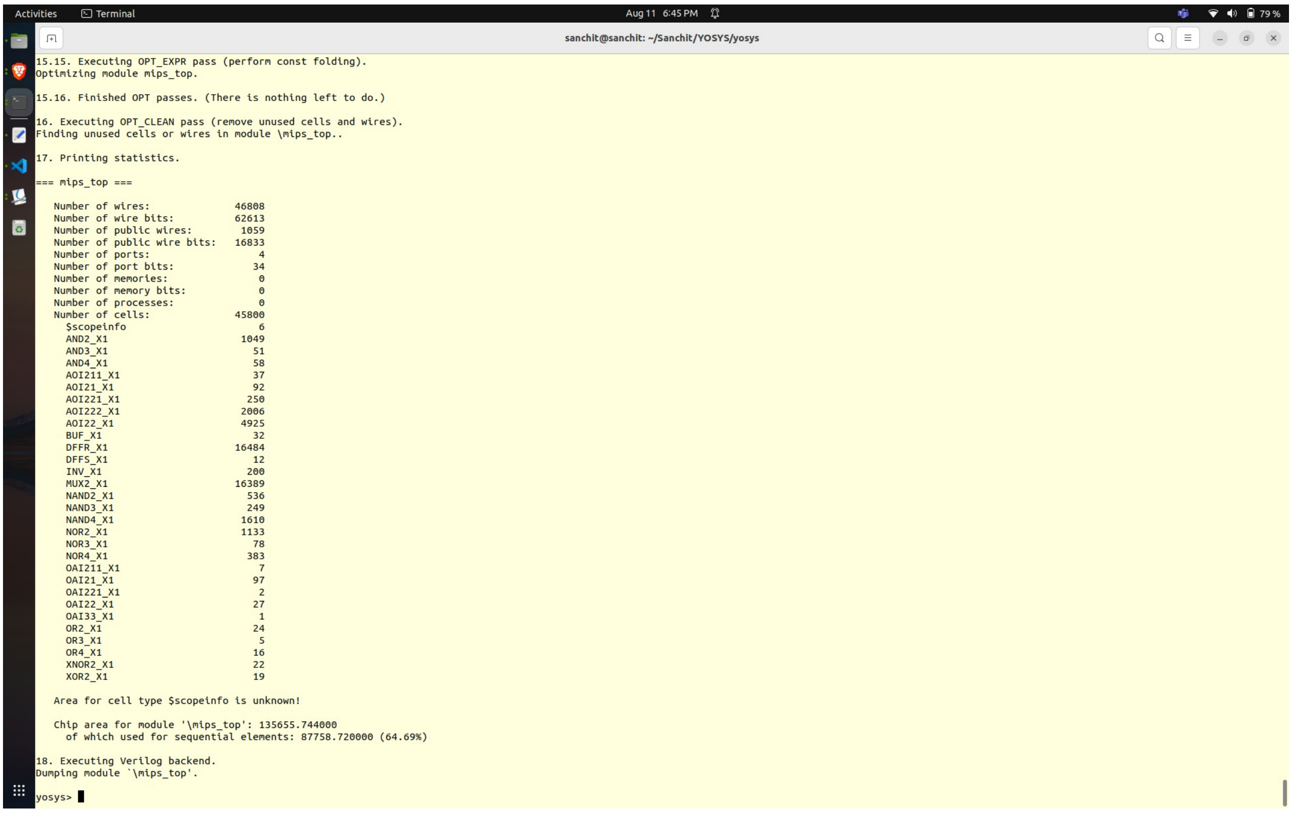
Floor Planning and Power Planning:
- Defined the chip layout and allocated space for the processor modules.
- Ensured optimal power distribution across the design to prevent IR drops and ground bounces.

Placement and Routing:
- Placed the standard cells on the chip.
- Routed the interconnections between cells, ensuring minimal delay and congestion.

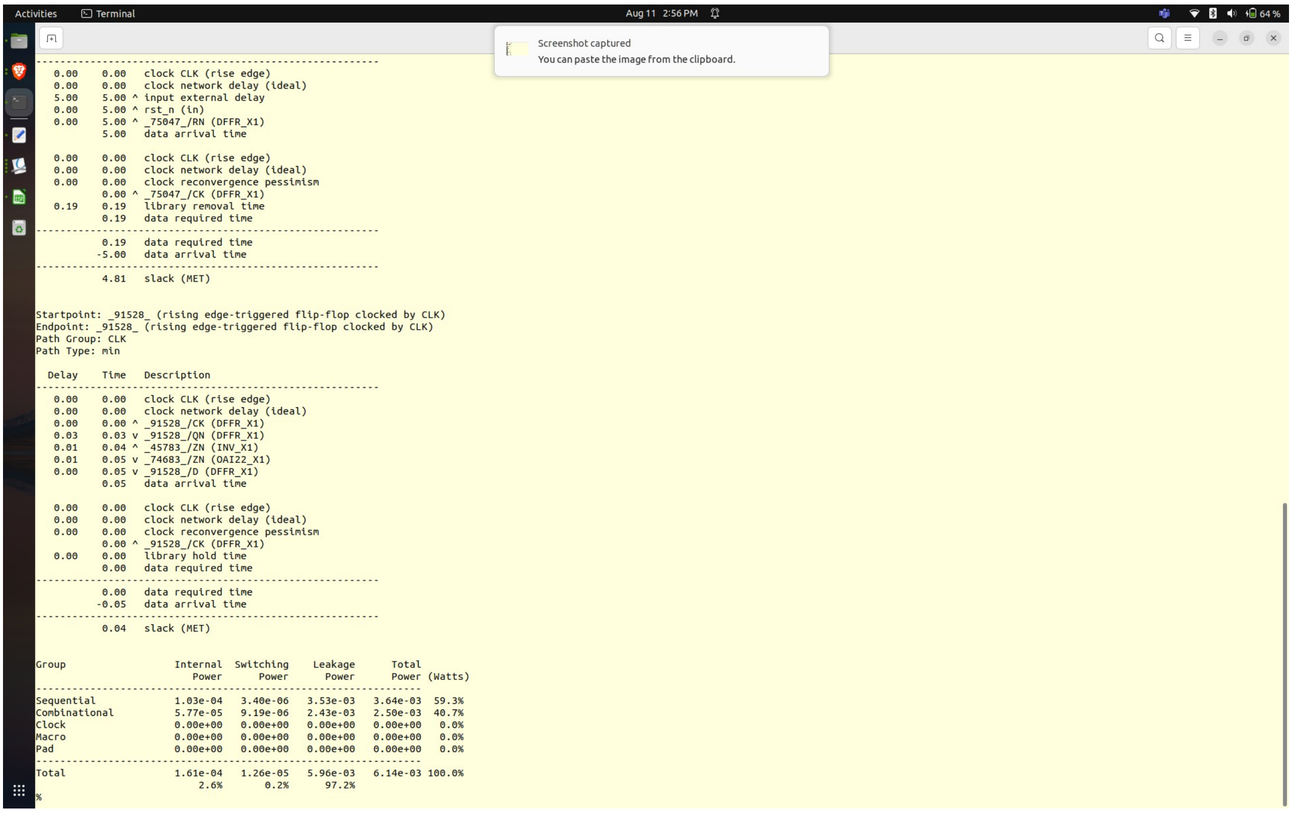
Clock Tree Synthesis (CTS):
- Implemented a balanced clock tree to distribute the clock signal uniformly across the design.

#### **3. Results and Validation**

The final design was validated through rigorous timing analysis and power estimation. The timing analysis confirmed that all critical paths met the required constraints, ensuring reliable operation at the target clock frequency. Power analysis indicated that the design was energy-efficient, making it suitable for various embedded applications.

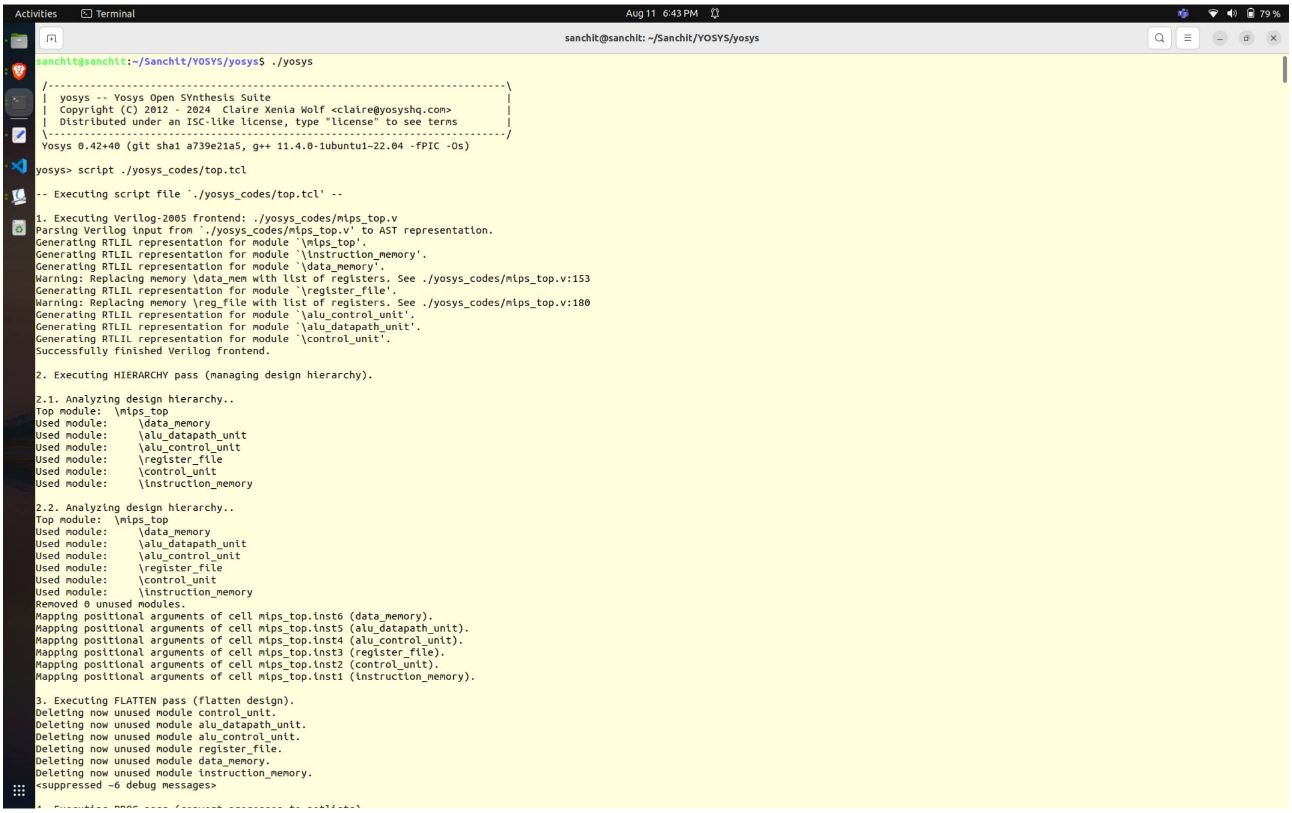


```
15.15. Executing OPT_EXPR pass (perform const folding).
Optimizing module mips_top.

15.16. Finished OPT passes. (There is nothing left to do.)

16. Executing OPT_CLEAN pass (remove unused cells and wires).
Finding unused cells or wires in module \mips_top..

17. Printing statistics.

=== mips_top ===

   Number of wires:              46808
   Number of wire bits:          62613
   Number of public wires:        1059
   Number of public wire bits:   16833
   Number of ports:                  4
   Number of port bits:             34
   Number of memories:               0
   Number of memory bits:            0
   Number of processes:              0
   Number of cells:              45800
     $scopeinfo                      6
     AND2_X1                      1049
     AND3_X1                        51
     AND4_X1                        58
     AOI211_X1                      37
     AOI21_X1                       92
     AOI221_X1                     250
     AOI222_X1                    2006
     AOI22_X1                     4925
     BUF_X1                         32
     DFFR_X1                     16484
     DFFS_X1                        12
     INV_X1                        200
     MUX2_X1                     16389
     NAND2_X1                      536
     NAND3_X1                      249
     NAND4_X1                     1610
     NOR2_X1                      1133
     NOR3_X1                        78
     NOR4_X1                       383
     OAI211_X1                       7
     OAI21_X1                       97
     OAI221_X1                       2
     OAI22_X1                       27
     OAI33_X1                        1
     OR2_X1                         24
     OR3_X1                          5
     OR4_X1                         16
     XNOR2_X1                       22
     XOR2_X1                        19

   Area for cell type $scopeinfo is unknown!

   Chip area for module '\mips_top': 135655.744000
     of which used for sequential elements: 87758.720000 (64.69%)

18. Executing Verilog backend.
Dumping module `\mips_top'.

yosys> 
```

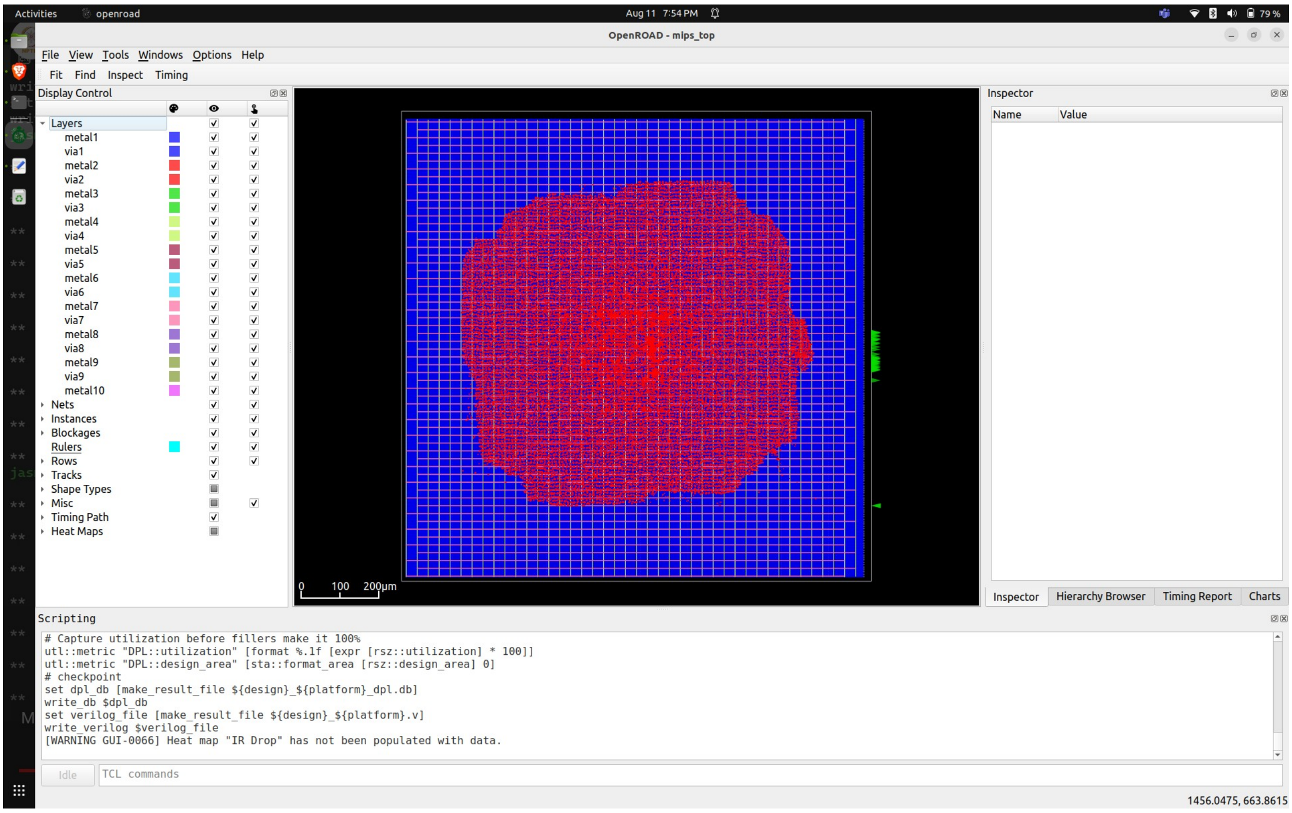


Screenshot captured
You can paste the image from the clipboard.

```
------------------------------------------------------------
   0.00    0.00   clock CLK (rise edge)
   0.00    0.00   clock network delay (ideal)
   5.00    5.00 ^ input external delay
   0.00    5.00 ^ rst_n (in)
   0.00    5.00 ^ _75047_/RN (DFFR_X1)
           5.00   data arrival time

   0.00    0.00   clock CLK (rise edge)
   0.00    0.00   clock network delay (ideal)
   0.00    0.00   clock reconvergence pessimism
           0.00 ^ _75047_/CK (DFFR_X1)
   0.19    0.19   library removal time
           0.19   data required time
------------------------------------------------------------
           0.19   data required time
          -5.00   data arrival time
------------------------------------------------------------
           4.81   slack (MET)


Startpoint: _91528_ (rising edge-triggered flip-flop clocked by CLK)
Endpoint: _91528_ (rising edge-triggered flip-flop clocked by CLK)
Path Group: CLK
Path Type: min

  Delay    Time   Description
------------------------------------------------------------
   0.00    0.00   clock CLK (rise edge)
   0.00    0.00   clock network delay (ideal)
   0.00    0.00 ^ _91528_/CK (DFFR_X1)
   0.03    0.03 v _91528_/QN (DFFR_X1)
   0.01    0.04 ^ _45783_/ZN (INV_X1)
   0.01    0.05 v _74683_/ZN (OAI22_X1)
   0.00    0.05 v _91528_/D (DFFR_X1)
           0.05   data arrival time

   0.00    0.00   clock CLK (rise edge)
   0.00    0.00   clock network delay (ideal)
   0.00    0.00   clock reconvergence pessimism
           0.00 ^ _91528_/CK (DFFR_X1)
   0.00    0.00   library hold time
           0.00   data required time
------------------------------------------------------------
           0.00   data required time
          -0.05   data arrival time
------------------------------------------------------------
           0.04   slack (MET)


Group                  Internal  Switching    Leakage      Total
                          Power      Power      Power      Power (Watts)
----------------------------------------------------------------
Sequential             1.03e-04   3.40e-06   3.53e-03   3.64e-03  59.3%
Combinational          5.77e-05   9.19e-06   2.43e-03   2.50e-03  40.7%
Clock                  0.00e+00   0.00e+00   0.00e+00   0.00e+00   0.0%
Macro                  0.00e+00   0.00e+00   0.00e+00   0.00e+00   0.0%
Pad                    0.00e+00   0.00e+00   0.00e+00   0.00e+00   0.0%
----------------------------------------------------------------
Total                  1.61e-04   1.26e-05   5.96e-03   6.14e-03 100.0%
                           2.6%       0.2%      97.2%
%
```



```
sanchit@sanchit:~/Sanchit/YOSYS/yosys$ ./yosys

 /----------------------------------------------------------------------------\
 |  yosys -- Yosys Open SYnthesis Suite                                       |
 |  Copyright (C) 2012 - 2024  Claire Xenia Wolf <claire@yosyshq.com>         |
 |  Distributed under an ISC-like license, type "license" to see terms        |
 \----------------------------------------------------------------------------/
 Yosys 0.42+40 (git sha1 a739e21a5, g++ 11.4.0-1ubuntu1~22.04 -fPIC -Os)

yosys> script ./yosys_codes/top.tcl

-- Executing script file `./yosys_codes/top.tcl' --

1. Executing Verilog-2005 frontend: ./yosys_codes/mips_top.v
Parsing Verilog input from `./yosys_codes/mips_top.v' to AST representation.
Generating RTLIL representation for module `\mips_top'.
Generating RTLIL representation for module `\instruction_memory'.
Generating RTLIL representation for module `\data_memory'.
Warning: Replacing memory \data_mem with list of registers. See ./yosys_codes/mips_top.v:153
Generating RTLIL representation for module `\register_file'.
Warning: Replacing memory \reg_file with list of registers. See ./yosys_codes/mips_top.v:180
Generating RTLIL representation for module `\alu_control_unit'.
Generating RTLIL representation for module `\alu_datapath_unit'.
Generating RTLIL representation for module `\control_unit'.
Successfully finished Verilog frontend.

2. Executing HIERARCHY pass (managing design hierarchy).

2.1. Analyzing design hierarchy..
Top module:  \mips_top
Used module:     \data_memory
Used module:     \alu_datapath_unit
Used module:     \alu_control_unit
Used module:     \register_file
Used module:     \control_unit
Used module:     \instruction_memory

2.2. Analyzing design hierarchy..
Top module:  \mips_top
Used module:     \data_memory
Used module:     \alu_datapath_unit
Used module:     \alu_control_unit
Used module:     \register_file
Used module:     \control_unit
Used module:     \instruction_memory
Removed 0 unused modules.
Mapping positional arguments of cell mips_top.inst6 (data_memory).
Mapping positional arguments of cell mips_top.inst5 (alu_datapath_unit).
Mapping positional arguments of cell mips_top.inst4 (alu_control_unit).
Mapping positional arguments of cell mips_top.inst3 (register_file).
Mapping positional arguments of cell mips_top.inst2 (control_unit).
Mapping positional arguments of cell mips_top.inst1 (instruction_memory).

3. Executing FLATTEN pass (flatten design).
Deleting now unused module control_unit.
Deleting now unused module alu_datapath_unit.
Deleting now unused module alu_control_unit.
Deleting now unused module register_file.
Deleting now unused module data_memory.
Deleting now unused module instruction_memory.
<suppressed ~6 debug messages>

4. Executing PROC pass (convert processes to netlists).
```



```
# Capture utilization before fillers make it 100%
utl::metric "DPL::utilization" [format %.1f [expr [rsz::utilization] * 100]]
utl::metric "DPL::design_area" [sta::format_area [rsz::design_area] 0]
# checkpoint
set dpl_db [make_result_file ${design}_${platform}_dpl.db]
write_db $dpl_db
set verilog_file [make_result_file ${design}_${platform}.v]
write_verilog $verilog_file
[WARNING GUI-0066] Heat map "IR Drop" has not been populated with data.
```

#### **4. Conclusion**

The project successfully demonstrated the complete RTL to GDS flow of a Single-Cycle MIPS processor using open-source tools. The processor design, synthesized using Yosys and physically implemented with OpenROAD, was validated to meet both timing and power constraints. This project highlights the potential of open-source EDA tools in enabling high-quality ASIC design.