Memory-Map Selection for Firm Real-Time SDRAM Controllers
Sven Goossens, Tim Kouters, Benny Akesson, and Kees Goossens

1. SDRAM
- Consists of banks, rows and columns.
- The controller decides which address bits map to the bank row and column respectively. This is the memory map.
- 5 commands: activate (ACT), precharge (PRE), read (RD), write (WR), refresh (REF).
- Can only RD or WR to the activated row in the row buffer.
- Timing constraints between the commands have to be satisfied.

2. Problem statement
How does the memory map influence the memory performance in terms of bandwidth, latency and power? Which memory map should an SDRAM controller use to satisfy the real-time requirements of its applications?

3. An example read access for a DDR3-800 using a (BI 2, BC 4) memory map

Step 1: Determine the worst-case bandwidth guaranteed by each memory map.
Step 2: Determine the worst-case power for each memory map. Prune the design space using your power budget.
Step 3: Determine the worst-case latency guaranteed by each memory map. Prune the design space using the latency requirements ($\Theta_{lat}$).
Step 4: Convert bandwidth requirements into an aggregate bandwidth requirement that takes the request size into account. Prune the design space with this bandwidth requirement.

4. Memory map selection guidelines
- Use BI ≤ 4 banks to avoid hitting the 4-ACT window constraint.
- Minimize timing constraint interference by choosing a large BI and/or BC. This maximizes the worst-case bandwidth.
- Match the AG of the memory map with the request size. This maximizes the useful bandwidth for the applications.
- Minimizing AG minimizes latency, at the expense of bandwidth.
- For a given AG, reducing BI reduces power, at the expense of bandwidth.

5. Conclusions
- Existing firm real-time controllers provide only limited memory map configurability.
- We allow the number of banks requests are interleaved over (BI) to be configured.
- Used to optimize the memory map based on the mix of applications.
- Beneficial for the worst-case performance in terms of bandwidth, latency and power.
- Allows for smaller access granularities, increasing performance for small requests.
- Combined with BC the complete range of access granularities is selectable.
- We propose a methodology that takes the real-time application constraints and the power budget into account and finds a memory map satisfying the bandwidth, latency and power requirements.