ECE - 585 - MICROPROCESSOR SYSTEM DESIGN
TEST PLAN - GROUP 6
Simulation of the scheduler portion of a DDR5 Memory Controller

Ashwin Kumar Sivakumar

PSU ID:977214327

Mahidhar Regalla

PSU ID:951927746

Swaroop Chandrashekar

PSU ID: 919564194

Sahil Khan

PSU ID:980980741

Test plan for closed-page policy (Level 0)

Scenario 1: Basic Read and Write

Purpose: Validate the fundamental functionality of the closed-page DDR5 design byexecuting simple read and write operations, ensuring correct data transfer.

Testcase 1: Take a request to read from a particular bank group, bank, row and column

Testcase 2: Take a request to write to a particular bank group, bank , row and column

Expected Result: The memory controller processes a single CPU request and providesDRAM commands as output to the DIMM for the specified Bank group, Bank, row, and column.

Scenario 2: Repeated Access within the Same banks

Purpose: Confirm that sequential read and write operations within the same bank exhibit the expected closed-page behavior by accounting the trp delay.

Execution: For the **testcase1** take 3 requests of only read where the memory controller needs to access the same bank again after closing the page.
For example: B0, B1, B0, ...
to the nearest CPU clock cycle.

For the **testcase2** take 3 requests of only write where the memory controller needs to access the same bank again after closing the page.
For example: B0, B1, B0, ...
to the nearest CPU clock cycle.

For the **testcase3** take multiple requests of mixing write and read where the memory controller needs to access the same bank again after closing the page.

For example: B0, B1, B0, ...
to the nearest CPU clock cycle.

For the **testcase4** take multiple requests of mixing write and read where the memory controller needs to access the same bank again after closing the page.
For example: B0, B1, B2, B3, B0, ...
to the larger gap of CPU clock cycle where trp delay is already satisfied before the next request to the same bank enters.
Expected result: Check if the memory controller introduces the required tRP delay when accessing the same bank again.
Verify that the scheduler correctly handles the timing constraints.

Boundary Testing:

For the **testcase5** Test scenarios where the page close and reopen are at the edge of tRP.

Expected result: It should start processing the ACT 0 in the immediate dimm clock after Trp

For the **testcase6** Test scenarios where the page close and reopen are at the middle of tRP.

Expected result: It should start processing the ACT 0 after satisfying the remaining Trp delay

---

Scenario 3: Sequential Access to Different Banks

Purpose: Assesses the sequential capability of the memory controller by issuing read and write requests to different banks simultaneously to exhibit the expected close pagebehavior without considering Trp delay.

Create test cases to sequentially access different banks without considering Trp delay.
For example: B0, B1, B2, ...
Verify Sequential Bank Access:
Ensure that the memory controller schedules commands for sequential bank accesses correctly.
Check the output to confirm that accessing different banks in sequence doesn't introduce Trp delay.

Execution: For **testcase1** access different banks without considering Trp delay for read alone
For example: B0, B1, B2, ... for the nearest CPU clock cycle.

For **testcase2** access different banks without considering Trp delay for write alone
For example: B0, B1, B2, ... for the nearest CPU clock cycle
For the **testcase3** take 4 or more instructions that are sequential mixed up with writes and reads for the different bank group, bank, and row to the nearest CPU clock cycle.

Expected result:

Ensure that the memory controller schedules commands for sequential bank accesses correctly.
Check the output to confirm that accessing different banks in sequence doesn't introduce Trp delay.

Scenario 4: Random Bank Access:

Purpose: Create test cases with random bank accesses, including both sequential to different banks and repeated accesses to same banks.
Mix and match different banks and patterns to simulate real-world scenarios.

Expected result:

Ensure that the memory controller schedules commands for sequential bank accesses to different banks and repeated access to the same bank correctly .
Check the output to confirm that accessing different banks in sequence doesn't introduce Trp delay and accessing to the same bank introduces Trp delay.

Scenario 5: Timing constraints check

Purpose: Evaluate the timings tRP,tRCD, tRTP, ,tburst following read or write operations to ensure optimal memory access and prevent unnecessary delays.

For **testcase 1** , read instruction we will check if the timing between activate and read which is tRCD and the timing after the read to precharge which is tRTP and timing between precharge and activate which is tRP and at last the time to get the data outwhich tcl+tburst after the read before precharge and to ensure command timings are met.

For **testcase 2** , write instruction we will check if the timing between activate and writewhich is tRCD and the timing after the write to precharge which is tRTP and timing between precharge and activate to another row instruction which tRP and at last the time to write data in which tcl+tburst after the write before precharge and to ensure command timings are met.

Note: DRAM commands like ACT0, ACT1, RD0, RD1, WR0, WR1 and PRE each willtake one DIMM clock cycle.

Expected output:

tRCD: The time between the activate command and the start of the read or writeoperation meets the specified tRCD timing.

tRTP: The time after the read or write operation to precharge meets the specified tRTPtiming.

tRP: The time between precharge and activate to another row instruction meets thespecified tRP timing.

tCL: The read to valid data out is within the specified limits. tBURST: The burst time meets the specified timing requirements.
DRAM commands like ACT0, ACT1, RD0, RD1, WR0, WR1 and PRE each will takeone DIMM clock cycle.

Scenario 6: check for Page empty

Purpose: To check the handling for page empty

For **testcase 1**, to check page empty give 2 requests, issuing first request at 0 CPUclock cycle and do not issue the second request until the first request is complete sothat the next ACT will not be issued for long time which makes it a page empty.

Expected result:
To check for the page empty scenario by violating the tRP for the next instruction ACT

Scenario 7: Back-to-Back Operations

Purpose: Assess the ability of the memory controller to handle consecutive read orwrite operations issued in rapid succession, checking for sustained performance.

For **testcase1** take 20 consecutive reads to the same bankgroup, bank, row and checkfor the functionality by the analysis of the obtained data

For **testcase2** take 20 consecutive writes to the same bank group, bank,row and checkfor the functionality by the analysis of the obtained data

For **testcase3** take consecutive 10 reads to the same address i.e same bankgroup, bank,row and column to check for the functionality by the analysis of the obtained data.

For **testcase4** take consecutive 10 writes to the same address i.e same bankgroup, bank,row and column to check for the functionality by the analysis of the obtained data.

Expected output: The memory controller successfully manages consecutive read/writerequests to the same address and also for the same row without notable performance degradation.

Analyze the obtained data to confirm that each read/write operation correctly stores thespecified data in the designated address.

Sustained performance is demonstrated by the consistent and timely completion of all10 consecutive read/write operations to the same address.

Scenario 8: Boundary Testing

Purpose: Explore edge cases, such as out of boundary for addresses, ensuring theclosed-page DDR5 design handles extreme scenarios robustly.

check the constraints including operation and channel are obeying within the definedrange of condition.

Testcase 1: Operation- Take the value of operation greater than 2 and test whether ifthis condition is failing as it is exceeding the limit of the designed range.

Channel bit- Check this testcase by giving the channel bit as 1 and check for the failureof boundary condition as we have defined the design for channel 0.

Expected Output:

The test should result in a failure, indicating that the channel bit or operation input is notwithin the defined range.

The memory controller should reject the request and may issue an error or exception

---

Scenario 9: Queue testing

Testcase 1: Check for queue empty

Purpose: Check for queue empty .The expected result is to confirm that, after processing all given input requests, the memory controller accurately recognizes and reports the empty queue condition.

Check for the condition by giving set of requests to a nearer CPU clock cycle and waituntil all the requests are satisfied and queue becomes empty.

Expected Result: if all the given input requests are satisfied and no request is pendingthen display as queue empty.

Testcase 2: Check for queue full

Purpose: Verify that the memory controller correctly handles the case when the queueis full.

Test Steps: Fill the memory controller queue to its capacity. Attempt to add more operations to the input queue. Verify that the memory controller stalls the input operations until there is space in the queue.

Expected Result: The memory controller correctly stalls input operations when thequeue is full.

Testcase3: Check for queue full by adding and removing request at the same time.

Purpose: The purpose is to assess the system's ability to handle the complexity of both operations occurring concurrently under full queue conditions. The expected result is to verify whether the memory controller correctly recognizes a full queue and effectively manages the simultaneous addition and removal of requests

Expected result: verify if the queue is full and check at the same time for add orremove requests happening simultaneously.

Testcase4: Check for adding and removing request at the same time without queue isfull.

Purpose: The purpose is to verify the controller's correct synchronization during concurrent pushing and popping operations at even CPU cycles.

Check this condition by the giving the input at the CPU at an even cycle which is at theDIMM clock, then the input scheduled at that is pushed into the queue and then the request at the front of the queue is popped simultaneously. Here pushing and poppinghappens at the same CPU cycle when the queue is not full.

Expected output: The memory controller successfully handles the simultaneousaddition and removal of requests when the queue is not full.

The requests are pushed into the queue, and the request at the front of the queue ispopped simultaneously.

The memory controller returns the expected result, indicating correct handling of simultaneous adding and removing when the queue is not full.

Scenario 10 : CPU clock instruction

Purpose: To check if the functionality is working proper for request in different CPU clock cycles.

Testcase 1: To test if there is new instruction on CPU clock after a long gap.

Check this by taking two consecutive requests scheduled with a larger time interval.

Expected output:

The memory controller should detect the new instruction on the CPU clock after a long gap.

The requests scheduled with a larger time interval should be processed, and the memory controller returns the expected result indicating the successful execution of theinstructions.

Testcase 2: To test if there is new instruction on every CPU clock without any gap.

Check this by taking two consecutive requests scheduled with consecutive time intervalwithout gap.

Expected output: The memory controller should detect the new instruction on every CPU clock without any gap.

The requests scheduled with consecutive time intervals should be processed efficiently,and the memory controller returns the expected result indicating the correct handling of consecutive instructions.

Testcase3: Test for zero simulation/CPU clock time

This is to be tested by giving a request at the zeroth simulation time and observing thetranscript.

Expected output:

The memory controller should detect the request at the zeroth simulation time.

The request should start processing without any issues at the first DIMM time, and thememory controller returns the expected result indicating the correct handling.

Testcase4: Test for giving request at the maximum simulation time/CPU clock.

This is to be tested by scheduling a request at the maximum value of CPU clock time.

Expected output: The memory controller should detect the request at the maximum simulation time. The request should be processed without any issues, and the memorycontroller returns the expected result indicating the correct handling of a request at themaximum simulation time.

Scenario 11: Stress Testing

Purpose: Apply stress to the system by continuously issuing read and write requests, evaluating the design's stability and sustained performance.

Testcase 1: take a very larger number of consecutive reads and write requests to thesame bank group, bank, row and check for the functionality by the analysis of the obtained data.

Expected result:

The memory controller should handle the stress by processing the consecutive readand write requests efficiently.

Analyze the obtained data to ensure that each read/write operation retrieves the correctdata from the specified bank group, bank, and row.

Scenario 12: Complete Simulation

Purpose: Run a complete simulation with a sequence of read and write operations,queue full scenarios, and empty queue scenarios.

Test Case steps:

Add a sequence of read and write operations to the input queue.Introduce
scenarios where the queue becomes full.
Introduce scenarios where the queue becomes empty.

Verify that the memory controller processes operations correctly and produces theexpected output.

Expected Result: The memory controller handles various scenarios correctly throughout the simulation.

Scenario 12: Debugging Information

Description: Verify that the memory controller produces useful debugging informationwhen debug_en is set.

Test Case Steps:

Set debug_en to 1.

Run a simulation with various operations.

Verify that the debugging information in the output file is informative.

Expected Result: The debugging information in the output file helps identify any issuesor unexpected behavior.
