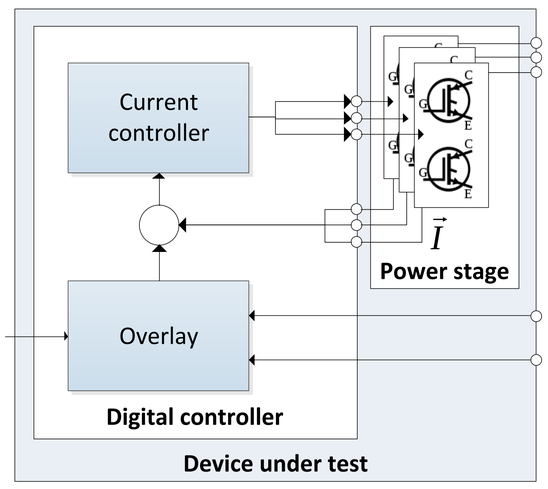
SYSTEM#: HIGH-LEVEL SYNTHESIS OF PHYSICAL SIMULATIONS FOR FPGA-BASED REAL-TIME EXECUTION

Christian Köllner, Nico Adler, Klaus D. Müller-Glaser



HARDWARE-IN-THE-LOOP TEST AND ELECTRIC MOTOR EMULATION

Electrified vehicular powertrains undergo exhaustive test procedures during development. In order to test an electric motor controller, motor and vehicle mechanics are replaced by a virtual device - the electric motor emulator (EME). An EME computes motor currents and rotor position from repeated voltage measurements and recreates these currents with a power stage. An EME involves a built-in real-time model requiring a challenging reaction time of approximately 1 µs. This is achieved by executing the model on an FPGA.

THE PROBLEM

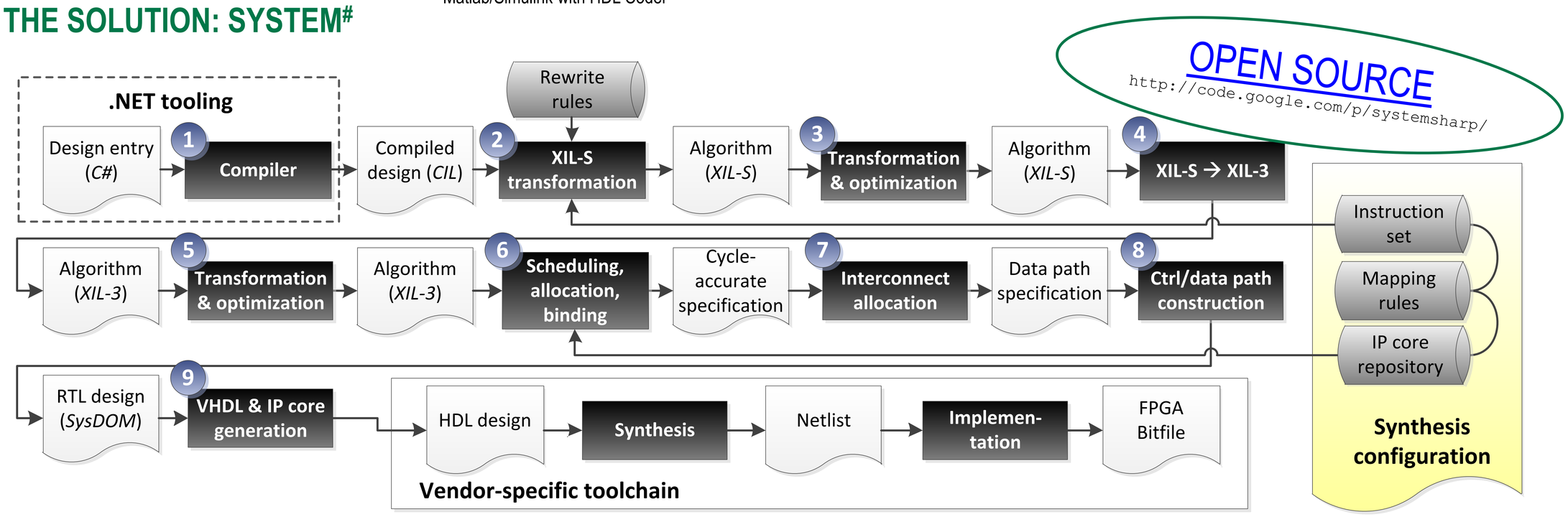
- Mapping such models to FPGAs requires tremendous engineering efforts.
- Can automation simplify the engineering task?

TOOL COMPARISON

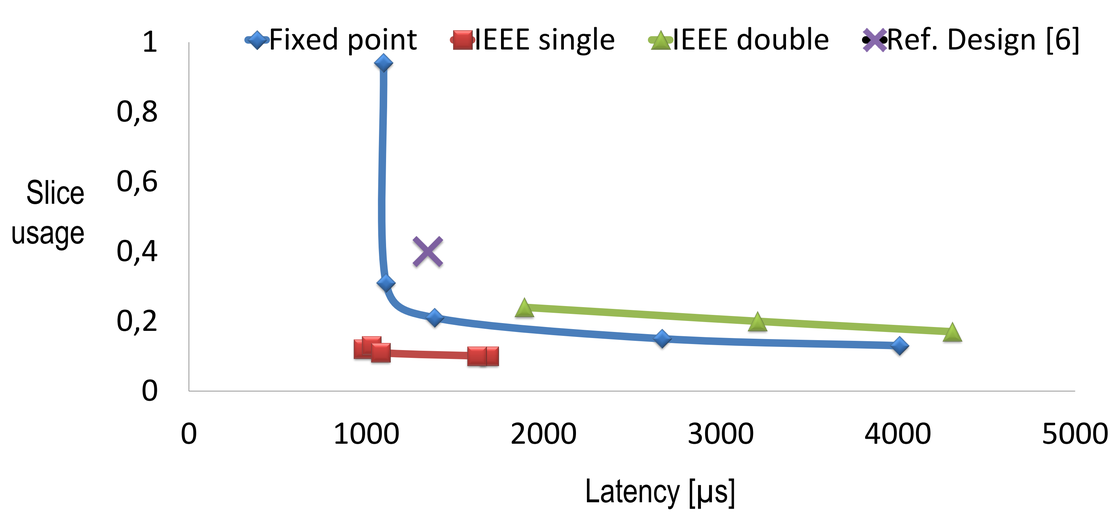
	*Matlab**	*AutoESL*	*DIME-C*	*Trident* [1]	*Legup* [2]	*Gaut* [3]	*Spark* [4]	*Kiwi* [5]	*System#*
Input languages	Proprietary	C, C++, SystemC	C	C	C	C, C++	C	C#	C#
Infers microarchitecture?	✗	✓	✓	✓	✓	✓	✓	✓	✓
Arbitrary-witdth fixed point arithmetic?	✓	✓	✗	✗	✗	✓	✗	✗	✓
Supports floating point arithmetic?	✗	✓	✓	✓	✗	✗	✗	✓	✓
Open source?	✗	✗	✗	✓	✓	✗	✗	✗	✓

*Matlab/Simulink with HDL Coder

THE SOLUTION: SYSTEM#



CASE STUDY: SQUIRREL-CAGE INDUCTION MACHINE



TUNING PARAMETERS

- Arithmetic: 56 bit fixed point, IEEE single/double
- Scheduling: Force-directed scheduling with various schedule lengths
- Xilinx IP cores with various latency settings

CONCLUSIONS

- Broad range of performance/area tradeoff is achievable.
- Quality of hand-coded design was attained.
- Floating-point arithmetic is affordable.

REFERENCES

[1] J. L. Tripp, K. D. Peterson, C. Ahrens, J. D. Poznanovic, and M. Gokhale, “Trident: An fpga compiler framework for floating-point algorithms,” in Proc. Intl. Conf. on Field Programmable Logic and Applications (FPL). IEEE, 2005.

[2] A. Canis, J. Choi, M. Aldham, V. Zhang, A. Kammoona, J. H. Anderson, S. Brown, and T. Czajkowski, “Legup: high-level synthesis for fpga-based processor/accelerator systems,” in Proc. 19th ACM/SIGDA Intl. symposium on Field programmable gate arrays, 2011.

[3] P. Coussy, C. Chavet, P. Bomel, D. Heller, E. Senn, and E. Martin, “GAUT: A High-Level Synthesis Tool for DSP Applications,” in High-Level Synthesis. Springer Netherlands, 2008.

[4] S. Gupta, N. Dutt, R. Gupta, and A. Nicolau, “Spark: a highlevel synthesis framework for applying parallelizing compiler transformations,” in 16th Intl. Conf. on VLSI Design, 2003.

[5] S. Singh and D. J. Greaves, “Kiwi: Synthesis of fpga circuits from parallel programs,” in Proceedings of the 2008 16th International Symposium on Field-Programmable Custom Computing Machines, 2008.

[6] H. Chen, S. Sun, D. Aliprantis, and J. Zambreno, “Dynamic simulation of electric machines on fpga boards,” in Electric Machines and Drives Conference (IEMDC), 2009.

http://code.google.com/p/systemsharp/

YOUR CONTACT

FZI Forschungszentrum Informatik | Embedded Systems and Sensors Engineering
Christian Köllner | koellner@fzi.de | +49 721 9654-166
Mehr Informationen unter www.fzi.de/

STAY IN TOUCH