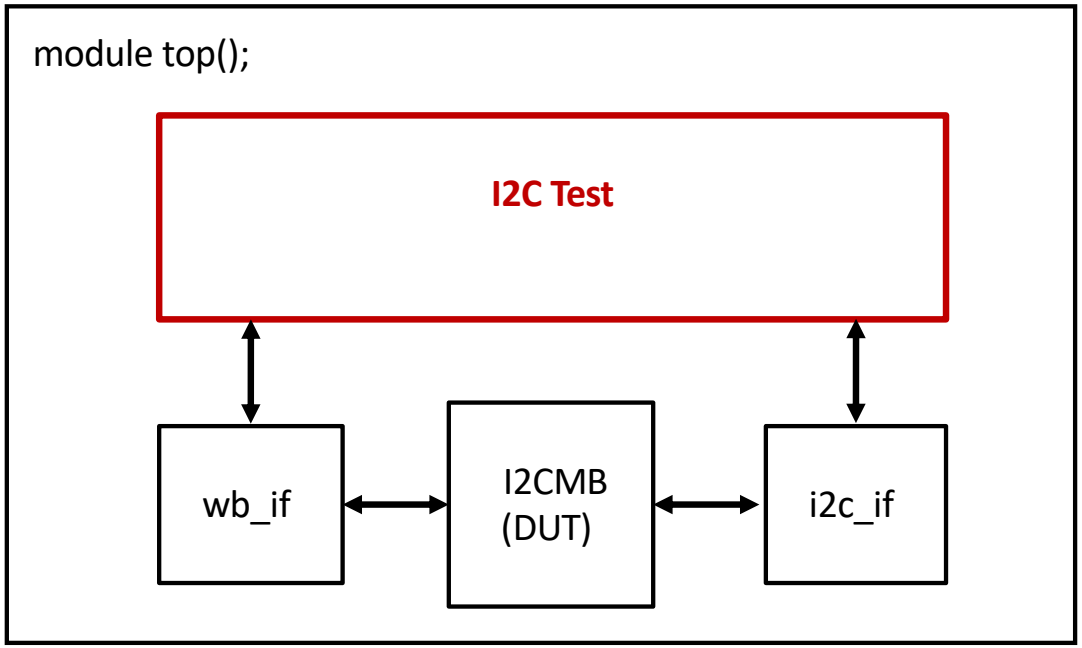
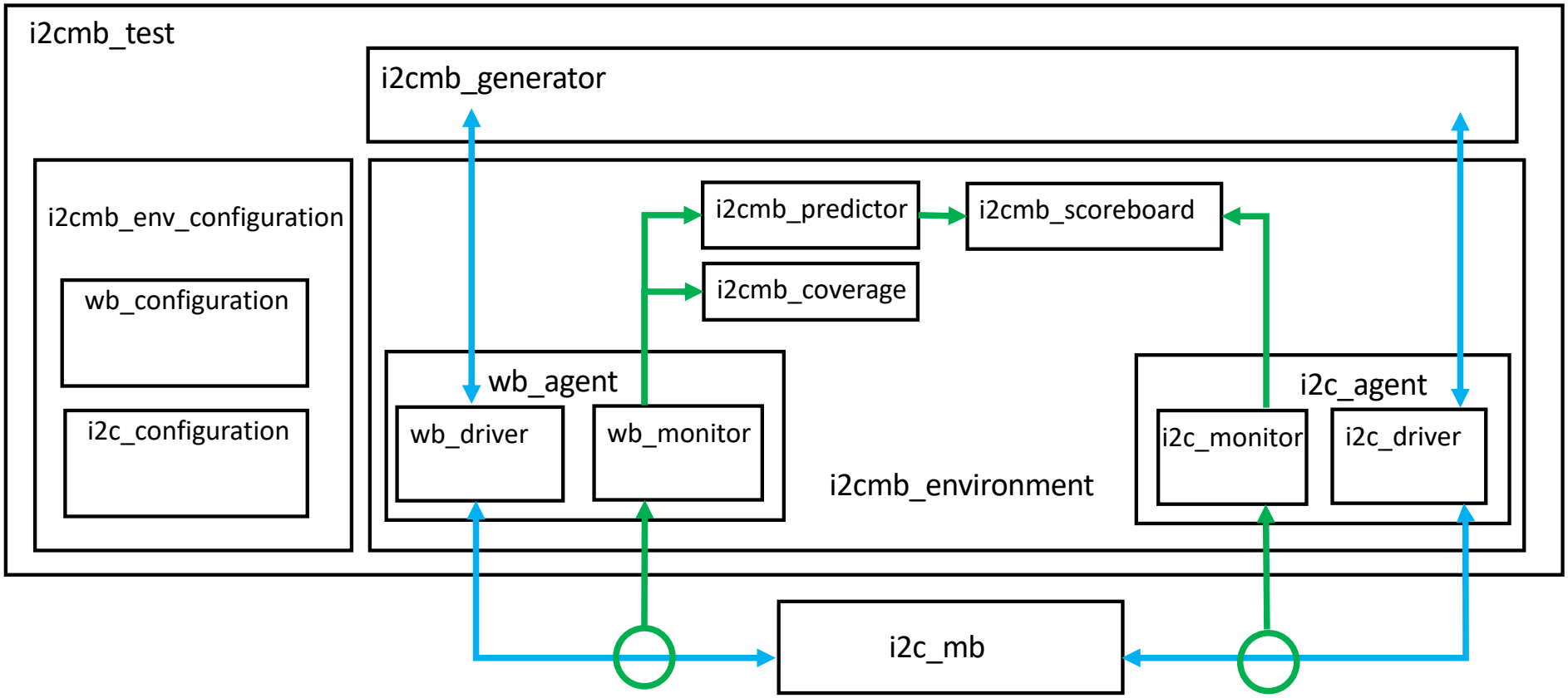
ECE 745

ASIC Verification

Project 2 Assignment – I2CMB Layered Test Bench



I2C Multi Bus Layered Test Bench Architecture



Provided in Assignment

- Directory structure and files from Project 1
- On Moodle – project_2_provided_files
  - Base class library – ncsu_pkg
  - Updated wb_pkg – new tasks in wb_if.sv
    - wait_for_interrupt()
    - wait_for_reset()
    - wait_for_num_clocks(int num_clocks);

Project Instructions - Setup

1. Make an environment_packages directory under verification_ip
2. In the verification_ip/environment_packages directory:
   1. Make an i2cmb_env_pkg directory
   2. Create a src directory under i2cmb_env_pkg
   3. Create an i2cmb_env_pkg.sv file under the i2cmb_env_pkg directory
   4. Create a Makefile under i2cmb_env_pkg that compiles i2cmb_env_pkg.sv
3. In the project_benches directory:
   1. Copy your proj_1 project directory to proj_2 directory
4. In the project_benches/proj_2 directory
   1. Include the i2cmb_env_pkg/Makefile in the sim/Makefile
   2. Include the ncsu_pkg/Makefile in the sim/Makefile
   3. Add the compile target for i2cmb_env_pkg.sv to the sim/Makefile as a dependency to the comp_bench target
   4. Add the compile target for ncsu_pkg.sv to the sim/Makefile as a dependency to the comp_bench target
5. Run 'make debug' to compile and run the bench
   - Simulation should run as in proj_1

Project Instructions – i2c_pkg Creation

- Create an i2c_pkg.sv in the i2c_pkg directory
  - Update the i2c_pkg/Makefile to compile i2c_pkg.sv
- Include the following files in i2c_pkg.sv
  - i2c_configuration, i2c_agent, i2c_driver, i2c_monitor, i2c_transaction
- Add the following files in i2c_pkg/src
  - i2c_configuration.svh, i2c_agent.svh, i2c_driver.svh, i2c_monitor.svh, i2c_transaction.svh
- Create the class definitions and implement the functionality for the following classes – classes should extend from classes in ncsu_pkg
  - i2c_configuration, i2c_agent, i2c_driver, i2c_monitor, i2c_transaction

Project Instructions – wb_pkg Creation

- Create an wb_pkg.sv in the wb_pkg directory
  - Update the wb_pkg/Makefile to compile wb_pkg.sv
- Include the following files in wb_pkg.sv
  - wb_configuration, wb_agent, wb_driver, wb_monitor, wb_transaction
- Add the following files in wb_pkg/src
  - wb_configuration.svh, wb_agent.svh, wb_driver.svh, wb_monitor.svh, wb_transaction.svh
- Create the class definitions and implement the functionality for the following classes– classes should extend from classes in ncsu_pkg
  - wb_configuration, wb_agent, wb_driver, wb_monitor, wb_transaction

Project Instructions – i2cmb_env_pkg Creation

- Create an i2cmb_env_pkg.sv in the i2cmb_env_pkg directory
  - Update the i2cmb_env_pkg/Makefile to compile i2cmb_env_pkg.sv
- Include the following files in i2cmb_env_pkg.sv
  - i2cmb_test, i2cmb_generator, i2cmb_env_configuration, i2cmb_environment, i2cmb_predictor, i2cmb_scoreboard, i2cmb_coverage
- Add the following files in i2cmb_env_pkg/src
  - i2cmb_test.svh, i2cmb_generator.svh, i2cmb_env_configuration.svh, i2cmb_environment.svh, i2cmb_predictor.svh, i2cmb_scoreboard.svh, i2cmb_coverage.svh
- Create the class definitions and implement the functionality for the following classes– classes should extend from classes in ncsu_pkg
  - i2cmb_test, i2cmb_generator, i2cmb_env_configuration, i2cmb_environment, i2cmb_predictor, i2cmb_scoreboard, i2cmb_coverage

Project Instructions – top.sv

- Place an instance of i2cmb_test within top.sv
- Modify the test_flow initial block to do the following
  - Place virtual interface handles into ncsu_config_db
  - Construct the test class
  - Execute the run task of the test after reset is released
  - Execute $finish after test complete

Project Instructions – i2cmb_generator

- Implement **project 1** I2C responses using i2c_transactions within the i2cmb_generator
- Implement **project 1 test flow** within i2cmb_generator using wb_transactions
- Wait for DON bit options
  - Poll the CMDR register
  - Use the wait_for_interrupt task in wb_if.sv

Project 1 and 2 Test Flow

- Write 32 incrementing values, from 0 to 31, to the i2c_bus
- Read 32 values from the i2c_bus
  - Return incrementing data from 100 to 131
- Alternate writes and reads for 64 transfers
  - Increment write data from 64 to 127
  - Decrement read data from 63 to 0
- Verify DUT operation using messages from i2cmb_scoreboard
- Note:
  - Project submission will be run by TA's to verify operation of these steps
    - Do not clutter up transcript with your debug messages

Project Submission

- Submit by 11:59pm on Sunday, March 24th
- Single tar file
  - Containing: ece745_projects directory and all sub-directories
    - Project_benches directory should include lab_1, proj_1, and proj_2 sub-directories
  - Named: <unityId>_p2.tar
  - Execute ‘make clean’ in sim directory before creating tar file