



UvA-DARE (Digital Academic Repository)

High performance reconfigurable computing with cellular automata

Murtaza, S.

Publication date
2010

[Link to publication](#)

Citation for published version (APA):

Murtaza, S. (2010). *High performance reconfigurable computing with cellular automata*.

General rights

It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations

If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: <https://uba.uva.nl/en/contact>, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

Acronyms and Symbols

ASIC	Application specific integrated circuit
BRAM	Block RAM
CA	Cellular automata
CB	Compute block
CE	Compute engine
CLB	Configurable logic block
COTS	Commercial off the shelf
DDR	Double data rate
DRAM	Dynamic RAM
DSP	Digital signal processor
FDTD	Finite difference time domain
FPGA	Field programmable gate array
HPC	High performance computing
HPRC	High performance reconfigurable computing
LBM	Lattice Boltzmann method
LGCA	Lattice Gas Cellular Automata
LUT	Lookup table
MLUPS	Million lattice-site updates per second
PCI	Peripheral component interconnect
PE	Processing element
PMC	PCI mezzanine card
RAM	Random access memory
SDRAM	Synchronous dynamic RAM
SRAM	Static RAM
T_{pc}	CA execution time on a PC
T_{fpga}	CA execution time on a FPGA
T_{ft}	CA execution time on a FPGA-enabled PC
T_{pre}	Time to pre-process a CA lattice by a host machine
T_{pos}	Time to post-process a CA lattice by a host machine
T_s	Time to download a CA lattice from the host PC to the FPGA on-board memory
T_r	Time to upload a CA lattice from the FPGA on-board memory to the host PC

T_{fo}	Total overhead time ($= T_{pre}+T_s+T_r+T_{pos}$)
p	Number of PEs running in a FPGA
k	Number of CA cell FPGA reads in parallel from source memory bank
N	Number of CA lattice cells
x	Number of CA lattice columns
y	Number of CA lattice rows
n	Number of compute blocks (pipeline depth) within a FPGA
g	Number of CA iterations
w	Buffer size within a compute block
F	Number of FPGAs
τ_r	FPGA time to read k -cells from the on-board memory bank
τ_w	FPGA time to write k -cells to a on-board memory bank
τ_c	FPGA time to update a cell
τ_d	Host PC time to download a cell to a FPGA on-board memory
τ_m	Time to update a boundary cell across two neighbouring FPGAs
T_c	FPGA execution time for a compute bound computation
T_i	FPGA execution time for an I/O bound computation
T_1	FPGA execution time to compute a single CA iteration
T_2	Execution time to compute a single CA iteration using a dual FPGA-enabled system
T_f	Execution time to compute a CA iteration using a F FPGA-enabled PC
T_F	Execution time to compute a CA iteration using a F FPGA-enabled PC cluster