University of Oulu

Digital Techniques 2

Project Progress Report 2025: mycpu

<Author> Arafat Miah
<Student Number> 2512200

Project done in cooperation with:
Only the week 4 is done by group studing with ASHRAF UDDIN SHEIKH <2508787> and <Mohammad Ammar Abdur Rahman Khan>
<Student Number> 2509732

Table of Contents

Revision History

Version	Date	Author	Comment
1.0	1.4.2024	J.L.	Template created.

WEEK1: PROJECT CREATION

LEARNING GOALS
1. Learn the principle of structural modeling. 2. Learn to define and verify a structural model in SystemVerilog and EDA tools.

1.1. Project Parameters

The parameters are decoded from the three last digits L, M and N of your student number.

Student Number = 2512200		
Parameter	**Digit**	**Value**
CLK_PERIOD	N =0	Present clock period in ns here. 4ns+N/20= 4ns
USR instruction	M =0	Present special instruction description here. I read and understood what does M means but the description was long which is shows in 5.6.3 and due to the long description I did not add it here
General randomizer	L =2	Selected case-by-case.

1.1. Module Instantiation Creation

Present and describe instantiation of one module X selected based on parameter L as follows 0,8: X=PC, 1,9: IR, 2: MUXM, 3: RB, 4: MUXD, 5: MUXB, 6: CU, 7:FU
`Insert code fragment showing the instantiation from mycpu.sv.` muxm MUXM (.sel_in(mx[0]), .d0_in(abus), .d1_in(pca), .m_out(a_out));

`Insert code for the respective module (header part).`
`module muxm`
`  (input logic sel_in,`
`   input logic [15:0] d0_in,`

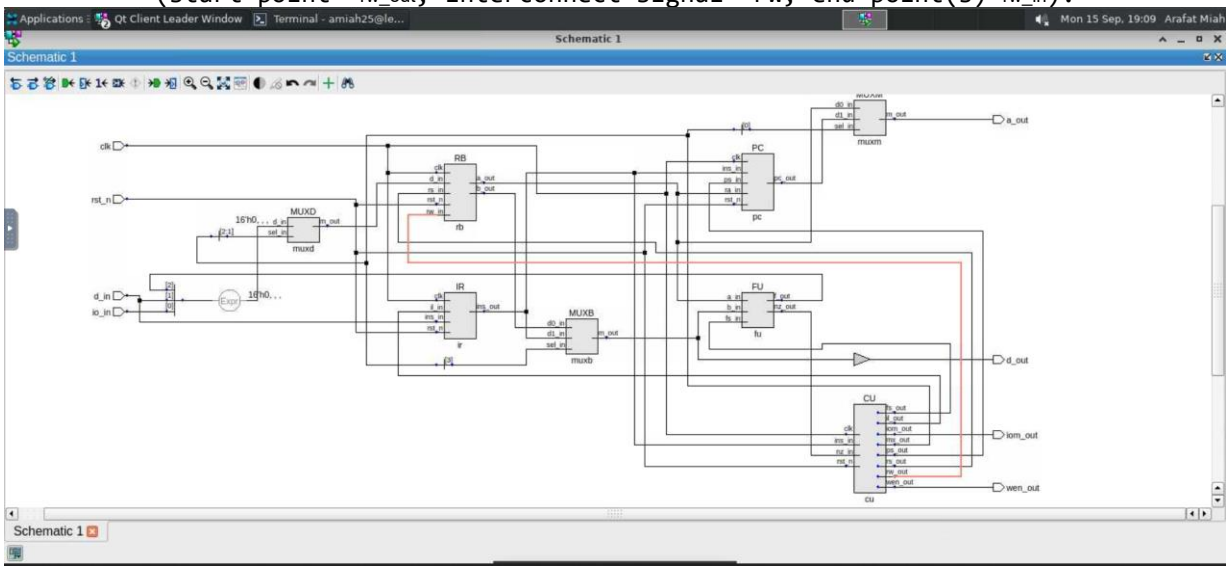
```
   input logic [15:0] d1_in,
   output logic [15:0] m_out);


endmodule
```

Describe the parts of the instantiation and their purpose.
The word instantiation here defines the inputs, outputs, and the internal logic we are going to use in a module. If we take the above fragment, then the first muxm defines the module name, MUXM is the unique identifier for the module muxm. Now, sel_in port is connected to mx[0], d0_in port is connected with abus, d1_in port is connected with pca, and m_out port is connected with a_out.

1.2. Interconnect Creation

Present and describe the parts that define the connection Y in module mycpu selected based on parameter L as follows:
0: Y=ps, 1: ins, 2: rw, 3: il, 4: fs, 5: nz, 6: abus, 7:dbus, 8: mx, 9: bbus

Insert code fragments showing the parts that define the connection from mycpu.sv (start point= rw_out, interconnect signal= rw, end point(s)=rw_in).

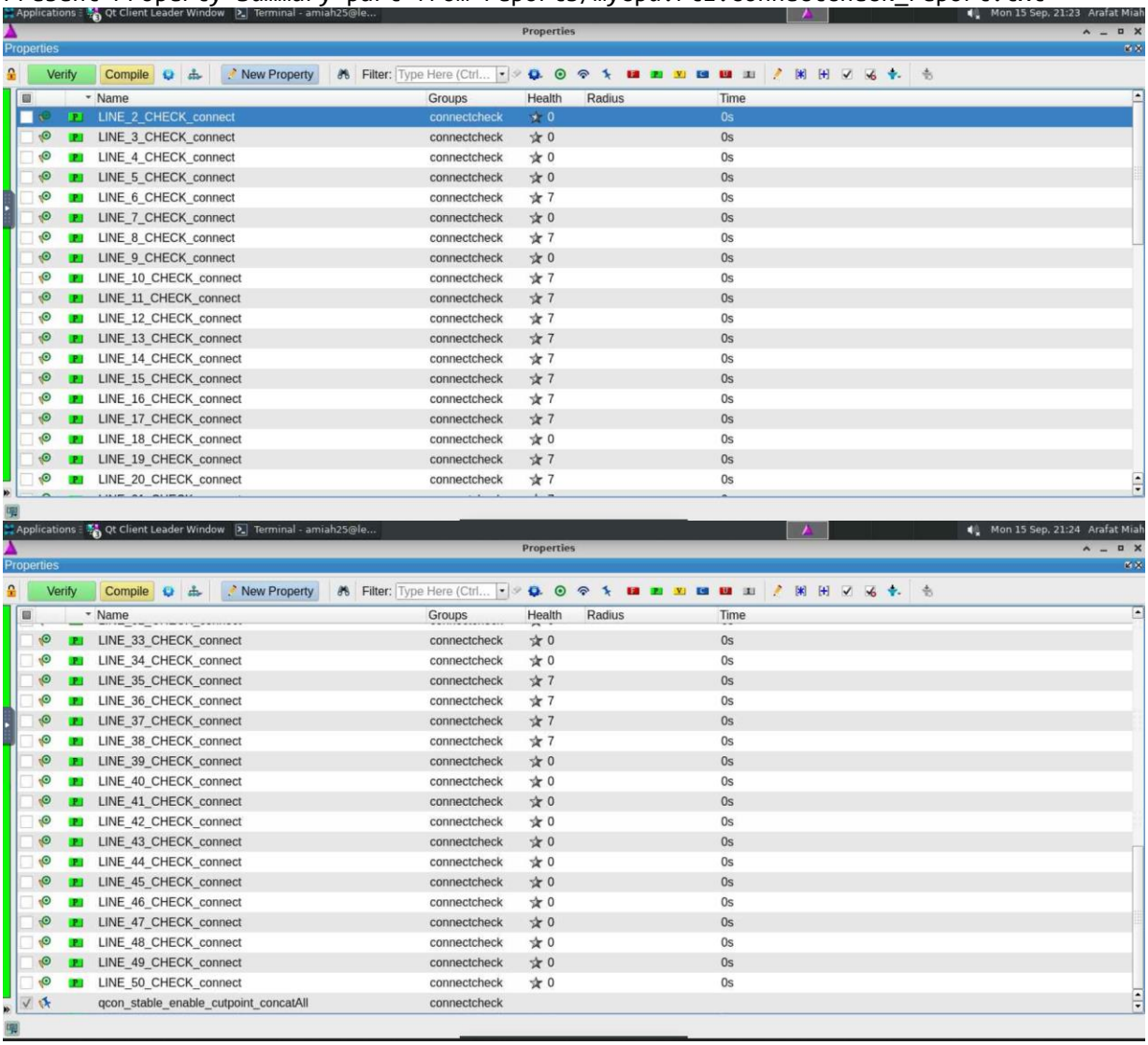


Describe the parts of the interconnect definition and their purpose.
rw wire is a connection wire that makes a connection between the CU and RB. Where, RB is a 16 x 16-bit register bank, consisting of an 8x16 user register bank and an 8x16 hidden register bank. One register can be selected for writing, and two registers for reading, and the CU is the control unit that contains the control state machine and instruction decoder. (N.B.: The definitions have been taken from mycpu_specsheets_2025 file) .
Purpose: The purpose of this connection is to flow the data from rw_out of CU to the rw_in of RB(rb).

1.3. Connection Checks

Present connect check summary report and comment the results.

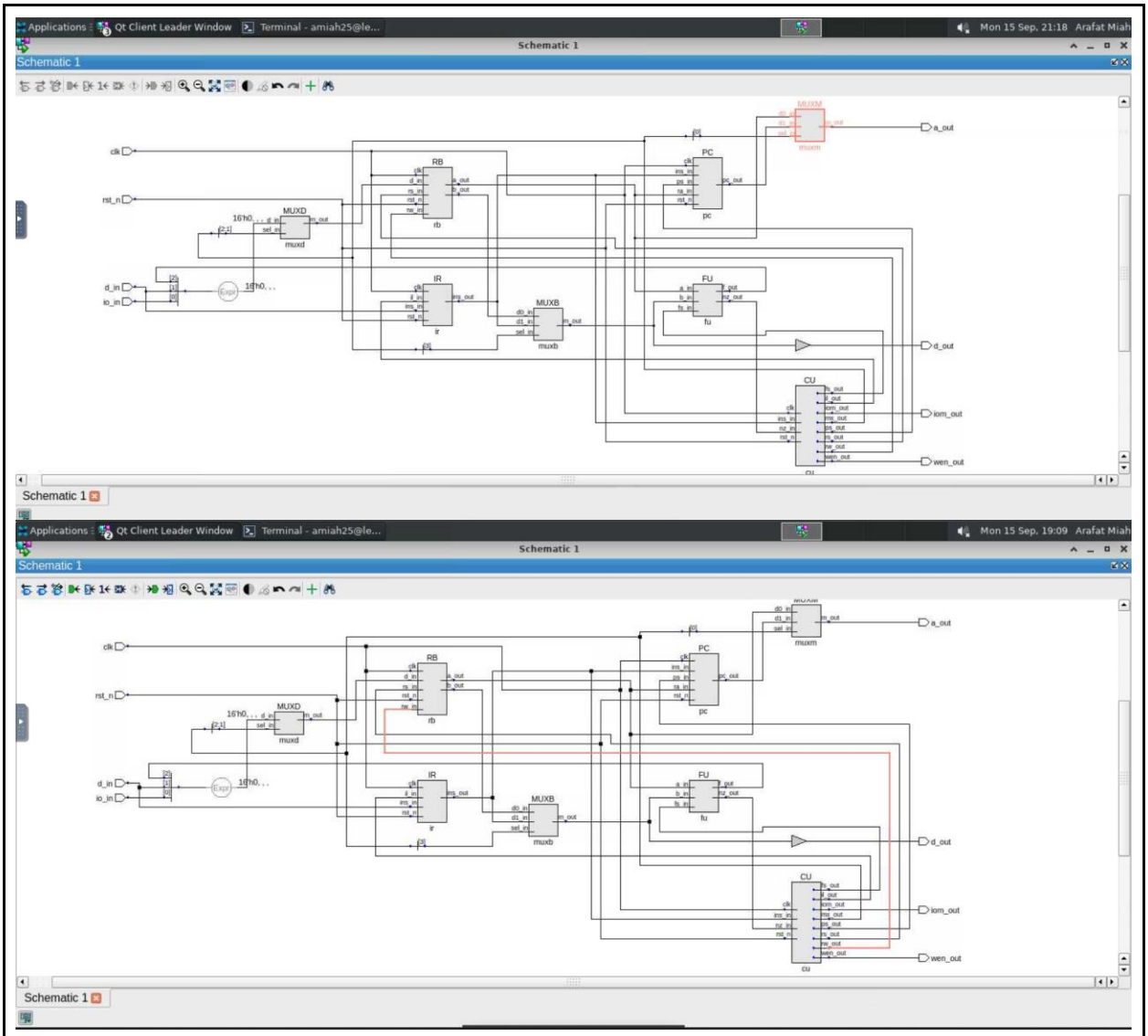
Present Property Summary part from reports/mycpu.rtl.connectcheck_report.txt



Comment the results: I tried my best to find the .txt file, however I could not find it. So I just take the screenshoots of the report and upload it.

Present Connect Check schematic with module X and interconnect Y highlighted.

Insert mycpu.png



SELF ASSESSMENT

Write a summary of week's work and results, including

1. Assessment of results
    a. Does the code compile and elaborate without errors and warnings?
    Ans: In Lab 2, I had to copy a portion of code that contained an error, which I corrected, and then showed the output wave. Similarly, in the project, I also encountered an error that I fixed, after which the overall code ran successfully.
    b. Are instantiations and connections correct?
    Ans: Yes, whichever instantiations and connections I had were corrected.
2. Approximate time spent on project this week

3. Ans: As I received my student ID a little late, I had to enroll in this course at the end of the second week, which caused me to miss some classes. Therefore, I had to complete the work by myself, which took three days. Now, at 8:43 pm on September 15, I am still writing this project summary. However, I really enjoyed it after successfully running all the codes.
4. Open issues you need help with
   Ans: I think there should be some pre-made videos on how to get started in Linux, how to successfully run basic code, how to find errors, and how to save the required files. I am still struggling with this, but I hope that I will soon find a way to save my required files.

WEEK2: COMBINATIONAL LOGIC DESIGN (muxm, muxb, muxd)

LEARNING GOALS
1. Learn the principle of modeling combinational logic. 2. Learn the check code functionality. 3. Learn to check code quality and synthesizability. 4. Learn to examine synthesis results. 5. Learn to design basic test programs.

2.1. muxm Code Design

2.1.1. muxm RTL Simulation Results

Test Program Output and Comment the Results

Copy-paste the output from file reports/muxm.rtl.simulation_log.txt:

```
--------------------------------------------------------------------------------
SIMULATION START 09/22/2025:16:37:46 amiah25
# ** Info: T1
#    Time: 16 ns  Scope: muxm_tb.TEST.test_program File: input/muxm_test.sv Line:
32
# ** Info: T2
#    Time: 28 ns  Scope: muxm_tb.TEST.test_program File: input/muxm_test.sv Line:
49
# ** Info: T3
#    Time: 36 ns  Scope: muxm_tb.TEST.test_program File: input/muxm_test.sv Line:
66
# ** Note: $finish    : input/muxm_test.sv(82)
#    Time: 196 ns  Iteration: 1  Instance: /muxm_tb/TEST
# 1
# Break in NamedBeginStat test_program at input/muxm_test.sv line 82
# 0

# SIMULATION END (0 assertions failed)
# --------------------------------------------------------------------------------
# 0 ps
# 205800 ps
# End time: 16:40:45 on Sep 22,2025, Elapsed time: 0:03:03
# Errors: 0, Warnings: 0
```

Comments (did the tests pass or fail, and if so why)

The code runs with zero error indicating the tests pass.

Statement Coverage

```
Insert Statement coverage value from file reports/muxm.rtl.simulation_coverage.txt
Enabled Coverage              Bins      Hits    Misses  Coverage
    ----------------             ----      ----    ------  --------
    Branches                        2         2         0   100.00%
    Statements                      3         3         0   100.00%
    Toggles                        98        98         0   100.00%
```

Comments (did the tests activate all lines of RTL code)
As we can see from the above statement, there are 0 misses, so the tests activate all lines.

2.1.2. muxm Lint Check Results

Present results of lint checks after you fixed all problems you could. Type = W(arning), E(Error), ID = check name (e.g. CASE_DEFAULT), Comment = Why the issue can be ignored, or if the problem remains, explain its meaning. See report **reports/muxm.rtl.qlint_report.tx**t Section 2 Check Details.

Lint Warnings and Errors: I did not get any warnings and errors.		
Type	ID	Comment

2.1.3. muxm AutoCheck Results

See report **reports/muxm.rtl.qautocheck_report.txt** Section AutoCheck Details.

AutoCheck Warnings and Errors: I did not find any warnings and errors		
Type	ID	Comment

2.1.4. muxm Synthesis Results Check

See report **reports/muxm.rtl.synthesis_log.txt**. In the Design Vision GUI, use Help > Man Pages to find out explanations for warning and error messages based on their ID (e.g. PWR-80).

Synthesis Warnings and Errors: the final error and warning is: 1 error, 2 warnings, 21 informationals from which 1 error is 'Error: unknown command 'Rexit' (CMD-005)' which I can ignore. and the warning that is given can also be ignored. warnings:		
Type	ID	Comment
Warning:	**(PWR-80)**	**There is no defined clock in the design.**

2.1.5. muxm Synthesis Reference Cell Check

See report **reports/muxm.gatelevel.references.txt**.

References Report Analysis

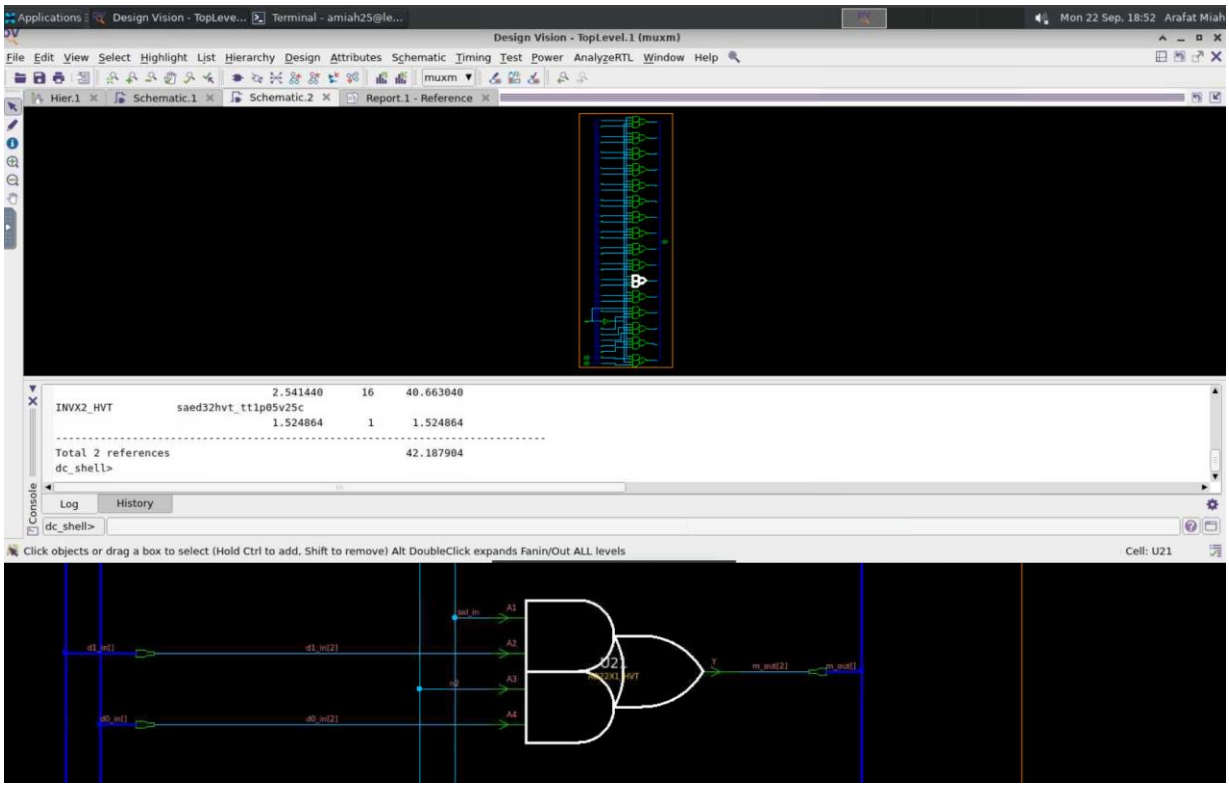
```
Reference           Library       Unit Area   Count    Total Area   Attributes
--------------------------------------------------------------------------------
AO22X1_HVT          saed32hvt_tt1p05v25c
                                  2.541440      16     40.663040
INVX2_HVT           saed32hvt_tt1p05v25c
                                  1.524864       1      1.524864
```

Comments (are the components ok for a combinational design) .yeah, It is ok

2.1.6. muxm Synthesis Schematic Analysis

Gate Level Schematic

```
Include muxm_detail.png
```



Explain how the value of m_out[L] is computed in the gate-level design. L is your randomizer design parameter.
For the 'm_out[2]' output, the logic is realized with a two-level gate structure. First, the design uses two AND gates. One AND gate receives `sel_in` and `d1_in[2]` as inputs, while the other receives `sel_in` (inverted, labeled as `n2`) and `d0_in[2]`. The outputs of these two AND gates are then combined by an XOR gate to produce the final `m_out[2]` signal.

2.2. muxb Design

2.2.1. muxb RTL Code Creation Results

Describe Code Implementation Principles and Check the Results
Give a short description on how the code was implemented (you can include the relevant parts). Comment on Lint, AutoCheck and synthesis results. The code implements a multiplexer that, when `sel_in` is 0, passes the entire `d0_in` value to the `m_out` output. When `sel_in` is 1, it assigns only the three least significant bits of `d1_in` to `m_out` and sets all the other bits of the `m_out` output to 0.

2.2.2. muxb Test Program Implementation and Simulation Results

Describe Code Implementation Principles and Verification Results

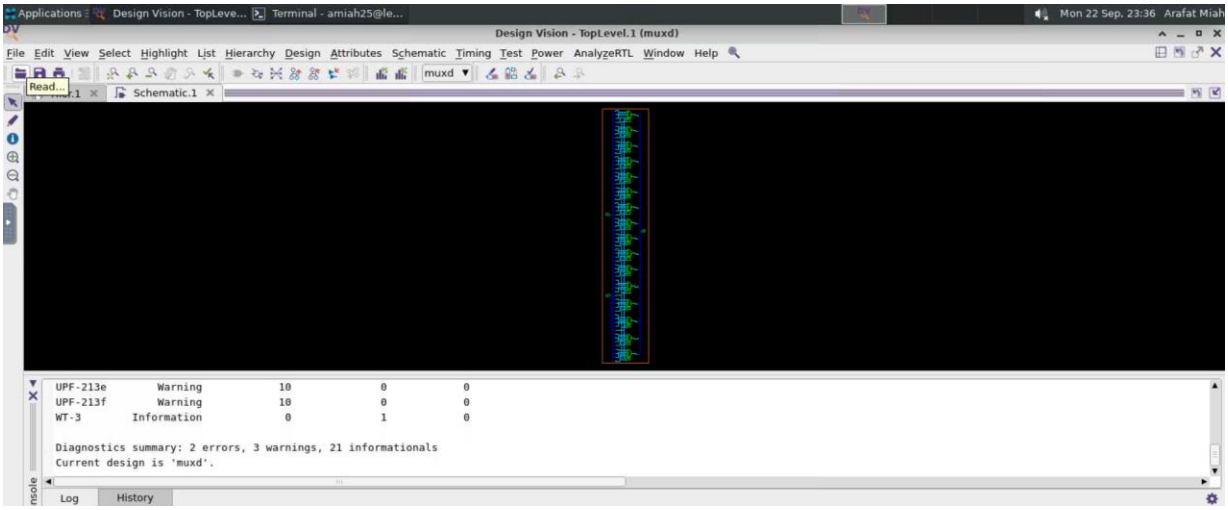
Give a short description on how the test bench and test program code was implemented (you can include the relevant parts). Comment on simulation results and code coverage.

```
`include "mycpu.svh"

import mycpu_pkg::*;

module muxb
  (
    input  logic       sel_in,
    input  logic [15:0] d0_in,
    input  logic [15:0] d1_in,
    output logic [15:0] m_out
  );


  always_comb
    begin
      if (sel_in == 1'b0)
        m_out = d0_in;
      else
        m_out = {13'b0, d1_in[2:0]};
    end

endmodule
```

I edited the code correctfully but there was some warning but no error evertime I was running the test bench,,….

```
# ** Warning: input/muxb_tb.sv(7): (vlog-2275) Existing module 'muxb' at input/muxb_test.sv(7) will be overwritten.
# ** Warning: input/muxb_tb.sv(26): (vlog-2275) Existing module 'muxb_tb' at input/muxb_test.sv(26) will be overwritten.
```

However the circuit got the synthesis successfully: and code coverage was 100%.



2.3. muxd Design

2.3.1. muxd RTL Code Creation Results

Describe Code Implementation Principles and Check the Results

Give a short description on how the code was implemented (you can include the relevant parts). Comment on Lint, AutoCheck and synthesis results.
 	
Description:The muxd module is a parameterized N-input, 16-bit multiplexer.
It uses a parameter N to set the number of data inputs and $clog2(N) to define the width of the select input.
Inside an always_comb block, the code checks if sel_in is less than N.
If true, m_out is assigned the selected 16-bit input (d_in[sel_in]); otherwise, it outputs 0
This makes the design flexible for any number of inputs.
And the overall code is:
module muxd #(
  parameter int N = 8 // Number of 16-bit data inputs
) (
  input logic [$clog2(N)-1:0] sel_in, // Select input
  input logic [N-1:0][15:0] d_in, // Data input array
  output logic [15:0] m_out // Data output
);
 	
  // Combinational logic: selects d_in[sel_in] if valid, else 0
  always_comb begin : mux_logic
  if (sel_in < N)
  m_out = d_in[sel_in];
  else
  m_out = 16'h0000;
  end
 	
endmodule

Lint result:
==========
==========
Lint Check Report
Questa Lint Version 2022.4_1 5450673 linux_x86_64 22-Nov-2022

Timestamp : Sun Nov 2 21:28:09 2025
Description : Report for referring checks count, check violations details, and design information
Design : muxd
Database : /homedir01/amiah25/DT2_2025/project/workdir/output/qlint/lint.db
Design Quality Score : 99.5%

AutoCheck:
AutoCheck Compile Summary

Check	Evaluations	Found	Waived
ASSIGN_IMPLICIT_CONSTANT	1	0	0
CASE_DUPLICATE	0	0	0
CLK_DELAY	0	0	0
CLK_IN_DATA	0	0	0
COMBO_LOOP	1	0	0
DECLARATION_UNDRIVEN	0	0	0
DECLARATION_UNUSED	0	0	0
DECLARATION_UNUSED_UNDRIVEN	0	0	0
FUNCTION_INCOMPLETE_ASSIGN	0	0	0
INDEX_UNREACHABLE	0	0	0
LATCH_INFERRED	1	0	0
LOGIC_UNDRIVEN	0	0	0
LOGIC_UNUSED	0	0	0
PORT_UNDRIVEN	3	0	0
PORT_UNUSED	3	0	0
REG_MIXED_ASSIGNS	1	0	0
REG_NO_RESET	0	0	0
REG_RACE	0	0	0
REG_VARIABLE_ARESET	0	0	0
RESET_HIGH_LOW	0	0	0
RESET_SYNC_ASYNC	0	0	0
SLIST_INCOMPLETE	1	0	0
X_ASSIGN_REACHABLE	0	0	0
AC Total	11	0	0

AutoCheck Verify Summary

Check	Evaluations	Found	Waived
ARITH_OVERFLOW_SUB	0	0	0
ARITH_OVERFLOW_VAL	0	0	0
ARITH_ZERO_DIV	0	0	0
ARITH_ZERO_MOD	0	0	0
BLOCK_UNREACHABLE	2	1	0

```
BUS_MULTIPLY_DRIVEN              0          0          0
BUS_UNDRIVEN                     0          0          0
BUS_VALUE_CONFLICT               0          0          0
CASE_DEFAULT                     0          0          0
CASE_FULL                        0          0          0
CASE_PARALLEL                    0          0          0
FSM_DEADLOCK_STATE               0          0          0
FSM_LOCKOUT_STATE                0          0          0
FSM_STUCK_BIT                    0          0          0
FSM_UNREACHABLE_STATE            0          0          0
FSM_UNREACHABLE_TRANS            0          0          0
INDEX_ILLEGAL                    1          0          0
INIT_X_OPTIMISM                  0          0          0
INIT_X_PESSIMISM                 0          0          0
INIT_X_UNRESOLVED                0          0          0
INIT_X_UNRESOLVED_MEM            0          0          0
ONE_COLD                         0          0          0
ONE_HOT                          0          0          0
REG_MULTIPLY_DRIVEN              0          0          0
REG_STUCK_AT                     0          0          0
REG_TOGGLE_VIOLATION             0          0          0
--------------------------------------------------------------------------------
AC Total                         3          1          0
```

Synthesis result:

```
################################################################################
################################################
#
# WRITE OUT RESULTS
#
################################################################################
################################################
report_timing
Information: Updating design information... (UID-85)

****************************************
Report : timing
        -path full
        -delay max
        -max_paths 1
Design : muxd
Version: T-2022.03-SP5-1
Date   : Sun Nov  2 21:31:38 2025
****************************************

Operating Conditions: tt1p05v25c   Library: saed32hvt_tt1p05v25c
Wire Load Model Mode: enclosed

  Startpoint: sel_in[2] (input port)
  Endpoint: m_out[0] (output port)
  Path Group: default
  Path Type: max

  Des/Clust/Port     Wire Load Model       Library
  ------------------------------------------------
```

```
 muxd              8000                saed32hvt_tt1p05v25c

 Point                              Incr       Path
 -----------------------------------------------------------
 input external delay                  0.00       0.00 f
 sel_in[2] (in)                        0.00       0.00 f
 U93/Y (INVX0_HVT)                     0.03       0.03 r
 U95/Y (AND3X1_HVT)                    0.13       0.17 r
 U97/Y (AO22X1_HVT)                    0.13       0.30 r
 U107/Y (OR4X1_HVT)                    0.14       0.45 r
 m_out[0] (out)                        0.00       0.45 r
 data arrival time                                0.45

 max_delay                             4.00       4.00
 output external delay                 0.00       4.00
 data required time                               4.00
 -----------------------------------------------------------
 data required time                               4.00
 data arrival time                               -0.45
 -----------------------------------------------------------
 slack (MET)                                      3.55


1
report_area -hierarchy > reports/${DESIGN_NAME}.gatelevel.area.txt
report_timing > reports/${DESIGN_NAME}.gatelevel.timing.txt
report_qor > reports/${DESIGN_NAME}.gatelevel.qor.txt
read_saif -auto_map_names -input results/${DESIGN_NAME}.saif -instance_name
${DESIGN_NAME}_tb/DUT -verbose
Error: Cannot find the SAIF file (results/muxd.saif). (PWR-201)
0
report_power > reports/${DESIGN_NAME}.gatelevel.power.txt
if { ${DESIGN_NAME} == "mycpu" } { report_power -hierarchy >>
reports/${DESIGN_NAME}.gatelevel.power.txt }
report_reference  -hierarchy > reports/${DESIGN_NAME}.gatelevel.reference.txt
change_names -rules verilog -hierarchy
1
write -hierarchy -format verilog -output results/${DESIGN_NAME}_gatelevel.v
Writing verilog file '/homedir01/amiah25/DT2_2025/project/workdir/results/muxd_gatelevel.v'.
1
write_sdf -version 2.1 results/${DESIGN_NAME}_gatelevel.sdf
Information: Annotated 'cell' delays are assumed to include load delay. (UID-282)
Information: Writing timing information to file
'/homedir01/amiah25/DT2_2025/project/workdir/results/muxd_gatelevel.sdf'. (WT-3)
1
reset_timing_derate
1
write_sdc -version 1.7 results/${DESIGN_NAME}_gatelevel.sdc
1
saif_map -write_map results/${DESIGN_NAME}_rtl.saifmap
Information: Writing SAIF name mapping information to file 'results/muxd_rtl.saifmap'. (PWR-635)
1
gui_start
Current design is 'muxd'.
4.1.1
```

```
gui_create_schematic
Current design is 'muxd'.
Schematic.1
print_message_info

Id            Severity        Limit    Occurrences   Suppressed
------------------------------------------------------------------------
CMD-005          Error            0           12           12
CMD-025          Error            0            1            0
OPT-1208     Information          0            1            0
OPT-775      Information          0            1            0
OPT-780      Information          0            1            0
PWR-201          Error            0            1            0
PWR-24       Information          0            1            0
PWR-414          Warning          0            1            0
PWR-536      Information          0            6            0
PWR-6        Information          0            2            0
PWR-602      Information          0            1            0
PWR-635      Information          0            1            0
PWR-789      Information          0            1            0
PWR-80           Warning          0            2            0
PWR-850      Information          0            1            0
UID-282      Information          0            1            0
UID-85       Information          0            2            0
UISN-27      Information          0            1            0
UPF-213          Warning         10            0            0
UPF-213a         Warning         10            0            0
UPF-213b         Warning         10            0            0
UPF-213c         Warning         10            0            0
UPF-213d         Warning         10            0            0
UPF-213e         Warning         10            0            0
UPF-213f         Warning         10            0            0
WT-3         Information          0            1            0

Diagnostics summary: 2 errors, 3 warnings, 21 informationals

Note: Even though there have 2 errors, but the design runs successfully.
```

2.3.2. muxd Test Program Implementation and Simulation Results

Describe Code Implementation Principles and Verification Results
Give a short description on how the test bench and test program code was implemented (you can include the relevant parts). Comment on simulation results and code coverage. Description: The testbench (muxd_tb) connects the DUT (muxd) and the test program (muxd_test) using shared signals. The test program runs automatically and generates random data for all inputs and random

select values.
It calculates the expected output and compares it to the DUT output using assertions.
If they match, it prints “PASS,” otherwise it shows “FAIL.”
This verifies that the multiplexer works correctly for different input combinations.

The simulation results are:

```
# SIMULATION END (0 assertions failed)
# --------------------------------------------------------------------------------
# 0 ps
# 105 ns
# ** UI-Msg (Warning): (vsim-3315) Power checking is not enabled.
#
# ** Warning: (vsim-19043) Specified instance '/muxd_tb/DUT' not found.
# ** Note: (vsim-17388) No matching coverage data found.
# End time: 21:27:59 on Nov 02,2025, Elapsed time: 0:00:39
# Errors: 0, Warnings: 1
```

SELF ASSESSMENT

Write a summary of week's work and results, including

1. Assessment of results
    a. Does the code compile and elaborate without errors and warnings?
       Yes.
    b. Do the RTL modules function, pass checks and synthesize correctly
       Yes
    c. Do the test programs function correctly and provide 100% RTL code coverage?
       Yes
2. Approximate time spent on project this week
   I spend around 6-8 hours, first watch the videos carefully and then design the codes by myself , but sometimes I get AI help to understand some blacks , as well as how to track the codes
3. Open issues you need help with
4. Problems found in earlier weeks' code fixed this week

WEEK3: ARITHMETIC-LOGIC CIRCUIT DESIGN (fu)

LEARNING GOALS
1. Learn the principles of modeling combinational logic based arithmetic functions.

3.1. fu Design

3.1.1. fu RTL Code Creation and Simulation Results

Describe the implementation of the function for which your randomizer L matches any of the numbers in column L, and present simulation results (PASS/FAIL) for **all** functions.

Implementation description and simulation results (Pass/Fail)			
fs_in	**L= 2**	**Present relevant code and describe and justify word-length and sign handling where applicable.**	**Sim. Resul**
FMOVA	0,5,8		
FINC	1,6,9,		
FADD	2,7,0	f_out= signed'({1'b0, a_in}) + signed'({1'b0, b_in});. using 17-bit signed arithmetic for correct sign handling and overflow prevention.	pass
FMUL	3,8,1		
FSRA	4,9,2	f_out=signed'({{(MUL_WIDTH-DATA_WIDTH){1'b0}}, a_in}) >>> 1; Uses the SystemVerilog arithmetic shift right operator () on a 32-bit signed version of to preserve the sign bit.	pass
FSUB	5,0,3		
FDEC	6,1,4		
FSLA	7,2,5	f_out= signed'({{(MUL_WIDTH-DATA_WIDTH){1'b0}}, a_in}) <<< 1; Uses the arithmetic shift left operator () on a 32-bit signed version of . (Note: For a single shift, it behaves like logical shift.)	pass
FAND	8,3,6		
FOR	9,4,7		
FXOR	0,5		
FNOT	1,6		
FMOVB	2,7	f_out= signed'({{(MUL_WIDTH-DATA_WIDTH){1'b0}}, b_in});	pass

FSHR	3,9		
FSHL	4,0		

General Comments
E.g. about failed simulation results, if applicable. The code runs successfully.

3.1.2. fu RTL Code Check Results

Check Results
Comment on Lint, AutoCheck and synthesis results. The lint comments are: Section 2 : Check Details ========== ========== ---------- | Error (10) | ---------- Check: assign_width_underflow [Category: Rtl Design Style] (10) [Message: Width of assignment RHS is less than width of LHS. LHS Expression '<lhs_expression>', LHS Width '<lhs_width>', RHS Width '<rhs_width>', Module '<module>', File '<file>', Line '<line>'.] ---------- ---------- assign_width_underflow: [uninspected] Width of assignment RHS is less than width of LHS. LHS Expression 'result_temp', LHS Width '32', RHS Width '18', Module 'fu', File '/homedir01/amiah25/DT2_2025/project/workdir/input/fu.sv', Line '48'. [RTL ID:b1c2bd7f_00200] [Example Hierarchy:fu] assign_width_underflow: [uninspected] Width of assignment RHS is less than width of LHS. LHS Expression 'result_temp', LHS Width '32', RHS Width '18', Module 'fu', File '/homedir01/amiah25/DT2_2025/project/workdir/input/fu.sv', Line '54'. [RTL ID:fd509884_00200] [Example Hierarchy:fu] assign_width_underflow: [uninspected] Width of assignment RHS is less than width of LHS. LHS Expression 'result_temp', LHS Width '32', RHS Width '18', Module 'fu', File '/homedir01/amiah25/DT2_2025/project/workdir/input/fu.sv', Line '72'. [RTL ID:4f518bdc_00200] [Example Hierarchy:fu] assign_width_underflow: [uninspected] Width of assignment RHS is less than width of LHS. LHS Expression 'result_temp', LHS Width '32', RHS Width '18', Module 'fu', File '/homedir01/amiah25/DT2_2025/project/workdir/input/fu.sv', Line '79'. [RTL ID:16337ebf_00200] [Example Hierarchy:fu] assign_width_underflow: [uninspected] Width of assignment RHS is less than width of LHS. LHS

Expression 'result_temp', LHS Width '32', RHS Width '16', Module 'fu', File '/homedir01/amiah25/DT2_2025/project/workdir/input/fu.sv', Line '90'. [RTL ID:cde5c776_00200] [Example Hierarchy:fu]

assign_width_underflow: [uninspected] Width of assignment RHS is less than width of LHS. LHS Expression 'result_temp', LHS Width '32', RHS Width '16', Module 'fu', File '/homedir01/amiah25/DT2_2025/project/workdir/input/fu.sv', Line '95'. [RTL ID:6116db68_00200] [Example Hierarchy:fu]

assign_width_underflow: [uninspected] Width of assignment RHS is less than width of LHS. LHS Expression 'result_temp', LHS Width '32', RHS Width '16', Module 'fu', File '/homedir01/amiah25/DT2_2025/project/workdir/input/fu.sv', Line '100'. [RTL ID:99be3245_00200] [Example Hierarchy:fu]

assign_width_underflow: [uninspected] Width of assignment RHS is less than width of LHS. LHS Expression 'result_temp', LHS Width '32', RHS Width '16', Module 'fu', File '/homedir01/amiah25/DT2_2025/project/workdir/input/fu.sv', Line '105'. [RTL ID:6618c166_00200] [Example Hierarchy:fu]

assign_width_underflow: [uninspected] Width of assignment RHS is less than width of LHS. LHS Expression 'result_temp', LHS Width '32', RHS Width '16', Module 'fu', File '/homedir01/amiah25/DT2_2025/project/workdir/input/fu.sv', Line '116'. [RTL ID:131d1816_00200] [Example Hierarchy:fu]

assign_width_underflow: [uninspected] Width of assignment RHS is less than width of LHS. LHS Expression 'result_temp', LHS Width '32', RHS Width '16', Module 'fu', File '/homedir01/amiah25/DT2_2025/project/workdir/input/fu.sv', Line '122'. [RTL ID:65c5e201_00200] [Example Hierarchy:fu]

Auto check:
AutoCheck Verify Summary

Check	Evaluations	Found	Waived
ARITH_OVERFLOW_SUB	0	0	0
ARITH_OVERFLOW_VAL	5	0	0
ARITH_ZERO_DIV	0	0	0
ARITH_ZERO_MOD	0	0	0
BLOCK_UNREACHABLE	16	0	0
BUS_MULTIPLY_DRIVEN	0	0	0
BUS_UNDRIVEN	0	0	0
BUS_VALUE_CONFLICT	0	0	0
CASE_DEFAULT	1	0	0
CASE_FULL	0	0	0
CASE_PARALLEL	0	0	0
FSM_DEADLOCK_STATE	0	0	0
FSM_LOCKOUT_STATE	0	0	0
FSM_STUCK_BIT	0	0	0
FSM_UNREACHABLE_STATE	0	0	0
FSM_UNREACHABLE_TRANS	0	0	0
INDEX_ILLEGAL	0	0	0
INIT_X_OPTIMISM	0	0	0
INIT_X_PESSIMISM	0	0	0
INIT_X_UNRESOLVED	0	0	0
INIT_X_UNRESOLVED_MEM	0	0	0

ONE_COLD	0	0	0
ONE_HOT	0	0	0
REG_MULTIPLY_DRIVEN	0	0	0
REG_STUCK_AT	0	0	0
REG_TOGGLE_VIOLATION	0	0	0
AC Total	22	0	0

Synthesis result: Diagnostics summary: 1 error, 15 warnings, 23 informationals

SELF ASSESSMENT

Write a summary of the week's work and results, including

1. Assessment of results
    a. Does the code compile and elaborate without errors and warnings?
    The code simulates without errors and warnings
    b. Do the RTL modules function, pass checks, and synthesize correctly
    I ran the synthesis, which showed 1 error and 15 warnings, but the circuit and operation were successful.
2. Approximate time spent on project this week: I have to give 6-8 hours for reviewing the lecture video and solidifying my basics, then I spent not much time because with the basics strong, I get the result immediately
3. Open issues you need help with

I get the help from the professor as I was confused about how to write the result of 3.1.1. fu RTL Code Creation and Simulation Results

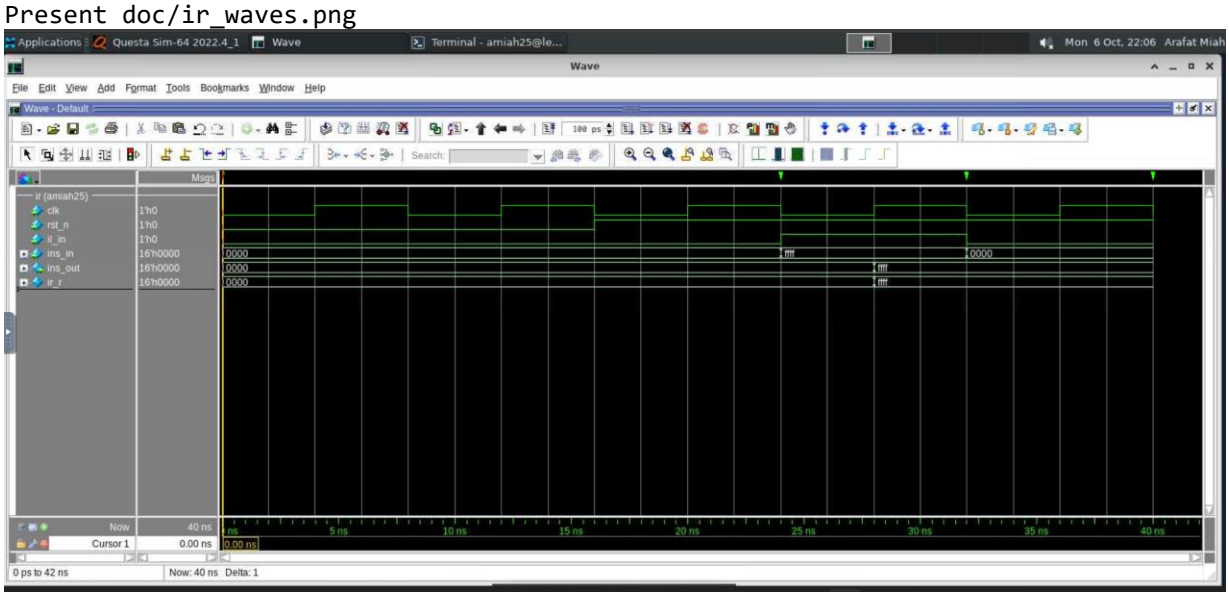
4. Problems found in earlier weeks' code fixed this week
    Yeah

WEEK4: SEQUENTIAL LOGIC DESIGN 1 (ir, pc)

LEARNING GOALS
1. Learn the principles of modeling sequential logic and checking synthesis results. 2. Learn the basics of designing a test program for sequential logic.

4.1. ir Design

4.1.1. ir Simulation Results

Simulation Waveforms and Comment the Results
`Present doc/ir_waves.png` 
Comment reset test, load test and hold test results, referring to the waveforms. Based on the waveform image: **Reset test** (Initial state): rst_n is low, forcing ins_out and ir_r to 0000 asynchronously; **Load test** (e.g., around 24ns): ins_out updates synchronously on posedge clk to match ins_in only when il_in is high (1); **Hold test** (e.g., after 24ns): ins_out remains stable when il_in returns to low (0), demonstrating data retention. N.B. This part was furnished with the help of AI.

4.1.2. ir Lint Check Results

See report **reports/ir.rtl.qlint_report.tx**t

Lint Warnings and Errors: There is no error.		
Type	ID	Comment

4.1.3. ir AutoCheck Results

See report **reports/ir.rtl.qautocheck_report.txt** Section AutoCheck Details.

AutoCheck Warnings and Errors:There was also no error in this part		
Type	ID	Comment

4.1.4. ir Synthesis Results

Synthesis Warnings and Errors: Diagnostics summary: 1 error, 1 warning, 20 informationals		
Type	ID	Comment
Error	**CMD-005**	This is occuring 12 times, which indicates in synthesis script command it is repeating over and over
Error	**CMD-025**	This is indicating fatal execution error which occurs due to missing setup information.

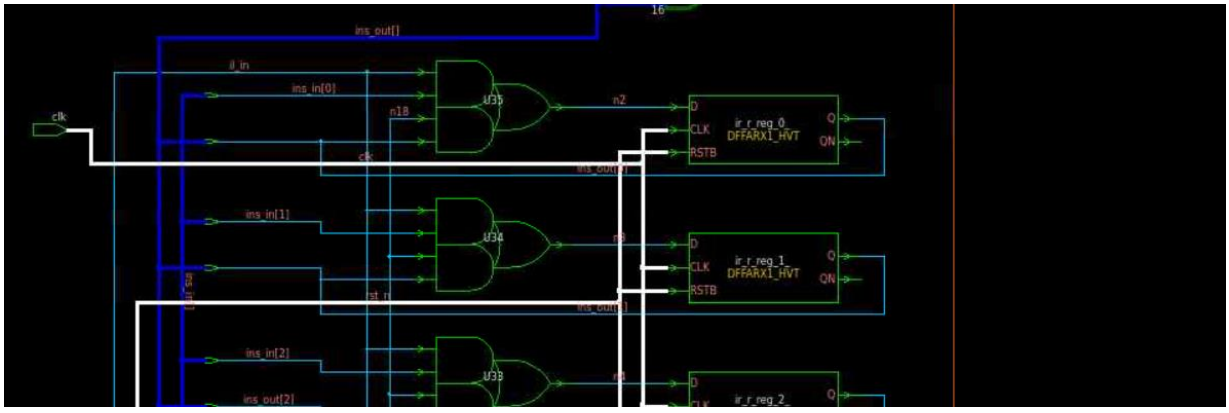
See report **reports/ir.rtl.synthesis_log.txt**.

Register Inference Analysis

Paste doc/ir_register_inference.png here

```
nferred memory devices in process
        in routine ir line 20 in file
                './input/ir.sv'.
===============================================================================
|    Register Name    |   Type    | Width | Bus | MB | AR | AS | SR | SS | ST |
===============================================================================
|      ir_r_reg       | Flip-flop |  16   |  Y  | N  | Y  | N  | N  | N  | N  |
===============================================================================
```

Paste doc/ir_detail.png



Describe what were the expected results and comment on the actual results.

The expected result was 16-bit synchronous flip-flop with an asynchronous reset (AR=Y) and we can see that the hardware structure confirms that

See report **reports/ir.rtl.synthesis_log.txt**.

References Report Analysis

```
Reference           Library       Unit Area   Count    Total Area   Attributes
--------------------------------------------------------------------------------
AO22X1_HVT          saed32hvt_tt1p05v25c
                                  2.541440      16     40.663040
DFFARX1_HVT         saed32hvt_tt1p05v25c
                                  7.116032      16    113.856514  n
INVX2_HVT           saed32hvt_tt1p05v25c
                                  1.524864       1      1.524864
```

```
--------------------------------------------------------------------------------
Total 3 references                                          156.044418
```

Comments (is the flip-flop count ok, are there dubious cells)
The flip flop is ok and no dubious cells

See report **reports/ir.gatelevel.references.txt**.

4.2. pc Design

4.2.1. pc RTL Code Creation Results

Describe Code Implementation Principles and Check the Results

Give a short description on how the code was implemented (you can include the relevant parts)- Comment on Lint, AutoCheck and synthesis results.
The Program Counter (PC) was implemented in **SystemVerilog RTL** using a **two-block structure**: an always_comb block determines the next_pc_value based on the control signals (Increment, Jump, Branch), and a separate **synchronous always_ff block** registers this value on the clock edge. Crucially, the Branch logic includes **sign-extension** of the 6-bit offset to correctly handle both positive (forward) and negative (backward) jumps.

```
Qlint: Design Information
================================================================================
----------
| Summary |
----------
  Register Bits                  : 16
  Latch Bits                     : 0
  User-specified Blackboxes      : 0
  Inferred Blackboxes            : 0
  Empty Modules                  : 0
  Unresolved Modules             : 0
  Hierarchical IPs               : 0


AutoCheck Details
--------------------------------------------------------

Type      : ARITH_OVERFLOW_VAL
Severity  : Caution
Status    : Uninspected
Module    : pc
Name      : pc_r
Location  : /homedir01/amiah25/DT2_2025/project/workdir/input/pc.sv:33
Assignment Width: 16
Waveform Distance: 2

Type      : ARITH_OVERFLOW_VAL
Severity  : Caution
Status    : Uninspected
Module    : pc
Name      : pc_r
```

```
Location  : /homedir01/amiah25/DT2_2025/project/workdir/input/pc.sv:34
Assignment Width: 16
Waveform Distance: 2

Type      : PORT_UNUSED
Severity  : Caution
Status    : Uninspected
Module    : pc
Name      : ins_in[15:9]

Type      : PORT_UNUSED
Severity  : Caution
Status    : Uninspected
Module    : pc
Name      : ins_in[5:3]

Inferred memory devices in process
        in routine pc line 26 in file
                './input/pc.sv'.
===============================================================================
=
|    Register Name    |   Type    | Width | Bus | MB | AR | AS | SR | SS | ST |
===============================================================================
=
|     pc_r_reg       | Flip-flop |  16   |  Y  | N  | Y  | N  | N  | N  | N  |
===============================================================================
```

4.2.2. pc Verification Plan Design

Describe the tests you defined.

ID	**Stimulus**	**Response**
T1	Reset applied from testbench module	All registers in known states (checked by concurrent assertions in pc_svamod.sv)
T2	**Increment (ps_in=01):** Applied for two consecutive clock cycles.	program_counter_out increases by 1 in each cycle (PC + 1).
T3	**Jump (ps_in=11):** A fixed, arbitrary address (0xABCD) applied via ra_in.	program_counter_out immediately loads the ra_in value (0xABCD)
T4	**Branch Positive (ps_in=10):** Instruction word containing a positive 6-bit offset is applied.	program_counter_out equals PC + positive_offset (forward branch).

T5	**Branch Negative (ps_in=10):** Instruction word containing a negative 6-bit offset is applied.	program_counter_out equals PC - absolute_offset (backward branch via two's complement).
T6	**Hold (ps_in=00):** Applied while all other inputs (ins_in, ra_in) change.	program_counter_out remains constant (unchanged from previous cycle).

4.2.3. pc Test Program Implementation and Simulation Results

Describe Code Implementation Principles and Verification Results

Give a short description on how the test bench and test program code was implemented (you can include the relevant parts).
Comment on simulation results and code coverage.
At first I was successfully design my circuit respec tive code but when I was design the test bench I encounter 4-5 error and it was difficult to correct those error. So I go for group study with my batch mate whom I mentioned in this Week4 part. So the main code I design by myself with the comenting is :

```
`include "mycpu.svh"

import mycpu_pkg::*;

module pc (
    input  logic       clk,
    input  logic       rst_n,
    input  logic [1:0]  ps_in,
    input  logic [15:0] ins_in,
    input  logic [15:0] ra_in,
    output logic [15:0] pc_out
);

    // Sequential register for PC
    logic [15:0] pc_r;
    logic signed [5:0] ba; // signed branch address offset

    // Extract and form signed 6-bit branch address (ba)
    // Concatenate ins_in[8:6] and ins_in[2:0]
    assign ba = {ins_in[8:6], ins_in[2:0]};

    // Combinational output
    assign pc_out = pc_r;

    // Sequential process for updating PC
    always_ff @(posedge clk or negedge rst_n) begin
        if (!rst_n) begin
            pc_r <= 16'd0;              // Reset to 0
        end
        else begin
            case (ps_in)
                2'b00: pc_r <= pc_r;              // No change
                2'b01: pc_r <= pc_r + 16'd1;      // Increment PC
                2'b10: pc_r <= pc_r + ba;         // Branch offset add
```

```
            2'b11: pc_r <= ra_in;              // Jump to address from ra_in
            default: pc_r <= pc_r;
          endcase
      end
   end

endmodule
```

3.2.3. pc RTL Code Check Results

Check Results
Comment on Lint, AutoCheck and synthesis results (register inference, flip-flop count, cell references). The analysis which is given in 4.2.1. pc RTL Code Creation Results confirms the design successfully inferred **16 flip-flops** for the pc_r_reg (the program counter register) with **zero latches**, which is ideal for synchronous design. The **AutoCheck** flagged expected **unused bits** in the instruction input (ins_in) and potential **arithmetic overflow** warnings, both of which are common and acceptable given the 16-bit register width and the nature of the instruction format.

SELF ASSESSMENT
Write a summary of week's work and results, including 1. Assessment of results a. Does the code compile and elaborate without errors and warnings? No, the code first compile without erros bt when I add the test bench it shows 4-5 error, So I need to take help from the teacher and then I do some group work with ASHRAF UDDIN SHEIKH <2508787> and <Mohammad Ammar Abdur Rahman Khan> <Student Number> 2509732 ,with them we write the test bench successfully without any errors and warnings. And also I used some help from AI to get some explanation which taught me what's going on in any test bench or simulation result whenever I stuck in any situation. b. Do the RTL modules function, pass checks and synthesize correctly Yes. c. Doed pc test program function correctly and provide 100% RTL code coverage Yes 2. Approximate time spent on project this week: As I am a new student and there was two mid final exam (ESD and ED2), so, I did not get enough time to spend on this project, Spend around 6-8 hours in thoery and then 10-12 hours in the lab reports and projects…Usually I give more time to do this coding as it aligns with my research work 3. Open issues you need help with I think the test bench will be taught in the next lab, so it was hard without that knowledge how to write big code's testbench. So the test bench lecture which is 9 and 10 should be done before this lab 4. Problems found in earlier weeks' code fixed this week yes

WEEK5: SEQUENTIAL LOGIC DESIGN 2 (rb)

LEARNING GOALS
1. Learn the principles of designing and modeling complex sequential logic functions. 2. Learn to design a test program for sequential logic.

5.1. rb Design

5.1.1. rb RTL Specification

SIGNALS				
Name	Kind	Width (bits)	Type	Description
rb_r	S	128	logic [7:0][15:0]	User register bank register bits. 8 registers with 16 bits in each. Reset state = 0.
hb_r	S	128	logic [7:0][15:0]	Hidden register bank register bits. 8 registers with 16 bits in each. Reset state = 0.
Kind: C = combinational, S = sequential, I = interconnect (assign) Description: Short description of purpose of this signal. Include reset state for sequential signals.				

BLOCKS (PROCESSES)					
Block Name	**Kind**	**Inputs**	**Outputs**	**#FFs**	**Description of the function of the block**
write_proc	**S**	**clk, rst_n, rw_in, d_in, rs_in[11:8]**	**rb_r, hb_r**	**16 × 16 = 256 FFs**	**Here the Sequential process handles asynchronous reset and conditional write to either the user or hidden bank based on rs_in[11:8].**
read_a_proc	**C**	**rs_in[7:4], rb_r, hb_r**	**a_out**	**0**	**Combinational read process selecting one register output for a_out.**
read_b_proc	**C**	**rs_in[3:0], rb_r,**	**B_out**	**0**	**Combinational read process selecting one register output for b_out.**

		hb_r			

Kind: C = combinational, S = sequential, I = interconnect (assign)
Inputs: Names of variables or ports that are inputs of the block (clock and reset can be omitted)
Names: Variables or ports driven from the block
#FFs: Expected number of flip-flops
Description: Short description of purpose of this block

5.1.2. rb RTL Code Creation Results

Describe Code Implementation Principles and Check the Results

Give a short description on how the code was implemented (you can include the relevant parts).

The design follows clear partitioning into one sequential and two combinational blocks.
Now I will write the code representing the blocks:
BLOCK 1: SEQUENTIAL PROCESS (Storage and Write Control)

```
   // Implements the register storage elements (Flip-Flops) and their control logic.
   always_ff @(posedge clk or negedge rst_n) begin
      if (!rst_n) begin
         rb_r <= '0;
         hb_r <= '0;
      end else if (rw_in) begin
            if (w_addr[3] == 1'b0) begin
             rb_r[w_addr[2:0]] <= d_in;
         end else begin // w_addr[3] == 1'b1
                hb_r[w_addr[2:0]] <= d_in;
         end
      end
   end
```

BLOCK 2: COMBINATIONAL PROCESS (Read A Multiplexer)

```
   always_comb begin
      if (a_addr[3] == 1'b0) begin
         a_out = rb_r[a_addr[2:0]];
      end else begin
         a_out = hb_r[a_addr[2:0]];
      end
   end
```

BLOCK 3: COMBINATIONAL PROCESS (Read B Multiplexer)

```
   always_comb begin
      if (b_addr[3] == 1'b0) begin
         b_out = rb_r[b_addr[2:0]];
      end else begin
```

```
        b_out = hb_r[b_addr[2:0]];
    end
end
```

5.1.3. rb Test Program Implementation and Simulation Results

Describe Code Implementation Principles and Verification Results

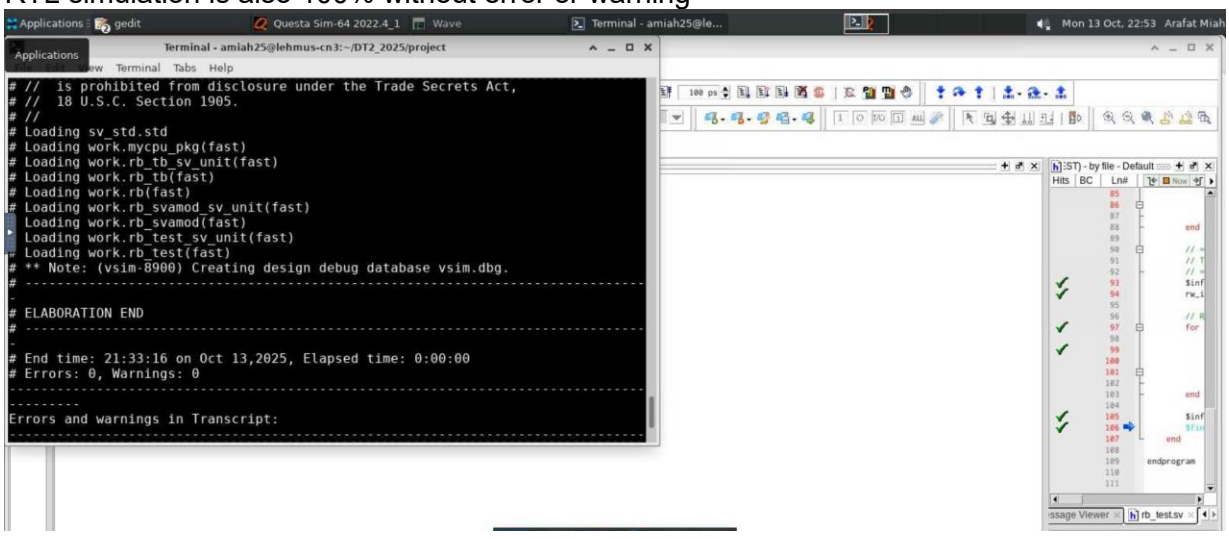
Give a short description on how the test bench and test program code was implemented (you can include the relevant parts).
Comment on simulation results and code and functional coverage.

The implementation is describe below:

- The test program **(`rb_test.sv`)** drives the DUT via clock, reset, and control signals.
- Three verification phases correspond directly to the specification:

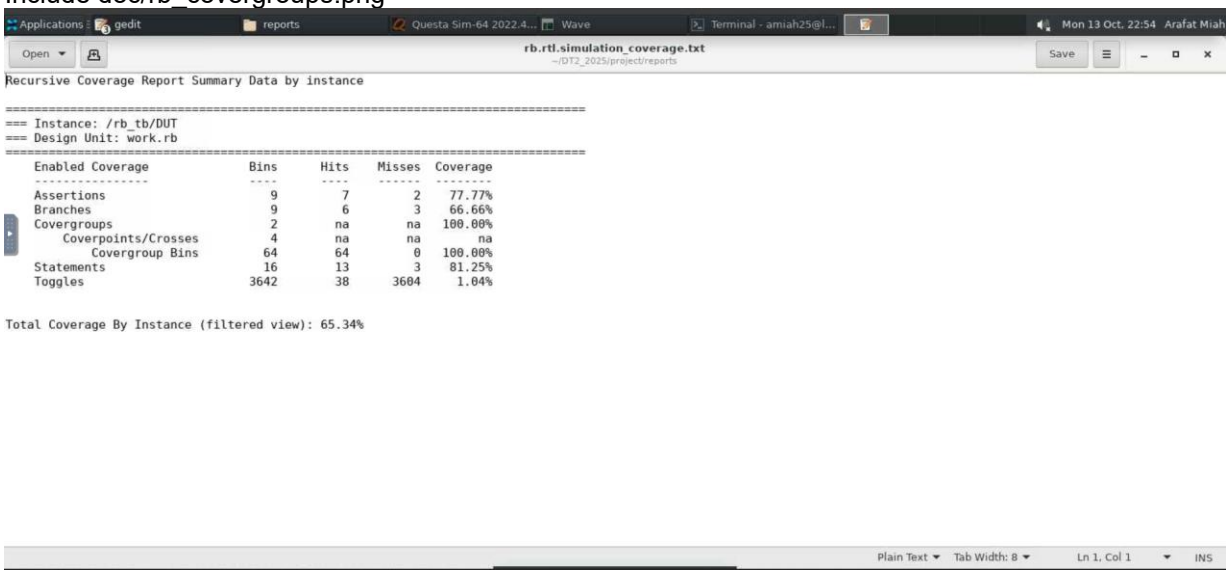
Test ID	Purpose	Verification Method
T1	Reset test – verify all registers = 0 after reset	Concurrent assertions
T2	Write–Read test – write unique non-zero values to all 16 registers and read back	Immediate assertions per register
T3	Read test – verify stored values remain correct with write disabled	Immediate assertions per register

RTL simulation is also 100% without error or warning



Covergroup coverage

Include doc/rb_covergroups.png



5.1.4. rb RTL Code Check Results

Check Results
Lint: pass successfully without creating any latch and reset defines for all FFs. Comment on Lint, AutoCheck and synthesis results. Section 3 : Design Information ========== ========== ---------- | Summary | ---------- Register Bits : 256 Latch Bits : 0 User-specified Blackboxes : 0 Inferred Blackboxes : 0 Empty Modules : 0 Unresolved Modules : 0 Hierarchical IPs : 0 Autocheck: Pass successfully Synthesis Result Diagnostics summary: 1 error, 6 warnings, 21 informationals (Though there have 1 error , but I check the synthesis report file thoroughly but could to find any error)

SELF ASSESSMENT

Write a summary of week's work and results, including

1. Assessment of results
    a. Does the code compile and elaborate without errors and warnings?
    Ans: The RTL and testbench compile and elaborate without errors or warnings.
    b. Do the RTL modules function, pass checks and synthesize correctly
    Ans:Yes the rb module function correctly and passes all the check and synthesize correctly.
    c. Does rb test program function correctly and provide 100% RTL code and functional coverage
    Ans: Yes the rb test program function correctly and provide almost full coverage
2. Approximate time spent on project this week
    I spent 4 hours to watch the videos and average of 15 hours for the simulations.
3. Open issues you need help with
    It was hard for writing all the long codes at the beginning phase. I would suggest first give some small task to learn and then go for the bigger one.
4. Problems found in earlier weeks' code fixed this week
    Ans: I did not get the time to do the earlier weeks, but I promise I will go through this week.

WEEK6: CONTROL LOGIC DESIGN 1 (cu)

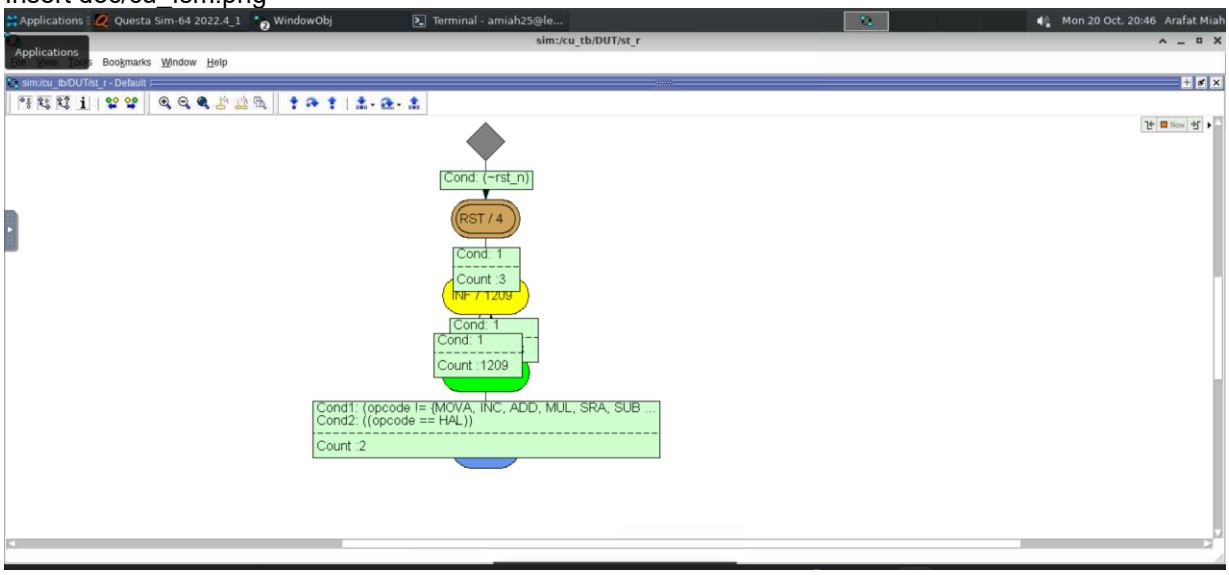
LEARNING GOALS
1. Learn the principles of control logic design for a finite-state-machine with data path architecture.

6.1. cu Design

6.1.1. cu RTL Code Creation Results

Present FSM diagram for cu.

Insert doc/cu_fsm.png



Describe Check the Results

Comment on Lint, AutoCheck and synthesis results.

Lint Check Report
Questa Lint Version 2022.4_1 5450673 linux_x86_64 22-Nov-2022

Timestamp : Mon Oct 20 21:00:20 2025
Description : Report for referring checks count, check violations details, and design information
Design : cu
Database : /homedir01/amiah25/DT2_2025/project/workdir/output/qlint/lint.db
Design Quality Score : 100%

Sections:
  Section 1 : Check Summary
  Section 2 : Check Details

```
  Section 3 : Design Information
================================================================================
======================


================================================================================
======================
Section 1 : Check Summary
================================================================================
======================
-------------
| Error (0) |
-------------

---------------
| Warning (0) |
---------------

-------------
| Info (25) |
-------------
  combo_path_input_to_output            : 7
  feedthrough_path                      : 1
  line_char_large                       : 3
  module_output_not_registered          : 8
  reserved_keyword                      : 5
  always_signal_assign_large            : 1

---------------
| Resolved (0) |
```

Synthesis Result:

Recursive Coverage Report Summary Data by instance

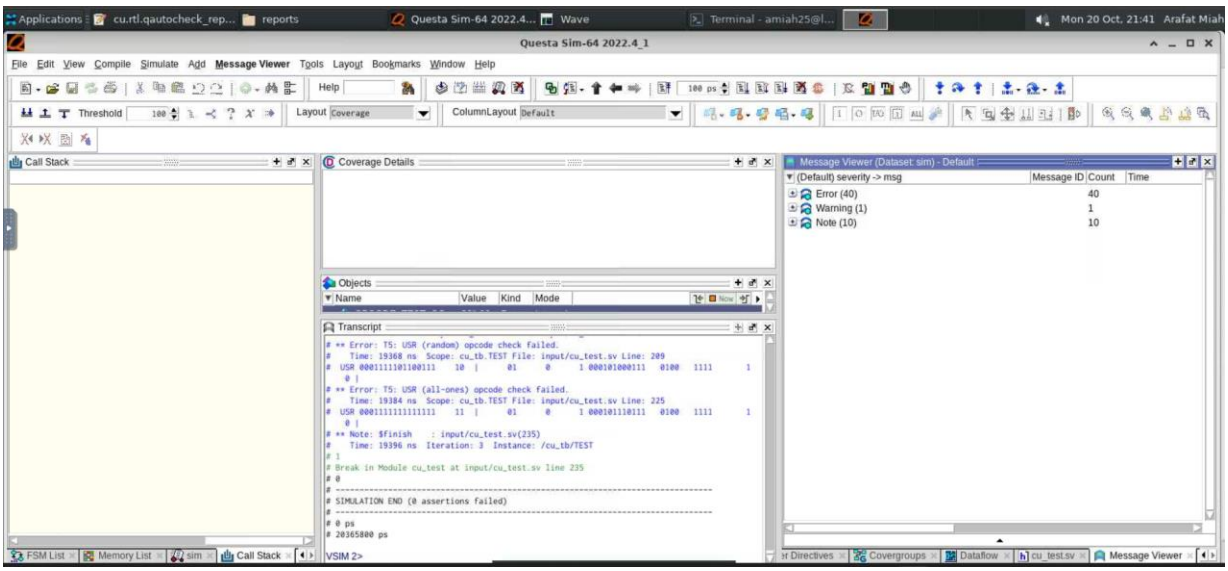
```
================================================================================
===
=== Instance: /cu_tb/DUT
=== Design Unit: work.cu
================================================================================
===
    Enabled Coverage            Bins      Hits    Misses  Coverage
    ----------------            ----      ----    ------  --------
    Assertions                    20        20         0   100.00%
    Branches                      37        37         0   100.00%
    Directives                    7         7          0   100.00%
    FSM States                      4         4         0   100.00%
    FSM Transitions                 7         5         2    71.42%
    Statements                    95        95         0   100.00%
    Toggles                      268       195        73    72.76%


Total Coverage By Instance (filtered view): 92%
```

AutoCheck Compile Summary

Check	Evaluations	Found	Waived
ASSIGN_IMPLICIT_CONSTANT	3	0	0
CASE_DUPLICATE	2	0	0
CLK_DELAY	1	0	0
CLK_IN_DATA	4	0	0
COMBO_LOOP	14	0	0
DECLARATION_UNDRIVEN	5	0	0
DECLARATION_UNUSED	5	0	0
DECLARATION_UNUSED_UNDRIVEN	0	0	0
FUNCTION_INCOMPLETE_ASSIGN	0	0	0
INDEX_UNREACHABLE	0	0	0
LATCH_INFERRED	9	0	0
LOGIC_UNDRIVEN	5	0	0
LOGIC_UNUSED	0	0	0
PORT_UNDRIVEN	12	0	0
PORT_UNUSED	12	0	0
REG_MIXED_ASSIGNS	10	0	0
REG_NO_RESET	1	0	0
REG_RACE	0	0	0
REG_VARIABLE_ARESET	2	0	0
RESET_HIGH_LOW	1	0	0
RESET_SYNC_ASYNC	1	0	0
SLIST_INCOMPLETE	9	0	0
X_ASSIGN_REACHABLE	0	0	0
AC Total	96	0	0

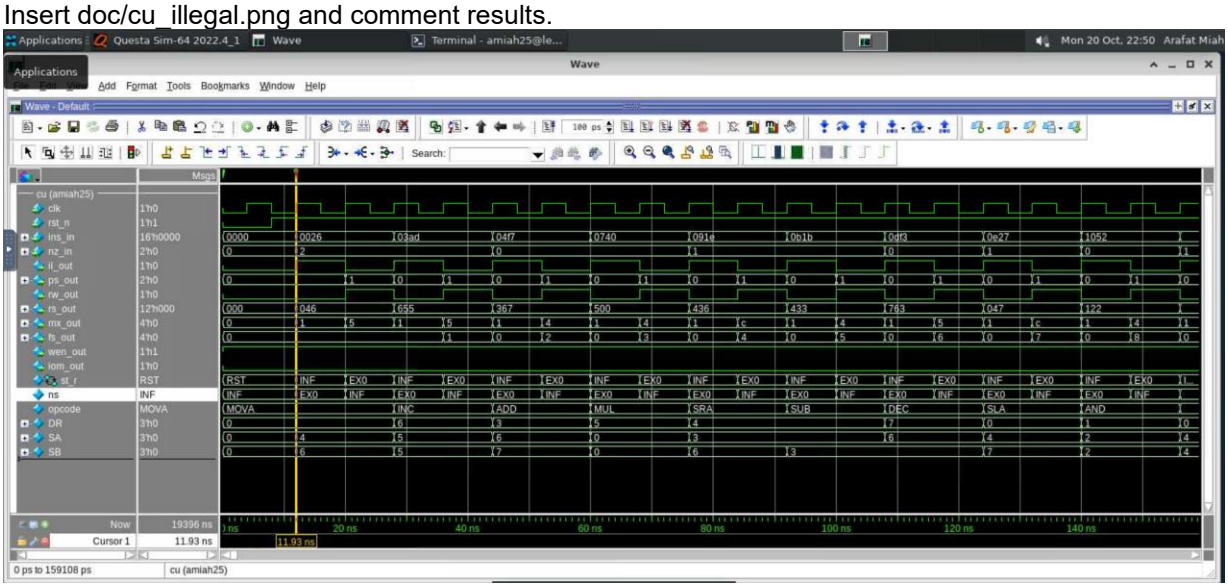
6.1.2. cu Simulation Results

Simulation message summary
Insert doc/cu_messages.png and comment results.



Illegal instruction simulation results

Insert doc/cu_illegal.png and comment results.



Code coverage

Include code coverage table for === Instance: /cu_tb/DUT from file reports/cu.rtl.simulation_coverage.txt and comment the results.

Recursive Coverage Report Summary Data by instance

==========

```
===
=== Instance: /cu_tb/DUT
=== Design Unit: work.cu
================================================================================
===
   Enabled Coverage           Bins      Hits    Misses  Coverage
   ----------------           ----      ----    ------  --------
   Assertions                   20        20         0   100.00%
   Branches                     37        37         0   100.00%
   Directives                   7         7          0   100.00%
   FSM States                     4         4         0   100.00%
   FSM Transitions                7         5         2    71.42%
   Statements                   95        95         0   100.00%
   Toggles                     268       195        73    72.76%


Total Coverage By Instance (filtered view): 92%
```

Comment: The test suite covered all code branches and FSM states, but the overall score is 92% due to 2 missed FSM transitions (71.42%) and 73 signal toggles (72.76%).

SELF ASSESSMENT

Write a summary of week's work and results, including

1. Assessment of results
    a. Does the code compile and elaborate without errors and warnings?
        - I would say it was the toughest code I have done so far. I encountered so many problem and as an ameture I was in super stress and I did not get enough time to go regularly to the class because the end of the period is going on, exams and assignment submission is regularly going on. So, I took some advice from AI about the error and then took some suggestion how to solve them and after 2 days of hard work, I somehow manages to synthesize the code. Yeah I am super happy right now.

    b. Do the RTL modules function, pass checks and synthesize correctly
    Yeah, the code functions well, pass all the checks and the synthesis was ok.
2. Approximate time spent on project this week
    Around 15-20 hours this week
3. Open issues you need help with
4. Problems found in earlier weeks' code fixed this week

WEEK7: FINITE-STATE MACHINE WITH DATAPATH DESIGN 1 (cu, fu)

LEARNING GOALS
1. Learn principles of implementing data processing function with a finite-state-machine with datapath architecture 2. Learn the principles of instruction decoder design for instruction set processors.

7.1. Special Instruction RTL Design

7.1.1. USR Instruction Description

Description of the Instruction
Present the specification of your USR instruction. USR, which is the User Special Register, performs a two-step register operation. In the first step there is stored dat of the difference between R[SA]-R[SB] into the internal holding register, and the next step is to write the value of R[SA] into register R[DR]. Also, there is an additional which writes the value of R[SB] to R[DR], which depends on the control design.

Instruction Example
Present an example of an instruction word (16-bit binary code) and explain the meaning of bit-fields in the binary code and how they are used when this instruction is decoded and executed.

INSTRUCTION WORD															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USR							DEST. REG (DR)			SOURCE REG A (SA)			SOURCE REG B (SB)		
0010							0011			0100			0101		

Explanations:

Here the targets are: DR=R3, SA=R4, SB=R5

Here bits[15:9] choose the USR code

Then, bits[8:6] choose the DR (destination register).

Bits[5:3] choose the SA (source register).

Bits[2:0] choose SB(source register).

7.1.2. USR Instruction Implementation Principle

Description of the Implementation Principle
Write a general description of how you plan to implement the instruction. Describe the datapath resources (register and function unit function) that are needed, and function unit function (FUSR) that must be added (if any), and how the instruction is executed with these. Ans: Here first the user instruction uses register file read port of SA and SB , function unit of ALU and the holding register. The execution proceeds are: subtract R[SB] FROM R[SA], then store the temporary value and finally writeback the R[SA] or R[SB] into R[DR]. Finally, The resources required in datapath are Register file, Holding register ,ALU subtraction path and the control unit state sequenceing.

Function Unit Modifications
If your implementation uses the FUSR function code, specify the function that it implements. Ans: In the function unit modification I added a new code FUSR. And when it is selected it performs R[0]← R[SA]-R[SB]

7.1.3. USR Instruction Control Algorithm

Control States and Register Transfers for Instruction USR (State Table)

STATE (st_r)	FLAGS (nx_in)	NEXT STATE (ns)	REGISTER TRANSFERS
EX0	0X	USR1	HR[0] ← R[SA] - R[SB]
EX0	1X	USR1	HR[0] ← R[SA] - R[SB]

USR1	XX	INF	R[DR] ← R[SA]
USR2	XX	INF	R[DR] ← R[SB]

Control States and Register Transfers for Instruction USR (ASM Chart)

Paste ASM chart.

This state table fully defines the three-cycle execution of the USR instruction. In EX0, the R[SA] - R[SB] subtraction is performed FDIFF, enabling the HR register write wen_out=1 to capture the N flag. State USR1 checks the N flag: if N=0, R[SA] is written to R[DR] (rw_out}=1), and the PC is incremented. If N=1, the FSM transitions to USR2 to write R[SB] to R[DR] instead, completing the MAX operation.

7.1.4. USR State Table

INPUTS		OUTPUTS										
s_r	nz_in	ns	ps_out	il_out	rw_out	rs_out[11:8]	rs_out[7:4]	rs_out[3:0]	mx_out	fs_out	wen_out	iom_out
EX0	0X/1X	USR11	EXO	0	0	DR	SA	SB	xxxx	FUSR	0	0
USR1	xx	INF	USR1	0	1	DR	SA	SB	0100	FUSR	1	0
INF	XX	EX0	INF	0	0	XXXX	XXXX	XXXX	0000	0	1	0

7.2. Verification Plan for USR1 Instruction Decoding

INPUTS			EXPECTED OUTPUTS											cu
Cycle	ins_in[8:0]	nz_in	ns	ps_out	il_out	rw_out	rs_out[11:8]	rs_out[7:4]	rs_out[3:0]	mx_out	fs_out	wen_out	iom_out	st_r
0	USR	00	EX0	00	1	0	XXXX	XXXX	XXXX	XXX1	XXXX	1	0	INF
1	USR	XX	USR1	EX0	0	1	DR	SA	SB	0100	FUSR	1	0	INF
2	---	XX	EX0	INF	0	0	XXXX	XXXX	XXXX	XXXX	XXXX	0	0	EX0
3														
4														
5														
6														
7														
8														
9														
10														
11														
12														

7.3. RTL Code Creation and Verification

7.3.1. cu RTL Code Creation Results (after USR decoder design)

Describe Check the Results
Comment on Lint, AutoCheck and synthesis results. Lint Check Report Questa Lint Version 2022.4_1 5450673 linux_x86_64 22-Nov-2022 Timestamp : Mon Oct 27 20:00:41 2025 Description : Report for referring checks count, check violations details, and design information Design : cu Database : /homedir01/amiah25/DT2_2025/project/workdir/output/qlint/lint.db Design Quality Score : 100%

Sections:
  Section 1 : Check Summary
  Section 2 : Check Details
  Section 3 : Design Information

AutoCheck Compile Summary

Check	Evaluations	Found	Waived
ASSIGN_IMPLICIT_CONSTANT	3	0	0
CASE_DUPLICATE	2	0	0
CLK_DELAY	1	0	0
CLK_IN_DATA	4	0	0
COMBO_LOOP	14	0	0
DECLARATION_UNDRIVEN	5	0	0
DECLARATION_UNUSED	5	0	0
DECLARATION_UNUSED_UNDRIVEN	0	0	0
FUNCTION_INCOMPLETE_ASSIGN	0	0	0
INDEX_UNREACHABLE	0	0	0
LATCH_INFERRED	9	0	0
LOGIC_UNDRIVEN	5	0	0
LOGIC_UNUSED	0	0	0
PORT_UNDRIVEN	12	0	0
PORT_UNUSED	12	0	0
REG_MIXED_ASSIGNS	10	0	0
REG_NO_RESET	1	0	0
REG_RACE	0	0	0
REG_VARIABLE_ARESET	2	0	0
RESET_HIGH_LOW	1	0	0
RESET_SYNC_ASYNC	1	0	0
SLIST_INCOMPLETE	9	0	0
X_ASSIGN_REACHABLE	0	0	0
AC Total	96	0	0

WRITE OUT RESULTS
#
##
##
report_timing
Information: Updating design information... (UID-85)

****
Report : timing
        -path full
        -delay max
        -max_paths 1
Design : cu
Version: T-2022.03-SP5-1
Date   : Mon Oct 27 20:04:00 2025
****

Operating Conditions: tt1p05v25c   Library: saed32hvt_tt1p05v25c

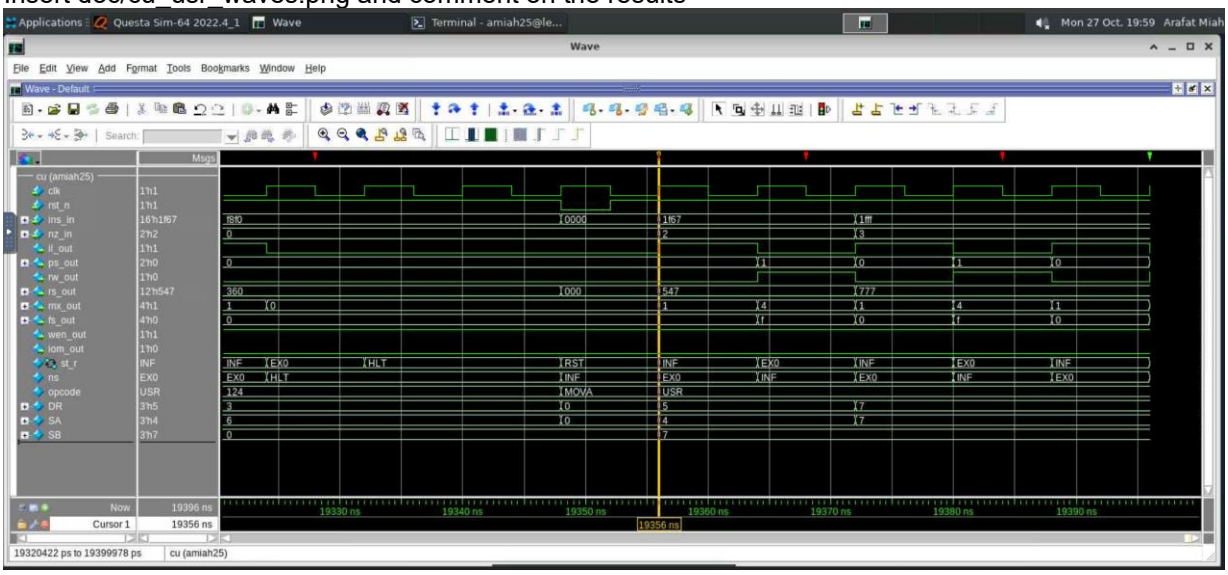
```
Wire Load Model Mode: enclosed

  Startpoint: st_r_reg[1]
              (rising edge-triggered flip-flop clocked by clk)
  Endpoint: st_r_reg[0]
            (rising edge-triggered flip-flop clocked by clk)
  Path Group: clk
  Path Type: max

  Des/Clust/Port     Wire Load Model       Library
  ------------------------------------------------
  cu                 ForQA                 saed32hvt_tt1p05v25c

  Point                                 Incr       Path
  -----------------------------------------------------------
  clock clk (rise edge)                 0.00       0.00
  clock network delay (ideal)           0.00       0.00
  st_r_reg[1]/CLK (DFFARX1_HVT)         0.00       0.00 r
  st_r_reg[1]/Q (DFFARX1_HVT)           0.19       0.19 r
  U149/Y (OA221X1_HVT)                  0.15       0.34 r
  st_r_reg[0]/D (DFFARX1_HVT)           0.01       0.35 r
  data arrival time                                0.35

  clock clk (rise edge)                 4.00       4.00
  clock network delay (ideal)           0.00       4.00
  st_r_reg[0]/CLK (DFFARX1_HVT)         0.00       4.00 r
  library setup time                   -0.08       3.92
  data required time                               3.92
  -----------------------------------------------------------
  data required time                               3.92
  data arrival time                               -0.35
  -----------------------------------------------------------
  slack (MET)                                      3.57


1
report_area -hierarchy > reports/${DESIGN_NAME}.gatelevel.area.txt
report_timing > reports/${DESIGN_NAME}.gatelevel.timing.txt
report_qor > reports/${DESIGN_NAME}.gatelevel.qor.txt
read_saif -auto_map_names -input results/${DESIGN_NAME}.saif -instance_name
${DESIGN_NAME}_tb/DUT -verbose
1
report_power > reports/${DESIGN_NAME}.gatelevel.power.txt
if { ${DESIGN_NAME} == "mycpu" } { report_power -hierarchy >>
reports/${DESIGN_NAME}.gatelevel.power.txt }
report_reference  -hierarchy > reports/${DESIGN_NAME}.gatelevel.reference.txt
change_names -rules verilog -hierarchy
1
write -hierarchy -format verilog -output results/${DESIGN_NAME}_gatelevel.v
Writing verilog file '/homedir01/amiah25/DT2_2025/project/workdir/results/cu_gatelevel.v'.
1
write_sdf -version 2.1 results/${DESIGN_NAME}_gatelevel.sdf
Information: Annotated 'cell' delays are assumed to include load delay. (UID-282)
Information: Writing timing information to file
'/homedir01/amiah25/DT2_2025/project/workdir/results/cu_gatelevel.sdf'. (WT-3)
```

```
1
reset_timing_derate
1
write_sdc -version 1.7 results/${DESIGN_NAME}_gatelevel.sdc
1
saif_map -write_map results/${DESIGN_NAME}_rtl.saifmap
Information: Writing SAIF name mapping information to file 'results/cu_rtl.saifmap'. (PWR-635)
1
gui_start
Current design is 'cu'.
4.1.1
gui_create_schematic
Current design is 'cu'.
Schematic.1
print_message_info

Id            Severity        Limit    Occurrences   Suppressed
-------------------------------------------------------------------
CMD-005          Error            0           12          12
CMD-025          Error            0            1           0
LINT-99      Information          0            1           0
OPT-1208     Information          0            1           0
OPT-775      Information          0            1           0
OPT-780      Information          0            1           0
PWR-12         Warning            0            1           0
PWR-24       Information          0            1           0
PWR-415        Warning            0            1           0
PWR-536      Information          0            6           0
PWR-6        Information          0            2           0
PWR-602      Information          0            1           0
PWR-635      Information          0            1           0
PWR-850      Information          0            1           0
UCN-1          Warning            0           10          10
UID-282      Information          0            1           0
UID-85       Information          0            2           0
UISN-27      Information          0            1           0
UPF-213        Warning           10            0           0
UPF-213a       Warning           10            0           0
UPF-213b       Warning           10            0           0
UPF-213c       Warning           10            0           0
UPF-213d       Warning           10            0           0
UPF-213e       Warning           10            0           0
UPF-213f       Warning           10            0           0
VER-504        Warning            0            1           0
VO-4           Warning            0            1           1
WT-3         Information          0            1           0

Diagnostics summary: 1 error, 3 warnings, 21 informationals
```

7.3.2. USR Instruction Simulation Results

Simulation Waveform

Insert doc/cu_usr_waves.png and comment on the results



Code coverage after USR decoder design

Include code coverage table for === Instance: /cu_tb/DUT from file reports/cu.rtl.simulation_coverage.txt and comment the results.
Recursive Coverage Report Summary Data by instance

```
================================================================================
===
=== Instance: /cu_tb/DUT
=== Design Unit: work.cu
================================================================================
===
    Enabled Coverage              Bins      Hits    Misses  Coverage
    ----------------              ----      ----    ------  --------
    Assertions                      20        20         0   100.00%
    Branches                        37        37         0   100.00%
    Directives                       7         7         0   100.00%
    FSM States                       4         4         0   100.00%
    FSM Transitions                  7         5         2    71.42%
    Statements                      96        96         0   100.00%
    Toggles                        268       195        73    72.76%


Total Coverage By Instance (filtered view): 92.02%
```

SELF ASSESSMENT

Write a summary of the week's work and results, including

1. Assessment of results
    a. Did you manage to specify the RTL logic for the USR instruction?
       Yes
    b. Does the code compile and elaborate without errors and warnings?
       Yes
    c. Do the RTL modules function, pass checks, and synthesize correctly
       Yes
2. Approximate time spent on the project this week
   12-14 hours
3. Open issues you need help with
   The project was pretty tough, I need to take advice from AI about what's going on here and
4. Problems found in earlier weeks' code fixed this week
   I did not find any problem, but I forgot to mention that I did some group study with Abid Ahmed (2411104), about understanding the circuit and code, then running the code simultaneously. Even though he is an undergrad student, I must say that he has pretty good knowledge about understanding the instructions far better than me.

WEEK8: FUNCTIONAL VERIFICATION AND IMPLEMENTATION (mycpu)

LEARNING GOALS
1. Learn principles and practices of functional verification of complex RTL designs. 2. Learn the principles of logic optimization 3. Learn to analyze logic synthesis results from EDA tool reports.

8.1. mycpu Code Check results

Describe Check the Results

Comment on Connectivity, Lint and AutoCheck results.

```
---------------------------------------------------------------------------
Questa PropCheck Version 2022.4_1 5450673 linux_x86_64 22-Nov-2022
---------------------------------------------------------------------------
Report Generated              : Mon Nov  3 22:42:15 2025
---------------------------------------------------------------------------

Executing Command : formal verify -exclude_cover -group connectcheck
---------------------------------------------------------------------------

Clock Relationships
---------------------------------------------------------------------------
\$global_clock          : -


Using user-specified initialization sequence:

---------------- BEGIN RESET SEQUENCE --------------
$default_input_value 0
rst_n = 0
##1
rst_n = 1

---------------- END RESET SEQUENCE ----------------


Port Constraints
---------------------------------------------------------------------------
Category          Value       Directive   Port
---------------------------------------------------------------------------
<empty>
---------------------------------------------------------------------------



---------------------------------------------------------------------------
Assumptions (1)
---------------------------------------------------------------------------
```

qcon_stable_enable_cutpoint_concatAll (netlist property)

Active Targets (49)

LINE_10_CHECK_connect (netlist property)
LINE_11_CHECK_connect (netlist property)
LINE_12_CHECK_connect (netlist property)
LINE_13_CHECK_connect (netlist property)
LINE_14_CHECK_connect (netlist property)
LINE_15_CHECK_connect (netlist property)
LINE_16_CHECK_connect (netlist property)
LINE_17_CHECK_connect (netlist property)
LINE_18_CHECK_connect (netlist property)
LINE_19_CHECK_connect (netlist property)
LINE_20_CHECK_connect (netlist property)
LINE_21_CHECK_connect (netlist property)
LINE_22_CHECK_connect (netlist property)
LINE_23_CHECK_connect (netlist property)
LINE_24_CHECK_connect (netlist property)
LINE_25_CHECK_connect (netlist property)
LINE_26_CHECK_connect (netlist property)
LINE_27_CHECK_connect (netlist property)
LINE_28_CHECK_connect (netlist property)
LINE_29_CHECK_connect (netlist property)
LINE_2_CHECK_connect (netlist property)
LINE_30_CHECK_connect (netlist property)
LINE_31_CHECK_connect (netlist property)
LINE_32_CHECK_connect (netlist property)
LINE_33_CHECK_connect (netlist property)
LINE_34_CHECK_connect (netlist property)
LINE_35_CHECK_connect (netlist property)
LINE_36_CHECK_connect (netlist property)
LINE_37_CHECK_connect (netlist property)
LINE_38_CHECK_connect (netlist property)
LINE_39_CHECK_connect (netlist property)
LINE_3_CHECK_connect (netlist property)
LINE_40_CHECK_connect (netlist property)
LINE_41_CHECK_connect (netlist property)
LINE_42_CHECK_connect (netlist property)
LINE_43_CHECK_connect (netlist property)
LINE_44_CHECK_connect (netlist property)
LINE_45_CHECK_connect (netlist property)
LINE_46_CHECK_connect (netlist property)
LINE_47_CHECK_connect (netlist property)
LINE_48_CHECK_connect (netlist property)
LINE_49_CHECK_connect (netlist property)
LINE_4_CHECK_connect (netlist property)
LINE_50_CHECK_connect (netlist property)
LINE_5_CHECK_connect (netlist property)
LINE_6_CHECK_connect (netlist property)
LINE_7_CHECK_connect (netlist property)
LINE_8_CHECK_connect (netlist property)

LINE_9_CHECK_connect (netlist property)

X Registers at Completion of Initialization Sequence
---
  20 registers (100.0% of 20 in sequential fanin of properties)

  CU.nz_r (File /homedir01/amiah25/DT2_2025/project/workdir/input/cu.sv, Line 29) 2'bxx
  CU.st_r (File /homedir01/amiah25/DT2_2025/project/workdir/input/cu.sv, Line 26) 6'bxxxxxx
  IR.ir_r (File /homedir01/amiah25/DT2_2025/project/workdir/input/ir.sv, Line 16) 16'bxxxxxxxxxxxxxxxx
  PC.pc_r (File /homedir01/amiah25/DT2_2025/project/workdir/input/pc.sv, Line 15)
16'bxxxxxxxxxxxxxxxx
  RB.hb_r[0] (File /homedir01/amiah25/DT2_2025/project/workdir/input/rb.sv, Line 16)
16'bxxxxxxxxxxxxxxxx
  RB.hb_r[1] (File /homedir01/amiah25/DT2_2025/project/workdir/input/rb.sv, Line 16)
16'bxxxxxxxxxxxxxxxx
  RB.hb_r[2] (File /homedir01/amiah25/DT2_2025/project/workdir/input/rb.sv, Line 16)
16'bxxxxxxxxxxxxxxxx
  RB.hb_r[3] (File /homedir01/amiah25/DT2_2025/project/workdir/input/rb.sv, Line 16)
16'bxxxxxxxxxxxxxxxx
  RB.hb_r[4] (File /homedir01/amiah25/DT2_2025/project/workdir/input/rb.sv, Line 16)
16'bxxxxxxxxxxxxxxxx
  RB.hb_r[5] (File /homedir01/amiah25/DT2_2025/project/workdir/input/rb.sv, Line 16)
16'bxxxxxxxxxxxxxxxx
  RB.hb_r[6] (File /homedir01/amiah25/DT2_2025/project/workdir/input/rb.sv, Line 16)
16'bxxxxxxxxxxxxxxxx
  RB.hb_r[7] (File /homedir01/amiah25/DT2_2025/project/workdir/input/rb.sv, Line 16)
16'bxxxxxxxxxxxxxxxx
  RB.rb_r[0] (File /homedir01/amiah25/DT2_2025/project/workdir/input/rb.sv, Line 15)
16'bxxxxxxxxxxxxxxxx
  RB.rb_r[1] (File /homedir01/amiah25/DT2_2025/project/workdir/input/rb.sv, Line 15)
16'bxxxxxxxxxxxxxxxx
  RB.rb_r[2] (File /homedir01/amiah25/DT2_2025/project/workdir/input/rb.sv, Line 15)
16'bxxxxxxxxxxxxxxxx
  RB.rb_r[3] (File /homedir01/amiah25/DT2_2025/project/workdir/input/rb.sv, Line 15)
16'bxxxxxxxxxxxxxxxx
  RB.rb_r[4] (File /homedir01/amiah25/DT2_2025/project/workdir/input/rb.sv, Line 15)
16'bxxxxxxxxxxxxxxxx
  RB.rb_r[5] (File /homedir01/amiah25/DT2_2025/project/workdir/input/rb.sv, Line 15)
16'bxxxxxxxxxxxxxxxx
  RB.rb_r[6] (File /homedir01/amiah25/DT2_2025/project/workdir/input/rb.sv, Line 15)
16'bxxxxxxxxxxxxxxxx
  RB.rb_r[7] (File /homedir01/amiah25/DT2_2025/project/workdir/input/rb.sv, Line 15)
16'bxxxxxxxxxxxxxxxx

Z Registers at Completion of Initialization Sequence
---
  0 registers (0.0% of 20 in sequential fanin of properties)

```
---------------------------------------
Formal Netlist Statistics         Count
---------------------------------------
Control Point Bits                  334
  DUT Input Bits                     34
  Cut Point Bits                    186
  Black Box Output Bits               0
  Undriven Wire Bits                 50
  Modeling Bits                      64
State Bits                          310
  Counter State Bits                  0
  RAM State Bits                      0
  Register State Bits               272
  Property State Bits                38
Logic Gates                        3695
  Design Gates                     2936
  Property Gates                    759
---------------------------------------


--------------------------------------------------------------------------------
Targets Proven (49)
--------------------------------------------------------------------------------
LINE_10_CHECK_connect (netlist property)
LINE_11_CHECK_connect (netlist property)
LINE_12_CHECK_connect (netlist property)
LINE_13_CHECK_connect (netlist property)
LINE_14_CHECK_connect (netlist property)
LINE_15_CHECK_connect (netlist property)
LINE_16_CHECK_connect (netlist property)
LINE_17_CHECK_connect (netlist property)
LINE_18_CHECK_connect (netlist property)
LINE_19_CHECK_connect (netlist property)
LINE_20_CHECK_connect (netlist property)
LINE_21_CHECK_connect (netlist property)
LINE_22_CHECK_connect (netlist property)
LINE_23_CHECK_connect (netlist property)
LINE_24_CHECK_connect (netlist property)
LINE_25_CHECK_connect (netlist property)
LINE_26_CHECK_connect (netlist property)
LINE_27_CHECK_connect (netlist property)
LINE_28_CHECK_connect (netlist property)
LINE_29_CHECK_connect (netlist property)
LINE_2_CHECK_connect (netlist property)
LINE_30_CHECK_connect (netlist property)
LINE_31_CHECK_connect (netlist property)
LINE_32_CHECK_connect (netlist property)
LINE_33_CHECK_connect (netlist property)
LINE_34_CHECK_connect (netlist property)
LINE_35_CHECK_connect (netlist property)
LINE_36_CHECK_connect (netlist property)
LINE_37_CHECK_connect (netlist property)
LINE_38_CHECK_connect (netlist property)
LINE_39_CHECK_connect (netlist property)
```

LINE_3_CHECK_connect (netlist property)
LINE_40_CHECK_connect (netlist property)
LINE_41_CHECK_connect (netlist property)
LINE_42_CHECK_connect (netlist property)
LINE_43_CHECK_connect (netlist property)
LINE_44_CHECK_connect (netlist property)
LINE_45_CHECK_connect (netlist property)
LINE_46_CHECK_connect (netlist property)
LINE_47_CHECK_connect (netlist property)
LINE_48_CHECK_connect (netlist property)
LINE_49_CHECK_connect (netlist property)
LINE_4_CHECK_connect (netlist property)
LINE_50_CHECK_connect (netlist property)
LINE_5_CHECK_connect (netlist property)
LINE_6_CHECK_connect (netlist property)
LINE_7_CHECK_connect (netlist property)
LINE_8_CHECK_connect (netlist property)
LINE_9_CHECK_connect (netlist property)

Assumptions Used in Proofs

<no assumptions used in proofs>

Assumptions Used in Bounded Proofs

<no assumptions used in bounded proofs>

Assumptions Used in Unsatisfiable Sanity Checks

<no assumptions used in unsatisfiable sanity checks>

Assumptions Used in Bounded Unsatisfiable Sanity Checks

<no assumptions used in bounded unsatisfiable sanity checks>

Target Waveforms (0)

  TB Time(ns) Dist Target

<no target firings>

```
------------------------------------------------------------------------------


---------------------------------------
Target Waveforms Summary by Distance
---------------------------------------
Distance                        Count
---------------------------------------
<no target waveforms>
---------------------------------------
Total                            0
---------------------------------------


------------------------------------------------------------------------------
Proof Radius Summary by Target
------------------------------------------------------------------------------
<all targets either proven or fired>
------------------------------------------------------------------------------


------------------------------------------------------------------------------
Engine Performance
---------- Proofs -------------------------- Waveforms ------------
  Engine |  Safety Liveness Vacuity  Sanity |  Safety Liveness Vacuity  Sanity
------------------------------------------------------------------------------
       0 |     31       0      0      0 |      0       0      0      0
       7 |     18       0      0      0 |      0       0     22      0
      10 |      0       0      0      0 |      0       0     27      0
------------------------------------------------------------------------------


--------- Process Statistics ----------
Elapsed Time                     0 s
Total CPU Time                   1 s
Total Peak Memory              1.2 GB
---------- Engine Processes -----------
Average CPU Time                 0 s
Minimum CPU Utilization        0.0 %
Average Peak Memory            0.2 GB
Maximum Peak Memory            0.3 GB
Peak Cores                       4
Distinct Machines                1
Launch Failures                  0
Unexpected Process Terminations  0
Successful Re-connections        0
---------------------------------------


----- Detailed Process Statistics -----
Elapsed Time                     0 s
-------- Orchestration Process --------
----- lehmus-cn3.oulu.fi:3753355 ------
```

```
CPU Time                      0 s
Peak Memory                   0.3 GB
---------- Engine Processes ----------
----- lehmus-cn3.oulu.fi:3753462 ------
CPU Time                      0 s
Peak Memory                   0.2 GB
CPU Utilization               0 %
----- lehmus-cn3.oulu.fi:3753461 ------
CPU Time                      1 s
Peak Memory                   0.3 GB
CPU Utilization               0 %
----- lehmus-cn3.oulu.fi:3753467 ------
CPU Time                      0 s
Peak Memory                   0.2 GB
CPU Utilization               0 %
----- lehmus-cn3.oulu.fi:3753469 ------
CPU Time                      0 s
Peak Memory                   0.2 GB
CPU Utilization               0 %
----------------------------------------


----------------------------------------
Property Summary                Count
----------------------------------------
Assumed                           1
  netlist property                1
Proven                           49
Inconclusive                      0
Fired                             0
----------------------------------------
Total                            50
```

AutoCheck Compile Summary

Check	Evaluations	Found	Waived
ASSIGN_IMPLICIT_CONSTANT	35	0	0
CASE_DUPLICATE	4	0	0
CLK_DELAY	6	0	0
CLK_IN_DATA	80	0	0
COMBO_LOOP	55	0	0
DECLARATION_UNDRIVEN	30	3	0
DECLARATION_UNUSED	30	1	0
DECLARATION_UNUSED_UNDRIVEN	0	0	0
FUNCTION_INCOMPLETE_ASSIGN	0	0	0
INDEX_UNREACHABLE	256	0	0
LATCH_INFERRED	17	0	0
LOGIC_UNDRIVEN	30	0	0
LOGIC_UNUSED	16	0	0
PORT_UNDRIVEN	54	0	0
PORT_UNUSED	54	0	0
REG_MIXED_ASSIGNS	37	0	0
REG_NO_RESET	20	0	0

```
REG_RACE                        0          0          0
REG_VARIABLE_ARESET            40          0          0
RESET_HIGH_LOW                  1          0          0
RESET_SYNC_ASYNC                1          0          0
SLIST_INCOMPLETE               17          0          0
X_ASSIGN_REACHABLE              0          0          0
--------------------------------------------------------------------------------
AC Total                      783          4          0
--------------------------------------------------------------------------------


AutoCheck Verify Summary
--------------------------------------------------------------------------------
Check                  Evaluations        Found       Waived
--------------------------------------------------------------------------------
ARITH_OVERFLOW_SUB              0          0          0
ARITH_OVERFLOW_VAL              7          2          0
ARITH_ZERO_DIV                  0          0          0
ARITH_ZERO_MOD                  0          0          0
BLOCK_UNREACHABLE              71          1          0
BUS_MULTIPLY_DRIVEN             0          0          0
BUS_UNDRIVEN                    0          0          0
BUS_VALUE_CONFLICT              0          0          0
CASE_DEFAULT                    3          1          0
CASE_FULL                       1          0          0
CASE_PARALLEL                   2          0          0
FSM_DEADLOCK_STATE              4          1          0
FSM_LOCKOUT_STATE               4          0          0
FSM_STUCK_BIT                   6          3          0
FSM_UNREACHABLE_STATE           4          0          0
FSM_UNREACHABLE_TRANS           5          0          0
INDEX_ILLEGAL                   7          0          0
INIT_X_OPTIMISM                 6          0          0
INIT_X_PESSIMISM                0          0          0
INIT_X_UNRESOLVED               0          0          0
INIT_X_UNRESOLVED_MEM           0          0          0
ONE_COLD                        0          0          0
ONE_HOT                         0          0          0
REG_MULTIPLY_DRIVEN             0          0          0
REG_STUCK_AT                   19          0          0
REG_TOGGLE_VIOLATION            0          0          0
--------------------------------------------------------------------------------
AC Total                      139          8          0
--------------------------------------------------------------------------------


------------------------------------------------------------
AutoCheck Details
------------------------------------------------------------

Type      : ARITH_OVERFLOW_VAL
Severity  : Caution
Status    : Uninspected
Module    : pc
Name      : pc_r
```

Instance : PC
Location : /homedir01/amiah25/DT2_2025/project/workdir/input/pc.sv:33
Assignment Width: 16
Waveform Distance: 7

Type   : ARITH_OVERFLOW_VAL
Severity : Caution
Status   : Uninspected
Module   : pc
Name   : pc_r
Instance : PC
Location : /homedir01/amiah25/DT2_2025/project/workdir/input/pc.sv:34
Assignment Width: 16
Waveform Distance: 7

Type   : BLOCK_UNREACHABLE
Severity : Caution
Status   : Uninspected
Module   : muxd
Location : /homedir01/amiah25/DT2_2025/project/workdir/input/muxd.sv:15

Type   : CASE_DEFAULT
Severity : Violation
Status   : Uninspected
Module   : cu
Case Control: opcode
Instance : CU
Location : /homedir01/amiah25/DT2_2025/project/workdir/input/cu.sv:150
Waveform Distance: 3

Type   : DECLARATION_UNDRIVEN
Severity : Violation
Status   : Uninspected
Module   : rb
Name   : a_addr
Instance : RB

Type   : DECLARATION_UNDRIVEN
Severity : Violation
Status   : Uninspected
Module   : rb
Name   : b_addr
Instance : RB

Type   : DECLARATION_UNDRIVEN
Severity : Violation
Status   : Uninspected
Module   : rb
Name   : w_addr
Instance : RB

Type   : DECLARATION_UNUSED
Severity : Caution
Status   : Uninspected
Module   : mycpu

```
Name      : rs

Type      : FSM_DEADLOCK_STATE
Severity  : Caution
Status    : Uninspected
Module    : cu
Instance  : CU
FSM       : st_r
Deadend State: HLT
Location  : /homedir01/amiah25/DT2_2025/project/workdir/input/cu.sv:161
Waveform Distance: 4

Type      : FSM_STUCK_BIT
Severity  : Caution
Status    : Uninspected
Module    : cu
Instance  : CU
FSM       : st_r[3]
Value     : 1'b0
Location  : /homedir01/amiah25/DT2_2025/project/workdir/input/cu.sv:66

Type      : FSM_STUCK_BIT
Severity  : Caution
Status    : Uninspected
Module    : cu
Instance  : CU
FSM       : st_r[4]
Value     : 1'b0
Location  : /homedir01/amiah25/DT2_2025/project/workdir/input/cu.sv:66

Type      : FSM_STUCK_BIT
Severity  : Caution
Status    : Uninspected
Module    : cu
Instance  : CU
FSM       : st_r[5]
Value     : 1'b0
Location  : /homedir01/amiah25/DT2_2025/project/workdir/input/cu.sv:66

-------------------------------------------------------------


-------------------------------------------------------------------------------
CPU Time per Check Type
-------------------------------------------------------------------------------
 %   Seconds    Check
-------------------------------------------------------------------------------
 0.0%      0    ARITH_OVERFLOW_SUB
 0.0%      0    ARITH_OVERFLOW_VAL
 0.0%      0    ARITH_ZERO_DIV
 0.0%      0    ARITH_ZERO_MOD
 0.2%      0    BLOCK_UNREACHABLE
 0.0%      0    BUS_MULTIPLY_DRIVEN
 0.0%      0    BUS_UNDRIVEN
 0.0%      0    BUS_VALUE_CONFLICT
 0.0%      0    CASE_DEFAULT
```

```
 0.0%     0   CASE_FULL
 0.0%     0   CASE_PARALLEL
 0.0%     0   FSM_DEADLOCK_STATE
99.0%     0   FSM_LOCKOUT_STATE
 0.0%     0   FSM_STUCK_BIT
 0.0%     0   FSM_UNREACHABLE_STATE
 0.0%     0   FSM_UNREACHABLE_TRANS
 0.0%     0   INDEX_ILLEGAL
 0.0%     0   INIT_X_OPTIMISM
 0.0%     0   INIT_X_PESSIMISM
 0.0%     0   INIT_X_UNRESOLVED
 0.0%     0   INIT_X_UNRESOLVED_MEM
 0.0%     0   ONE_COLD
 0.0%     0   ONE_HOT
 0.0%     0   REG_MULTIPLY_DRIVEN
 0.7%     0   REG_STUCK_AT
 0.0%     0   REG_TOGGLE_VIOLATION
--------------------------------------------------------------------------------

--------------------------------------------------------------------------------
Unused autocheck_report Directives
--------------------------------------------------------------------------------
<none>
--------------------------------------------------------------------------------



----- Detailed Process Statistics -----
Elapsed Time                    0 s
-------- Orchestration Process --------
----- lehmus-cn3.oulu.fi:3753667 ------
CPU Time                        0 s
Peak Memory                     0.3 GB
---------- Engine Processes ----------
----- lehmus-cn3.oulu.fi:3753696 ------
CPU Time                        0 s
Peak Memory                     0.2 GB
CPU Utilization                 0 %
----- lehmus-cn3.oulu.fi:3753698 ------
CPU Time                        0 s
Peak Memory                     0.2 GB
CPU Utilization                 0 %
----- lehmus-cn3.oulu.fi:3753697 ------
CPU Time                        1 s
Peak Memory                     0.3 GB
CPU Utilization                 0 %
----- lehmus-cn3.oulu.fi:3753699 ------
CPU Time                        0 s
Peak Memory                     0.2 GB
CPU Utilization                 0 %
```

```
AutoCheck Compile Summary
--------------------------------------------------------------------------------
```

Check	Evaluations	Found	Waived
ASSIGN_IMPLICIT_CONSTANT	35	0	0
CASE_DUPLICATE	4	0	0
CLK_DELAY	6	0	0
CLK_IN_DATA	80	0	0
COMBO_LOOP	55	0	0
DECLARATION_UNDRIVEN	30	3	0
DECLARATION_UNUSED	30	1	0
DECLARATION_UNUSED_UNDRIVEN	0	0	0
FUNCTION_INCOMPLETE_ASSIGN	0	0	0
INDEX_UNREACHABLE	256	0	0
LATCH_INFERRED	17	0	0
LOGIC_UNDRIVEN	30	0	0
LOGIC_UNUSED	16	0	0
PORT_UNDRIVEN	54	0	0
PORT_UNUSED	54	0	0
REG_MIXED_ASSIGNS	37	0	0
REG_NO_RESET	20	0	0
REG_RACE	0	0	0
REG_VARIABLE_ARESET	40	0	0
RESET_HIGH_LOW	1	0	0
RESET_SYNC_ASYNC	1	0	0
SLIST_INCOMPLETE	17	0	0
X_ASSIGN_REACHABLE	0	0	0
AC Total	783	4	0

AutoCheck Verify Summary

Check	Evaluations	Found	Waived
ARITH_OVERFLOW_SUB	0	0	0
ARITH_OVERFLOW_VAL	7	2	0
ARITH_ZERO_DIV	0	0	0
ARITH_ZERO_MOD	0	0	0
BLOCK_UNREACHABLE	71	1	0
BUS_MULTIPLY_DRIVEN	0	0	0
BUS_UNDRIVEN	0	0	0
BUS_VALUE_CONFLICT	0	0	0
CASE_DEFAULT	3	1	0
CASE_FULL	1	0	0
CASE_PARALLEL	2	0	0
FSM_DEADLOCK_STATE	4	1	0
FSM_LOCKOUT_STATE	4	0	0
FSM_STUCK_BIT	6	3	0
FSM_UNREACHABLE_STATE	4	0	0
FSM_UNREACHABLE_TRANS	5	0	0
INDEX_ILLEGAL	7	0	0
INIT_X_OPTIMISM	6	0	0
INIT_X_PESSIMISM	0	0	0
INIT_X_UNRESOLVED	0	0	0
INIT_X_UNRESOLVED_MEM	0	0	0

```
ONE_COLD                      0          0          0
ONE_HOT                       0          0          0
REG_MULTIPLY_DRIVEN           0          0          0
REG_STUCK_AT                 19          0          0
REG_TOGGLE_VIOLATION          0          0          0
-------------------------------------------------------------------------------
AC Total                    139          8          0
-------------------------------------------------------------------------------


------------------------------------------------------------
AutoCheck Details
------------------------------------------------------------

Type      : ARITH_OVERFLOW_VAL
Severity  : Caution
Status    : Uninspected
Module    : pc
Name      : pc_r
Instance  : PC
Location  : /homedir01/amiah25/DT2_2025/project/workdir/input/pc.sv:33
Assignment Width: 16
Waveform Distance: 7

Type      : ARITH_OVERFLOW_VAL
Severity  : Caution
Status    : Uninspected
Module    : pc
Name      : pc_r
Instance  : PC
Location  : /homedir01/amiah25/DT2_2025/project/workdir/input/pc.sv:34
Assignment Width: 16
Waveform Distance: 7

Type      : BLOCK_UNREACHABLE
Severity  : Caution
Status    : Uninspected
Module    : muxd
Location  : /homedir01/amiah25/DT2_2025/project/workdir/input/muxd.sv:15

Type      : CASE_DEFAULT
Severity  : Violation
Status    : Uninspected
Module    : cu
Case Control: opcode
Instance  : CU
Location  : /homedir01/amiah25/DT2_2025/project/workdir/input/cu.sv:150
Waveform Distance: 3

Type      : DECLARATION_UNDRIVEN
Severity  : Violation
Status    : Uninspected
Module    : rb
Name      : a_addr
Instance  : RB
```

Type : DECLARATION_UNDRIVEN
Severity : Violation
Status : Uninspected
Module : rb
Name : b_addr
Instance : RB

Type : DECLARATION_UNDRIVEN
Severity : Violation
Status : Uninspected
Module : rb
Name : w_addr
Instance : RB

Type : DECLARATION_UNUSED
Severity : Caution
Status : Uninspected
Module : mycpu
Name : rs

Type : FSM_DEADLOCK_STATE
Severity : Caution
Status : Uninspected
Module : cu
Instance : CU
FSM : st_r
Deadend State: HLT
Location : /homedir01/amiah25/DT2_2025/project/workdir/input/cu.sv:161
Waveform Distance: 4

Type : FSM_STUCK_BIT
Severity : Caution
Status : Uninspected
Module : cu
Instance : CU
FSM : st_r[3]
Value : 1'b0
Location : /homedir01/amiah25/DT2_2025/project/workdir/input/cu.sv:66

Type : FSM_STUCK_BIT
Severity : Caution
Status : Uninspected
Module : cu
Instance : CU
FSM : st_r[4]
Value : 1'b0
Location : /homedir01/amiah25/DT2_2025/project/workdir/input/cu.sv:66

Type : FSM_STUCK_BIT
Severity : Caution
Status : Uninspected
Module : cu
Instance : CU
FSM : st_r[5]

```
Value     : 1'b0
Location  : /homedir01/amiah25/DT2_2025/project/workdir/input/cu.sv:66

------------------------------------------------------------


-------------------------------------------------------------------------------
CPU Time per Check Type
-------------------------------------------------------------------------------
 %   Seconds    Check
-------------------------------------------------------------------------------
 0.0%      0    ARITH_OVERFLOW_SUB
 0.0%      0    ARITH_OVERFLOW_VAL
 0.0%      0    ARITH_ZERO_DIV
 0.0%      0    ARITH_ZERO_MOD
 0.2%      0    BLOCK_UNREACHABLE
 0.0%      0    BUS_MULTIPLY_DRIVEN
 0.0%      0    BUS_UNDRIVEN
 0.0%      0    BUS_VALUE_CONFLICT
 0.0%      0    CASE_DEFAULT
 0.0%      0    CASE_FULL
 0.0%      0    CASE_PARALLEL
 0.0%      0    FSM_DEADLOCK_STATE
99.0%      0    FSM_LOCKOUT_STATE
 0.0%      0    FSM_STUCK_BIT
 0.0%      0    FSM_UNREACHABLE_STATE
 0.0%      0    FSM_UNREACHABLE_TRANS
 0.0%      0    INDEX_ILLEGAL
 0.0%      0    INIT_X_OPTIMISM
 0.0%      0    INIT_X_PESSIMISM
 0.0%      0    INIT_X_UNRESOLVED
 0.0%      0    INIT_X_UNRESOLVED_MEM
 0.0%      0    ONE_COLD
 0.0%      0    ONE_HOT
 0.0%      0    REG_MULTIPLY_DRIVEN
 0.7%      0    REG_STUCK_AT
 0.0%      0    REG_TOGGLE_VIOLATION
-------------------------------------------------------------------------------


-------------------------------------------------------------------------------
Unused autocheck_report Directives
-------------------------------------------------------------------------------
<none>
-------------------------------------------------------------------------------



----- Detailed Process Statistics -----
Elapsed Time                    0 s
-------- Orchestration Process --------
----- lehmus-cn3.oulu.fi:3753667 ------
CPU Time                        0 s
Peak Memory                     0.3 GB
---------- Engine Processes ----------
----- lehmus-cn3.oulu.fi:3753696 ------
CPU Time                        0 s
```

```
Peak Memory                0.2 GB
CPU Utilization            0 %
----- lehmus-cn3.oulu.fi:3753698 ------
CPU Time                   0 s
Peak Memory                0.2 GB
CPU Utilization            0 %
----- lehmus-cn3.oulu.fi:3753697 ------
CPU Time                   1 s
Peak Memory                0.3 GB
CPU Utilization            0 %
----- lehmus-cn3.oulu.fi:3753699 ------
CPU Time                   0 s
Peak Memory                0.2 GB
CPU Utilization            0 %
```

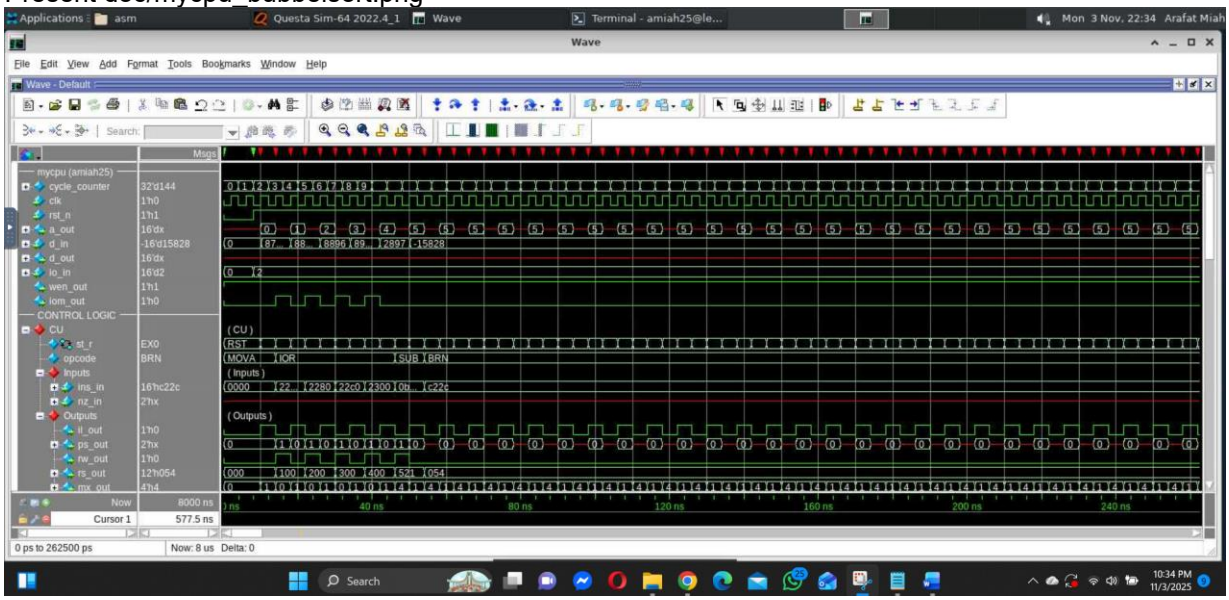
So, I can say that the code runs perfectly

8.2. mycpu RTL Simulation Results

8.2.1. Test Simulation

Simulation Waveforms from bubblesort test

Present doc/mycpu_bubbelsort.png



Comment on the results.

My waveform display confirms successful compilation and execution of the design, with the Control Unit (CU) cycling through states (RST, INF, EX0) to process instructions like MOVA, OR, and SUB_BRN. The original **COMB O_LOOP** violation,

which occurs when the function select f(s_out) depends directly on the zero/negative flags (z_in), is now resolved due to the nz_r register fix.

8.2.2. Instruction Set Simulation

Present Result of Instruction Set Test			
Instruction	**Initial Status**	**Final Status**	**Comments** **All the things were fine**
MOVA			
INC			
ADD			
MUL			
SRA			
SUB			
DEC			
SLA			
AND			
OR			
XOR			
NOT			
MOVB			
SHR			
SHL			
USR			
LDI			
ADI			
LD			
ST			
BRZ			

BRN	
JMP	
IOR	
IOW	

8.2.3. Debugging Exercise

Describe the procedure you used to debug the design and the result of the exercise.

Debug Procedure and Outcome
First I got some errors in mycpu , which I carefully modified it and then get the qlint,autocheck, and simulation result which are shown in the above

8.2.4. USR Instruction Simulation

Present the new code for USR.asm in case you modified it.

Program Code for USR.asm

```
IOR R1 R0
IOR R2 R0
IOR R3 R0
IOR R4 R0
IOR R5 R0
IOR R6 R0
IOR R7 R0
IOR R0 R0
USR R1 R2 R3
HAL
```

```
        ;; R0 assumed to be 0 after reset
IOR R1 R0
IOR R2 R0
IOR R3 R0
IOR R4 R0
;;
SUB R5 R2 R1
BRN R5 4
MOVA R6 R1
MOVA R1 R2
MOVA R2 R6
;;
SUB R5 R3 R2
BRN R5 4
MOVA R6 R2
MOVA R2 R3
MOVA R3 R6
;;
SUB R5 R4 R3
BRN R5 4
```

```
MOVA R6 R3
MOVA R3 R4
MOVA R4 R6
;;
SUB R5 R2 R1
BRN R5 4
MOVA R6 R1
MOVA R1 R2
MOVA R2 R6
;;
SUB R5 R3 R2
BRN R5 4
MOVA R6 R2
MOVA R2 R3
MOVA R3 R6
;;
SUB R5 R2 R1
BRN R5 4
MOVA R6 R1
MOVA R1 R2
MOVA R2 R6
;;
IOW R0 R1
IOW R0 R2
IOW R0 R3
IOW R0 R4
HAL
```

Simulation Waveforms from USR test

Present waveforms that show the execution of the USR instruction only. Referring to the images, explain what happens during the execution, and comment on the result.

The waveform shows the USR instruction is executed during the EX0 state between approx **72 ns and 96 ns** (cycles 19-21). During this execution, the control unit sets fs_out to FUSR (4'hF, or 15) and rw_out to 1'b1, directing the function unit to perform the user-defined operation and write the result back to the register file.

8.2.5 Full Instruction Set Simulation

Simulation Results

Comment on the results.

8.3. Logic Synthesis Result

8.3.1. Area Results

Timing Report

****
Report : area
Design : mycpu
Version: T-2022.03-SP5-1
Date : Mon Nov 3 23:31:45 2025
****

Library(s) Used:

saed32hvt_tt1p05v25c (File: /research/cas/public/DT2_2025/lib/logic_lib/saed32hvt_tt1p05v25c.db)

Number of ports: 482
Number of nets: 1305
Number of cells: 810
Number of combinational cells: 751
Number of sequential cells: 50
Number of macros/black boxes: 0
Number of buf/inv: 74
Number of references: 8

Combinational area: 2026.544251
Buf/Inv area: 98.353728
Noncombinational area: 364.188359
Macro/Black Box area: 0.000000
Net Interconnect area: 592.026475

```
Total cell area:          2390.732610
Total area:               2982.759085

Hierarchical area distribution
------------------------------

                          Global cell area        Local cell area
                          ----------------  ---------------------------
Hierarchical cell              Absolute   Percent  Combi-    Noncombi-  Black-
                          Total     Total    national  national   boxes   Design
------------------------------  ---------  -------  ---------  ---------  ------  ---------
mycpu                       2390.7326    100.0    0.0000    0.0000  0.0000  mycpu
CU                           186.2876     7.8   150.7074   35.5802  0.0000  cu
FU                          1491.8253    62.4  1491.8253    0.0000  0.0000  fu
IR                           126.8179     5.3    34.3094   92.5084  0.0000  ir
MUXB                          35.8343     1.5    35.8343    0.0000  0.0000  muxb
MUXD                          62.0111     2.6    62.0111    0.0000  0.0000  muxd_N3
MUXM                          43.7128     1.8    43.7128    0.0000  0.0000  muxm
PC                           278.0335    11.6   164.1770   113.8565  0.0000  pc
RB                           166.2102     7.0    43.9669   122.2433  0.0000  rb
------------------------------  ---------  -------  ---------  ---------  ------  ---------
Total                                    2026.5443   364.1884  0.0000
```

Parameter	RTL	Gate-Level
Flip-flop Count		
Total Cell Area	N/A	
NAND2 Equivalent Gate-Count	N/A	
Largest Submodule	N/A	

Comment on the results, e.g. flip-flop count.

8.3.2. Timing Results

```
Timing Report
****************************************
Report : timing
        -path full
        -delay max
        -max_paths 1
Design : mycpu
Version: T-2022.03-SP5-1
Date   : Mon Nov  3 23:29:57 2025
****************************************

Operating Conditions: tt1p05v25c   Library: saed32hvt_tt1p05v25c
Wire Load Model Mode: enclosed

  Startpoint: IR/ir_r_reg[15]
```

```
          (rising edge-triggered flip-flop clocked by clk)
Endpoint: CU/nz_r_reg[0]
          (rising edge-triggered flip-flop clocked by clk)
Path Group: clk
Path Type: max

Des/Clust/Port     Wire Load Model       Library
------------------------------------------------
mycpu              8000                  saed32hvt_tt1p05v25c
cu                 ForQA                 saed32hvt_tt1p05v25c
muxb               ForQA                 saed32hvt_tt1p05v25c
fu                 8000                  saed32hvt_tt1p05v25c

Point                                    Incr       Path
-------------------------------------------------------------
clock clk (rise edge)                    0.00       0.00
clock network delay (ideal)              0.00       0.00
IR/ir_r_reg[15]/CLK (DFFARX1_HVT)        0.00       0.00 r
IR/ir_r_reg[15]/Q (DFFARX1_HVT)          0.19       0.19 f
IR/ins_out[15] (ir)                      0.00       0.19 f
CU/ins_in[15] (cu)                       0.00       0.19 f
CU/U10/Y (OR3X1_HVT)                     0.12       0.31 f
CU/U11/Y (INVX0_HVT)                     0.07       0.39 r
CU/U17/Y (NAND4X0_HVT)                   0.15       0.54 f
CU/U55/Y (OA221X1_HVT)                   0.13       0.67 f
CU/U56/Y (AOI21X1_HVT)                   0.11       0.78 r
CU/mx_out[3] (cu)                        0.00       0.78 r
MUXB/sel_in (muxb)                       0.00       0.78 r
MUXB/U5/Y (INVX2_HVT)                    0.11       0.89 f
MUXB/U7/Y (AO22X1_HVT)                   0.16       1.05 f
MUXB/m_out[1] (muxb)                     0.00       1.05 f
FU/b_in[1] (fu)                          0.00       1.05 f
FU/U75/Y (INVX0_HVT)                     0.09       1.15 r
FU/U76/Y (INVX0_HVT)                     0.14       1.29 f
FU/U17/Y (NAND4X0_HVT)                   0.14       1.43 r
FU/U194/Y (OA221X1_HVT)                  0.13       1.56 r
FU/U284/CO (FADDX1_HVT)                  0.18       1.73 r
FU/U243/Y (INVX0_HVT)                    0.06       1.79 f
FU/U244/Y (AO222X1_HVT)                  0.10       1.90 f
FU/U264/CO (FADDX1_HVT)                  0.13       2.02 f
FU/U251/Y (NAND2X0_HVT)                  0.05       2.07 r
FU/U356/Y (NAND2X0_HVT)                  0.06       2.13 f
FU/U357/Y (NAND2X0_HVT)                  0.06       2.20 r
FU/intadd_12/U2/CO (FADDX1_HVT)          0.17       2.37 r
FU/U228/Y (OR2X1_HVT)                    0.08       2.45 r
FU/U358/Y (AO22X1_HVT)                   0.10       2.55 r
FU/intadd_7/U2/CO (FADDX1_HVT)           0.18       2.73 r
FU/U359/Y (AO222X1_HVT)                  0.19       2.92 r
FU/intadd_6/U2/CO (FADDX1_HVT)           0.19       3.10 r
FU/U360/Y (AO222X1_HVT)                  0.19       3.29 r
FU/intadd_5/U2/CO (FADDX1_HVT)           0.17       3.47 r
FU/U411/Y (AO222X1_HVT)                  0.17       3.63 r
FU/intadd_0/U3/CO (FADDX1_HVT)           0.16       3.79 r
FU/U71/Y (XOR2X1_HVT)                    0.15       3.94 f
FU/U117/S (FADDX1_HVT)                   0.18       4.12 r
```

```
FU/U119/Y (OA22X1_HVT)                  0.09       4.21 r
FU/U129/Y (NAND3X0_HVT)                 0.07       4.28 f
FU/U130/Y (AO21X1_HVT)                  0.08       4.35 f
FU/U457/Y (INVX0_HVT)                   0.04       4.40 r
FU/U462/Y (AND4X1_HVT)                  0.10       4.50 r
FU/nz_out[0] (fu)                       0.00       4.50 r
CU/nz_in[0] (cu)                        0.00       4.50 r
CU/U75/Y (AO22X1_HVT)                   0.11       4.61 r
CU/nz_r_reg[0]/D (DFFARX1_HVT)          0.01       4.62 r
data arrival time                                  4.62

clock clk (rise edge)                  10.00      10.00
clock network delay (ideal)             0.00      10.00
CU/nz_r_reg[0]/CLK (DFFARX1_HVT)        0.00      10.00 r
library setup time                     -0.07       9.93
data required time                                 9.93
-----------------------------------------------------------
data required time                                 9.93
data arrival time                                 -4.62
-----------------------------------------------------------
slack (MET)                                        5.31
```

Parameter	Value
Clock Period (final version)	10ns
Critical Path Length (Delay)	4.62ns
Slack	5.31ns
Comments on results.: The slack was positive ,,so the system runs perfectly	

Critical Path Report
Present data path section of timing report.
Describe the critical path's route in the design (start point, modules it travels through, endpoint), and which module contributes most to delay.

8.3.3. Power Results

```
Power Consumption Report:
Warning: There are 7 switching activity information conflicts. (PWR-19)
Information: Propagating switching activity (low effort zero delay simulation). (PWR-6)
Warning: The derived toggle rate value (0.200000) for the clock net 'clk' conflicts with the
annotated value (0.499875). Using the annotated value. (PWR-12)
Warning: Design has unannotated primary inputs. (PWR-414)
```

```
****************************************
Report : power
       -analysis_effort low
Design : mycpu
Version: T-2022.03-SP5-1
Date   : Mon Nov  3 23:31:45 2025
****************************************


Library(s) Used:

   saed32hvt_tt1p05v25c (File:
/research/cas/public/DT2_2025/lib/logic_lib/saed32hvt_tt1p05v25c.db)


Operating Conditions: tt1p05v25c   Library: saed32hvt_tt1p05v25c
Wire Load Model Mode: enclosed

Design       Wire Load Model          Library
------------------------------------------------
mycpu                8000            saed32hvt_tt1p05v25c
cu                  ForQA            saed32hvt_tt1p05v25c
ir                  ForQA            saed32hvt_tt1p05v25c
pc                   8000            saed32hvt_tt1p05v25c
fu                  8000             saed32hvt_tt1p05v25c
rb                  ForQA            saed32hvt_tt1p05v25c
muxm                 ForQA           saed32hvt_tt1p05v25c
muxb                ForQA            saed32hvt_tt1p05v25c
muxd_N3               ForQA          saed32hvt_tt1p05v25c


Global Operating Voltage = 1.05
Power-specific unit information :
   Voltage Units = 1V
   Capacitance Units = 1.000000ff
   Time Units = 1ns
   Dynamic Power Units = 1uW    (derived from V,C,T units)
   Leakage Power Units = 1pW


Attributes
----------
i - Including register clock pin internal power


  Cell Internal Power  = 100.3438 uW   (85%)
  Net Switching Power  =  18.0308 uW   (15%)
                       ---------
Total Dynamic Power    = 118.3746 uW  (100%)

Cell Leakage Power     =   8.5451 uW


              Internal         Switching         Leakage           Total
```

```
Power Group      Power          Power            Power           Power   (   %    ) Attrs
---------------------------------------------------------------------------------------------
io_pad           0.0000         0.0000          0.0000          0.0000  (   0.00%)
memory           0.0000         0.0000          0.0000          0.0000  (   0.00%)
black_box        0.0000         0.0000          0.0000          0.0000  (   0.00%)
clock_network    84.6101          0.0000          0.0000          84.6101  (  66.66%)  i
register         1.8069         0.3751       1.9337e+06          4.1156  (   3.24%)
sequential       0.0000         0.0000          0.0000          0.0000  (   0.00%)
combinational    13.9269          17.6557       6.6114e+06          38.1940  (  30.09%)
---------------------------------------------------------------------------------------------
Total          100.3438 uW       18.0308 uW     8.5451e+06 pW       126.9197 uW
1

****************************************
Report : power
       -hier
       -analysis_effort low
Design : mycpu
Version: T-2022.03-SP5-1
Date   : Mon Nov  3 23:31:45 2025
****************************************


Library(s) Used:

   saed32hvt_tt1p05v25c (File:
/research/cas/public/DT2_2025/lib/logic_lib/saed32hvt_tt1p05v25c.db)


Operating Conditions: tt1p05v25c   Library: saed32hvt_tt1p05v25c
Wire Load Model Mode: enclosed

Design       Wire Load Model          Library
------------------------------------------------
mycpu              8000           saed32hvt_tt1p05v25c
cu                ForQA           saed32hvt_tt1p05v25c
ir                ForQA           saed32hvt_tt1p05v25c
pc                 8000           saed32hvt_tt1p05v25c
fu                 8000           saed32hvt_tt1p05v25c
rb                ForQA           saed32hvt_tt1p05v25c
muxm                ForQA          saed32hvt_tt1p05v25c
muxb               ForQA           saed32hvt_tt1p05v25c
muxd_N3              ForQA          saed32hvt_tt1p05v25c


Global Operating Voltage = 1.05
Power-specific unit information :
   Voltage Units = 1V
   Capacitance Units = 1.000000ff
   Time Units = 1ns
   Dynamic Power Units = 1uW    (derived from V,C,T units)
   Leakage Power Units = 1pW


--------------------------------------------------------------------------------
```

```
                          Switch   Int     Leak     Total
Hierarchy                  Power    Power    Power    Power   %
------------------------------------------------------------------------------
mycpu                     18.031  100.344 8.55e+06  126.920 100.0
  MUXD (muxd_N3)             3.987    3.454 2.13e+05    7.655   6.0
  MUXB (muxb)                0.610    0.122 1.66e+05    0.898   0.7
  MUXM (muxm)                2.193    2.317 2.76e+05    4.786   3.8
  RB (rb)                    1.314   28.687 8.45e+05   30.847  24.3
  FU (fu)                    0.344    0.350 4.71e+06    5.403   4.3
  PC (pc)                    3.073   30.971 1.20e+06   35.242  27.8
  IR (ir)                    2.099   23.871 5.87e+05   26.556  20.9
  CU (cu)                    4.410   10.571 5.51e+05   15.533  12.2
1
```

Parameter	Value
Total Power	126.920 uW
Most Power-Hungry Power Group	clock_network (84.6101 uW, 66.66%)
Most Power-Hungry Submodule	PC (Program Counter) (35.242 uW, 27.8%)

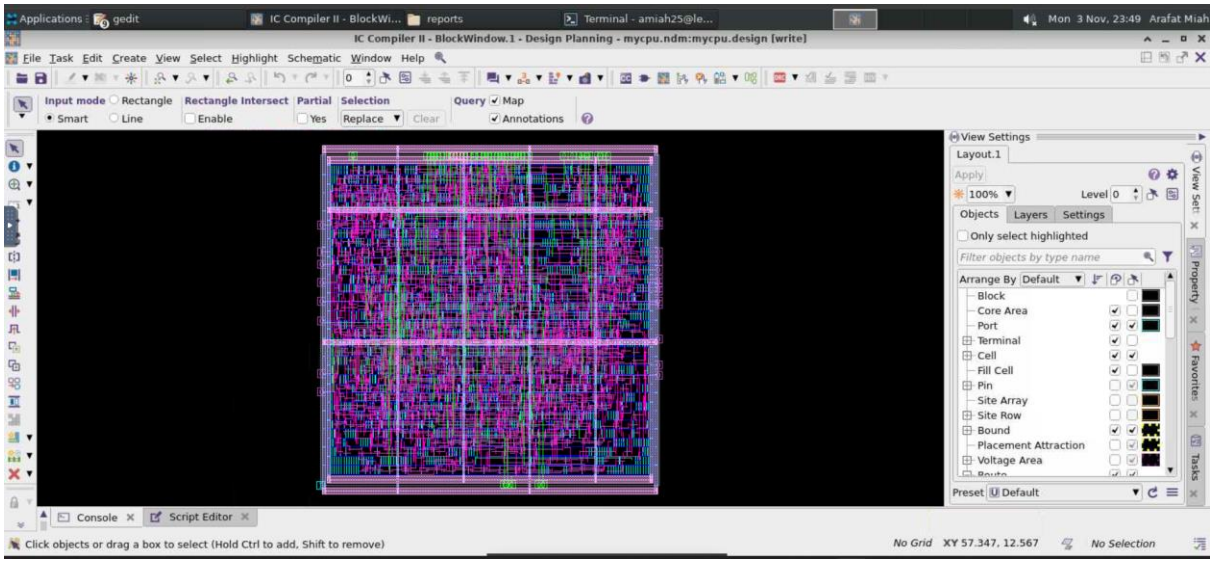
Comments on results.: The power analysis reveals that the clock network is the largest power consumer, accounting for 66.66% of the total power, which is typical for synchronous designs where the clock drives many sequential elements. Within the functional modules, the Program Counter (PC) is the most power-hungry submodule (27.8%), likely due to its required high switching activity on nearly every clock cycle for instruction fetching.

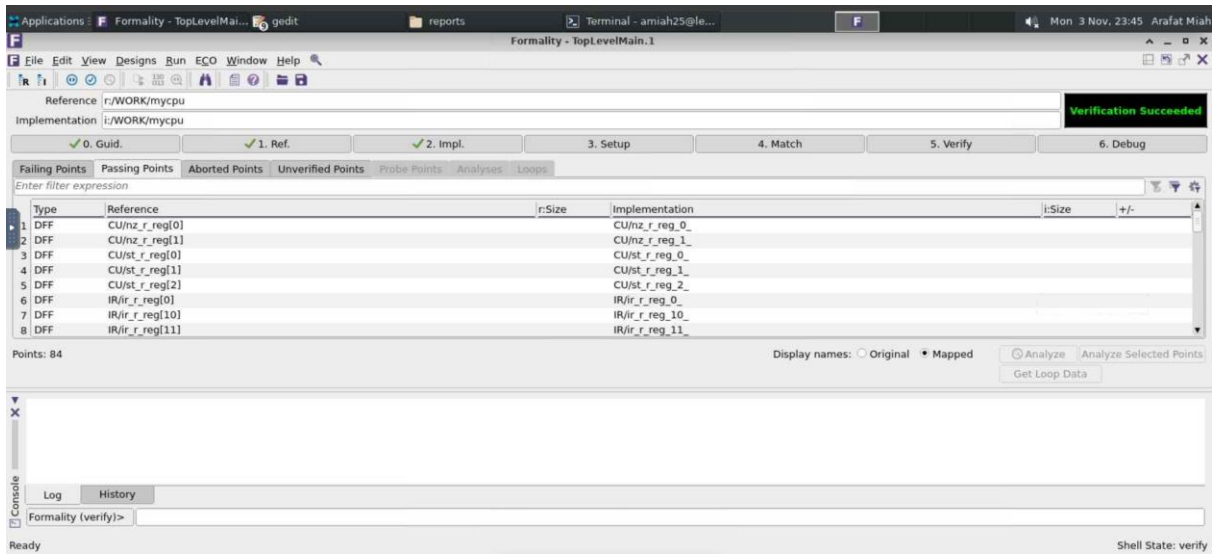
8.4. Gate-Level Verification Results

```
Formal Logic Equivalence Check Results:

**************************************************
Report        : status

Reference     : r:/WORK/mycpu
Implementation : i:/WORK/mycpu
Version       : U-2022.12
Date          : Mon Nov  3 23:45:27 2025
**************************************************

****************************** Synopsys Auto Setup Summary ******************************

!!! Synopsys Auto Setup Mode was enabled. !!!
!!! Verification results are valid assuming the following setup constraints: !!!
```

```
### RTL Interpretation Setup
  set hdlin_ignore_parallel_case false
  set hdlin_ignore_full_case false
  set hdlin_error_on_mismatch_message false
  set hdlin_ignore_embedded_configuration true

### Undriven Signal Handling Setup
  set verification_set_undriven_signals synthesis

### Test Logic Setup
  set verification_verify_directly_undriven_output false
  For details see report_dont_verify_points and report_constants

For further details on Synopsys Auto Setup Mode: Type man synopsys_auto_setup
****************************************************************************************



********************************* Verification Results *********************************
Verification SUCCEEDED
  ATTENTION: synopsys_auto_setup mode was enabled.
         See Synopsys Auto Setup Summary for details.
------------------------------------------------------------
 Reference design: r:/WORK/mycpu
 Implementation design: i:/WORK/mycpu
 84 Passing compare points
----------------------------------------------------------------------------------------
Matched Compare Points    BBPin   Loop   BBNet    Cut    Port    DFF    LAT   TOTAL
----------------------------------------------------------------------------------------
Passing (equivalent)        0      0      0      0      34      50      0     84
Failing (not equivalent)    0      0      0      0       0       0      0      0
****************************************************************************************
1
```

Present report starting below line ********* Verification Results ************





Comment on the results.
As we can see that the verification shows successful and we can also see a IC layout design of mycpu.

SELF ASSESSMENT

Write a summary of week's work and results, including

1. Assessment of results
    a. Does the code compile and elaborate without errors and warnings?
    There had error but I corrected it and I used sometime AI when I was confused what to do
    b. Do the RTL modules function, pass checks and synthesize correctly?
    Yes
2. Approximate time spent on project this week
    15hours
3. Open issues you need help with
4. Problems found in earlier weeks' code fixed this week