Data tiling for memory system design space exploration

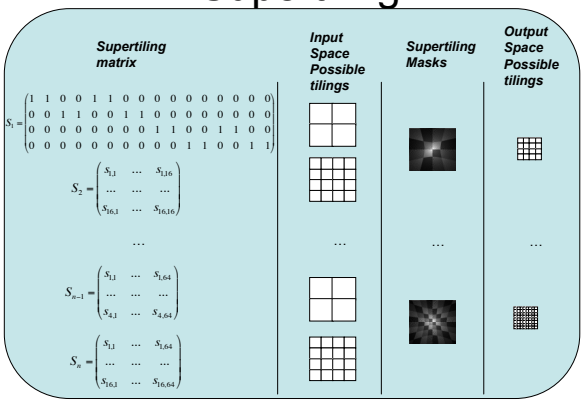
R. CORVINO*, S. MANCINI*, R. GUZZETTI**, B. DURETTE*, D. HOUZET*, J.HERAULT*
*GIPSA-Lab, INPG 46 Felix Viallet 38031 Grenoble rosilde.corvino@lis.inpg.fr
** Stmicroelectronics, HLS Group, 850, rue Jean Monnet 38926 Crolles

Context: High level synthesis of real-time systems with large amount of data

Problem: Reduce power consumption in memory access and bandwidth bottlenecks by reducing redundancy in data transfer and by using hierarchical memory organization

Idea: Tile separately input and output data spaces for a panel of possible tile's volume. Link tilings to each other by supertiling algorithm

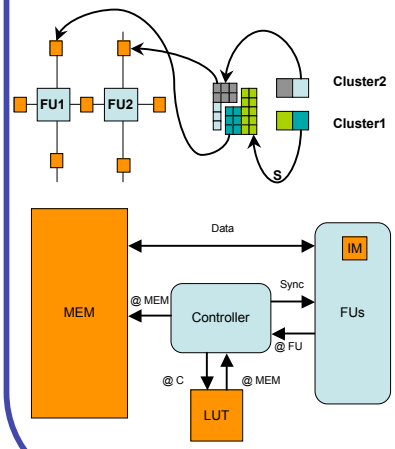
Supertiling



- Build input and output spaces
- Build supertiling space
- Explore supertiling space
  - –Chose supertiling with minimum input volume for all output tile calculations
  - –Schedule output tiles processing in order to reduce memory usage and input tiles transfer

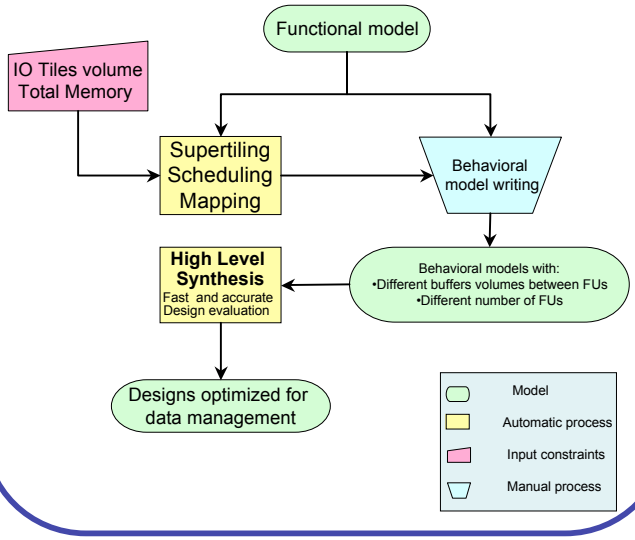
Hardware

Pre-calculated mapping and scheduling



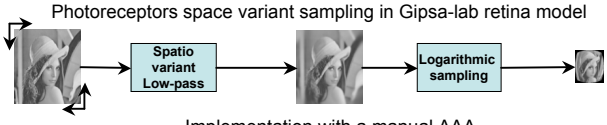
- The management of dependencies between tiles held by designer
- Possible mappings [1] of output tiles processing
  - –Sequentially on the same functional unit
  - –In parallel on multiple units
- A controller used to manage data transfer and functional units synchronization

Design flow impact

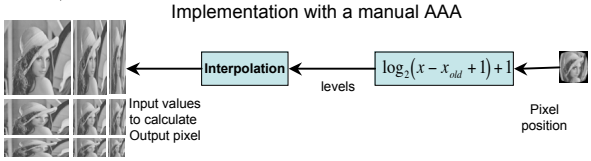


Application

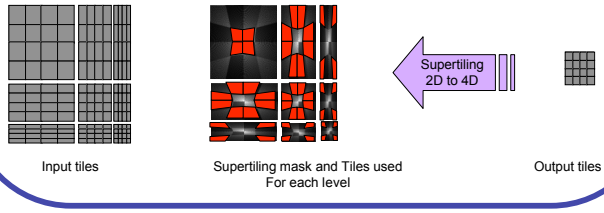
Photoreceptors space variant sampling in Gipsa-lab retina model



Implementation with a manual AAA



Supertiling applied to space variant filter MIP-MAPPED



INNOVATIONS

- ✓Tile data space instead of iteration space [2]
- ✓Appliable to treatments with non-affine memory access
- ✓Data management-driven design space exploration
- ✓Local and global optimizations of data management

PERSPECTIVES

- ✓Automate manual operations
- ✓Study the sensitivity of Supertiling to IO tiles shape
- ✓Treat automatically internal dependencies
- ✓Test supertiling on different targets (multiprocessors, NOC, etc.)
- ✓Test other applications

[1] A. Darte,"A Constructive solution to the Juggling Problem in Processor Array Synthesis", IEEE Comp. Soc., 2000
[2] D. Moldovan, Partitioning and mapping of algorithms into fixed size systolic arrays", IEEE trans. on Comp., 1986