

Hybrid Memory Cube in Embedded Systems

Carlos Michel Betemps^{†‡}, Bruno Zatt[†], Mauricio Lima Pilla[†]

[†]Federal University of Pelotas (UFPEL) - Graduate Program in Computing (PPGC) - Pelotas, RS, Brazil

[‡]Federal University of Pampa (UNIPAMPA) - Campus Bagé - Bagé, RS, Brazil

{cm.betemps, zatt, pilla}@inf.ufpel.edu.br

Abstract—Paper abstract [Problem. Solution. Methodology. Results.].

I. INTRODUCTION

[Hybrid Memory Cube. Embedded Systems. Paper's Objective. Methodology. Paper's structure.]

Objetivos:

- Realizar um estudo de revisão sobre memórias HMC ([10], [18], [3], [15]).
- Avaliar a utilização de Memórias HMC, em substituição às memórias DDR, como memória principal em sistemas embarcados (a partir de metodologia apresentada na próxima seção).
- Analisar resultados obtidos, apontando tendências, resultados interessantes e lições aprendidas.

II. RELATED WORKS

[Works that present and/or use HMC memories.]

Vários estudos abordaram o uso de memórias HMC e correlatas. Focando em um escopo mais amplo, especificamente em tecnologias 3D, o trabalho de Zou et al.[18] aponta a integração de memórias 3D em arquiteturas heterogêneas, possibilitando a integração de tecnologias diferentes no mesmo chip. Já o trabalho de Beica [3] apresenta uma revisão das tecnologias 3D com integração via TSV (*Through-Silicon Via*), focando em aplicações e tendências de mercado. Suresh et al. [17] apresenta uma avaliação da aplicação das tecnologias emergentes de memória no contexto de HPC e aplicações intensiva em dados, experimentando arquiteturas híbridas de memórias voláteis e não-voláteis. Santos et al. [16] apresenta experimentos com uso de HMC (com reduzida latência) em aplicações streaming e aponta situações em que o uso de caches L3 é desnecessário. Outro estudo lida com questões de desempenho e energia de uma memória HMC *Gen2* na execução de aplicações centradas em dados - emulação e execução são combinadas em uma placa FPGA [8]. Alves et al. [1] apresentam extensões na memória HMC para possibilitar o processamento-em-memória (PIM) de operações vetoriais, visando evitar a contenção nos canais de comunicação e poluição na memória cache. *Active Memory Cube* (AMC) é a arquitetura de processamento-em-memória apresentada no trabalho de Nair et al. [14], a mesma apresenta um conjunto de unidades de processamento implementadas na camada de base da memória (HMC). Basicamente os trabalhos apontam as vantagens do uso de memórias HMC, ressaltando a melhoria de latência, largura de banda, potência e densidade [10]. Dado

que sistemas embarcados normalmente possuem requisitos restritos quanto a área e consumo energético, mas ao mesmo tempo requisitos exigentes quanto ao tempo de execução e capacidade de processamento, vislumbra-se a possibilidade da aplicação de memórias HMC em sistemas embarcados, visando o aumento de desempenho (latência e largura de banda) com um eficiente consumo energético e de área. Assim, o trabalho visa realizar uma revisão do estado da arte sobre memórias HMC e experimentos (simulados) que visam avaliar a possibilidade de aplicar memórias HMC como memória principal em arquiteturas de sistemas embarcados. Memórias HMC estão em pleno desenvolvimento e estudo. O *Hybrid Memory Cube Consortium* [6] reúne uma série de parceiros dedicados ao desenvolvimento desta tecnologia de memória.

III. HYBRID MEMORY CUBE REVIEW

[A review about HMC memories.]

IV. BACKGROUND

[Concepts, Tools, Benchmarks, Standards, etc. used in the work.]

V. METHODOLOGY

[Presents the detailed steps applied in the work's development, mainly the ones related to the experiments.]

Passos:

- Realizar levantamento de artigos que abordem o estado-da-arte sobre memórias HMC, focando em sua arquitetura de implementação, suas aplicações, vantagens de uso e problemas associados. Deverão ser utilizados artigos publicados em eventos e/ou periódicos. As pesquisas serão realizadas utilizando a máquina de busca do Google Scholar¹;
- Utilizar o simulador gem5 [4], [7] para simular a arquitetura ARM [2] - considerando a grande utilização de processadores ARM no contexto de sistemas Embarcados;
- Preparar o *setup* para a simulação a ser realizada no gem5 [7]. Verificar como utilizar simulação com mais de um elemento de processamento (core) visando uma maior "pressão" sobre o sistema de memória;
- Utilizar programas do *benchmark* MiBench [9] para execução no simulador gem5;
- Realizar a geração, a partir dos códigos fonte do benchmark MiBench [13], com o compilador

¹<https://scholar.google.com.br/>

- gcc-arm-gnueabi (cross-compiling). Como programas para execução pretende-se utilizar pelo menos um programa em cada categoria do benchmark;
- Utilizar a ferramenta CACTI-3DD [5] para levantar dados de potência, área e tempo considerando as memórias HMC e DDR. Há uma dificuldade, até o momento, na obtenção da ferramenta. Alternativas são zsim-nvmain² e 3D-Memory-Simulator³;
 - Utilizar o simulador CasHMC [11] para levantar dados de latência e largura de banda. O simulador recebe como entrada traços (traces) de uso de memória. Como saída devolve informações como latência e largura de banda. Caso necessário, como alternativa, pode ser analisado o uso do simulador HMC-Sim [12];
 - Realizar as simulações considerando as seguintes configurações para a hierarquia de memória:
 - L1: tamanho de 32 KB, associativa 8-way, tamanho de linha 64 B (proposta inicial)
 - Memória Principal: 512MB (confirmar limitação do gem5).
 - * DDR
 - * HMC
 - Executar simulações com as seguintes hierarquias de memória:
 - * L1 + DDR (ddr) - base
 - * L1 + HMC (hmc)
 - * L1 + L2 + DDR (l2+ddr)
 - * L1 + L2 + HMC (l2+hmc)
 - Geração de estatísticas, na execução de cada programa em cada diferente configuração, para posterior levantamento de estimativas de tempo de execução, consumo energético, EDP (Energy-Delay Product), latência, largura de banda e área; com base nas estatísticas geradas pelo gem5, CACTI-3DD e CasHMC.
 - Análise dos dados levantados, geração de resultados e gráficos de interesse, discussão sobre os resultados, apresentação de conclusões e encaminhamento de possíveis trabalhos futuros.

VI. RESULTS AND ANALYSIS

[Presents the results and its analysis.]

VII. CONCLUSION AND FUTURE WORK

[Present the learned lessons, conclusions and possibilities of enhancement and future works.]

REFERENCES

- [1] Marco AZ Alves, Matthias Diener, Paulo C Santos, and Luigi Carro. Large vector extensions inside the hmc. In *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2016, pages 1249–1254. IEEE, 2016.
- [2] ARM Ltd. <http://www.arm.com/>, 2017. [Online. Accessed 10-Jul-2017].
- [3] Rozalia Beica. 3d integration: Applications and market trends. In *3D Systems Integration Conference (3DIC)*, 2015 International, pages TSS–1. IEEE, 2015.
- [4] Nathan Binkert, Bradford Beckmann, Gabriel Black, Steven K Reinhardt, Ali Saidi, Arka Prava Basu, Joel Hestness, Derek R Hower, Tushar Krishna, Somayeh Sardashti, et al. The gem5 simulator. *ACM SIGARCH Computer Architecture News*, 39(2):1–7, 2011.
- [5] Ke Chen, Sheng Li, Naveen Muralimanohar, Jung Ho Ahn, Jay B Brockman, and Norman P Jouppi. Cacti-3dd: Architecture-level modeling for 3d die-stacked dram main memory. In *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2012, pages 33–38. IEEE, 2012.
- [6] Hybrid Memory Cube Consortium. Hybrid memory cube specification rev. 2.1.
- [7] gem5. The gem5 Simulator - A modular platform for computer-system architecture research. http://gem5.org/Main_Page, 2017. [Online. Accessed 10-Jul-2017].
- [8] Maya Gokhale, Scott Lloyd, and Chris Macaraeg. Hybrid memory cube performance characterization on data-centric workloads. In *Proceedings of the 5th Workshop on Irregular Applications: Architectures and Algorithms*, page 7. ACM, 2015.
- [9] M. R. Guthaus, J. S. Ringenberg, D. Ernst, T. M. Austin, T. Mudge, and R. B. Brown. Mibench: A free, commercially representative embedded benchmark suite. In *IEEE International Workshop on Workload Characterization, 2001. WWC-4.*, WWC '01, pages 3–14, Washington, DC, USA, 2001. IEEE Computer Society.
- [10] Joe Jeddell and Brent Keeth. Hybrid memory cube new dram architecture increases density and performance. In *VLSI Technology (VLSIT)*, 2012 Symposium on, pages 87–88. IEEE, 2012.
- [11] Dong-Ik Jeon and Ki-Seok Chung. Cashmc: A cycle-accurate simulator for hybrid memory cube. *IEEE Computer Architecture Letters*, 16(1):10–13, 2017.
- [12] John D Leidel and Yong Chen. Hmc-sim: A simulation framework for hybrid memory cube devices. *Parallel Processing Letters*, 24(04):1442002, 2014.
- [13] MiBench. Github - embecosm/mibench: The mibench testsuite, extended for use in general embedded environments. <https://github.com/embecosm/mibench>, 2012. [Online. Accessed 26-Jan-2017].
- [14] Ravi Nair, Samuel F Antao, Carlo Bertolli, Pradip Bose, Jose R Brunheroto, Tong Chen, C-Y Cher, Carlos HA Costa, Jun Doi, Constantinos Evangelinos, et al. Active memory cube: A processing-in-memory architecture for exascale systems. *IBM Journal of Research and Development*, 59(2/3):17–1, 2015.
- [15] J Thomas Pawlowski. Hybrid memory cube (hmc). In *Hot Chips 23 Symposium (HCS)*, 2011 IEEE, pages 1–24. IEEE, 2011.
- [16] Paulo C Santos, Marco AZ Alves, Matthias Diener, Luigi Carro, and Philippe OA Navaux. Exploring cache size and core count tradeoffs in systems with reduced memory access latency. In *Parallel, Distributed, and Network-Based Processing (PDP)*, 2016 24th Euromicro International Conference on, pages 388–392. IEEE, 2016.
- [17] Amoghavarsha Suresh, Pietro Cicotti, and Laura Carrington. Evaluation of emerging memory technologies for hpc, data intensive applications. In *Cluster Computing (CLUSTER)*, 2014 IEEE International Conference on, pages 239–247. IEEE, 2014.
- [18] Qiaosha Zou, Matthew Poremba, Rui He, Wei Yang, Junfeng Zhao, and Yuan Xie. Heterogeneous architecture design with emerging 3d and non-volatile memory technologies. In *Design Automation Conference (ASP-DAC)*, 2015 20th Asia and South Pacific, pages 785–790. IEEE, 2015.

²<https://github.com/AXLEproject/axle-zsim-nvmain>

³<https://github.com/shreyas-singapura/3D-Memory-Simulator>