

B Changes in Version 1.2. – April 7th 2012

To switch from version 1.0 or 1.1 to version 1.2, download the new version and copy a previously written scheduler.c and scheduler.h to the src/ directory in the new version. In going from version 1.1 to version 1.2, only the code in file memory_controller.c has changed. We've also removed a few terms from the license to make it easier for groups to use the simulator. USIMM Version 1.2 incorporates the following changes:

- Bug fix: The clean_queue function in the file memory_controller.c has been updated to delete the read and write queue nodes after the corresponding request has been serviced. This fixes a memory leak in the simulator.
- Bug fix: The issue_request function in the file memory_controller.c has been updated to fix a bug in the calculation of the next_write variable for all banks on the channel after a read or write command is issued. In version 1.1, the next_write variable was set to a larger value than what it ideally should have been. The net effect of the change is that now a write command following a read or write command can be issued earlier than in version 1.1.
- Bug fix: In the function update_memory in the file memory_controller.c, a condition has been added to make sure that forced refreshes are not issued if the mandatory 8 (or more) refresh commands have already been issued in the refresh window.
- Bug fix: In the function issue_powerdown_command, before issuing a powerdown-slow command, the function is_powerdown_slow_allowed is now being called correctly to check the issuability of the command. Earlier, there might have been situations where a powerdown_slow would be issued even if only a powerdown-fast was allowed by the timing constraints (imposed solely by the refresh deadline).
- Bug fix: In the function issue_auto_precharge in the memory_controller.c file, the calculation of the commencement of the auto-precharge command is now updated to be the maximum of the next_pre timing set by earlier commands and the first cycle when a precharge can be issued following a read or write command.
- Bug fix: Initialized the user_ptr field in the request_t structure to NULL when a new read or write entry is enqueued in the corresponding queue. This variable can now be checked in the schedule function to determine if a read or write node has been recently enqueued; this allows the user to initialize other user-defined variables soon after a request is enqueued.