# AWDR: Allocation-aware Dram Refresh

Nurlan Kaisar

nurlan_k0811@hbalm.nis.edu.kz

## Abstract

*DRAM memory is a dynamic memory connecting with CPU along in multicore-architecture. This memory type allows CPU take current process data through Cache.Integration of multi-core processors forced architectures increase capacity of Dram or Dimm memory and chips.Which in turn, computer losing important aspects,such as performance and energy.*

*Since the refresh of Dram arrays are consuming plenty of energy and performance.We as architects should solve this problem due to prevent program or software issues and unstable execution.*

*Our studies show that AWDR significantly or gradually increase the performance depending on program or task itself* In an system with 8 GB DRAM,AWDR achieves a 80.7% refresh reduction, an average DRAM power reduction of 10.7%, and an average system performance improvement of 7.5% over existing systems. AWDR benefits are robust to variation in DRAM system configuration.

## 1. Introduction

Modern memory main is defined as dynamic random-access memory(Dram).Dram Stores data as a electric charge in cells.Over time, charge might be leaked and lose the data.Which in turn, might lead to software crush

- First, unfair resource sharing would render system software's (operating system or virtual machine) priority-based thread scheduling policies ineffective and therefore cause significant discomfort to the end user who naturally expects threads with higher (equal) priorities to get higher (equal) shares of the performance provided by the computing system.
- Malicious programs that intentionally deny service to other threads can be devised by exploiting the unfairness in the resource sharing schemes . This would result in significant productivity loss and degradation in system performance.
- Unfairness would reduce the performance predictability of applications since the performance of an application becomes much more dependent on the characteristics of other applications running on other cores. This would make it difficult to analyze and optimize system performance.
- In commercial *grid computing* systems where users are charged for CPU hours, unfair resource sharing would result in very unfair billing procedures because the performance a user program experiences would not necessarily correlate with CPU hours it takes as it is dependent on other programs running on the CMP.