UVM FIFO Project

Create a full UVM environment for the FIFO implemented in the SV project.

Steps:

- Create a full environment for the FIFO design.
- Add constraints in the sequence item class.
- Add covergroups, coverpoints in the coverage collector class.
- Add assertions and bind it in the top module.
- Split the FIFO main sequence into multiple sequences based on your verification plan, for example:
    - write_only_sequence
    - read_only_sequence
    - write_read_sequence
- Note: You must not stick to the work done in your SV project. You are free to add or modify in the assertions, covergroups or constraints to enrich your verification.

Requirements:

- Verification plan
- Draw your UVM testbench showing the UVM structure using powerpoint, draw.io or MS Visio
    - Write a section where you will describe in details how the UVM testbench work, from the top module then driving the interface then monitoring and the analyzing the output
- Code Coverage report
- Functional Coverage report
- Sequential Domain Coverage report
- Bug report
- Sections with QuestaSim snippets to for each UVM sequence and how the interface is driven in each sequence using the waveform snippets

Submission file:

.rar file containing the following:

- PDF file having the requirements
- Testbench and design files
- Do file to run simulation (The project will not be graded if the do file is not working or missing)

Also, provide a table in your PDF file showing the assertions used as follows

Feature	Assertion
Whenever the FIFO is full, wr_ack is always = 0	@(posedge clk) (full |-> !wr_ack)