References

1. ACE Associated Compiler Experts (1998) DSP-C An extension to ISO/IEC IS 9899:1990
2. Adl-Tabatabai AR, Kozyrakis C, Saha B (2006) Unlocking concurrency. Queue 4(10):24–33. doi:10.1145/1189276.1189288
3. Aho AV, Sethi R, Ullman JD (1986) Compilers: principles, techniques, and tools. Addison-Wesley Longman Publishing Co, Boston
4. Akesson B, Molnos A, Hansson A, Angelo JA, Goossens K (2010) Composability and predictability for independent application development, verification, and execution. In: Huebner M, Becker J (eds) Multiprocessor system-on-chip-hardware design and tool integration circuits and systems, chap 2. Springer, Berlin, pp 25–56
5. Amdahl GM (1967) Validity of the single processor approach to achieving large scale computing capabilities. In: Proceedings of the April 18–20, 1967, spring joint computer conference (AFIPS '67). ACM, (Springer), New York, pp 483–485, doi:10.1145/1465482.1465560. http://doi.acm.org/10.1145/1465482.1465560
6. Anderson E, Bai Z, Bischof C, Blackford S, Demmel J, Dongarra J, Du Croz J, Greenbaum A, Hammarling S, McKenney A, Sorensen D (1999) LAPACK users' guide, 3rd edn. Society for industrial and applied mathematics, Philadelphia. http://www.netlib.org/lapack/lug/lapack_lug.html
7. Anne-Francoise G (2011) Synopsys unveils multicore optimization technology. http://www.edn.com/article/519849-Synopsys_unveils_multicore_optimization_technology.php
8. ARM (2010) AHB CPU wrappers technicalr reference manual. http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.set.amba/index.html
9. ARM (2013) ARM926 processor. http://www.arm.com/products/processors/classic/arm9/arm926.php
10. Arnold O, Fettweis G (2010) Power aware heterogeneous MPSoC with dynamic task scheduling and increased data locality for multiple applications. In: 2010 International conference on embedded computer saystems (SAMOS), pp 110–117. doi:10.1109/ICSAMOS.2010.5642075
11. Asanovic K, Bodik R, Catanzaro BC, Gebis JJ, Husbands P, Keutzer K, Patterson DA, Plishker WL, Shalf J, Williams SW, Yelick KA (2006) The landscape of parallel computing research: a view from Berkeley technical report, EECS Department. University of California, Berkeley. http://www.eecs.berkeley.edu/Pubs/TechRpts/2006/EECS-2006-183.pdf
12. Atmel (2011) Diopsis, D940. http://www.atmelroma.it
13. Baert R, Brockmeyer E, Wuytack S, Ashby TJ (2009) Exploring parallelizations of applications for MPSoC platforms using MPA. In: DATE '09: proceedings of the conference on design, automation and test in europe
14. Balakrishnan S, Rajwar R, Upton M, Lai K (2005) The impact of performance asymmetry in emerging multicore architectures. In: Proceedings of the 32nd international symposium on computer architecture (ISCA '05), pp 506–517. doi:10.1109/ISCA.2005.51

J. Castrillón Mazo and R. Leupers, *Programming Heterogeneous MPSoCs*, DOI: 10.1007/978-3-319-00675-8,

15. Balarin F, Watanabe Y, Hsieh H, Lavagno L, Passerone C, Sangiovanni-Vincentelli A (2003) Metropolis: an integrated electronic system design environment. Computer 36(4):45–52. http://dx.doi.org/10.1109/MC.2003.1193228
16. Bamakhrama MA, Zhai JT, Nikolov H, Stefanov T (2012) A methodology for automated design of hard-real-time embedded streaming systems. In: Proceedings of the 15th design, automation, and test in europe conference (DATE 2012)
17. Banerjee U, Eigenmann R, Nicolau A, Padua DA (1993) Automatic program parallelization. Proc IEEE 81(2):211–243
18. Basten T, Hoogerbrugge J (2001) Efficient execution of process networks. In: Chalmers A, Mirmehdi M, Muller H (eds) Communicating process architectures—2001. IOS Press, Amsterdam, pp 1–14
19. Bauer C, Frink A, Kreckel R (2002) Introduction to the GiNaC framework for symbolic computation within the C++ programming language. J Symb Comput 33:1–12. http://dx.doi.org/10.1006/jsco.2001.0494,
20. Bekooij M, Hoes R, Moreira O, Poplavko P, Pastrnak M, Mesman B, Mol JD, Stuijk S, Gheorghita V, van Meerbergen J (2005) Dataflow analysis for real-time embedded multiprocessor system design. In: van der Stok P (ed) Dynamic and robust streaming in and between connected consumer-electronic devices. Springer, Berlin, pp 81–108
21. Benini L, Bertozzi D, Milano M (2008) Resource management policy handling multiple use-cases in MPSoC platforms using constraint programming. In: Proceedings of the 24th international conference on logic programming (ICLP '08). Springer-Verlag, Berlin, pp 470–484, doi:10.1007/978-3-540-89982-2_41. http://dx.doi.org/10.1007/978-3-540-89982-2_41
22. van Berkel CHK (2009) Multi-core for mobile phones. In: Proceedings of the conference on design, automation and test in europe, european design and automation association, 3001 (DATE '09), Belgium, pp 1260–1265. http://www.dl.acm.org/citation.cfm?id=1874620.1874924
23. Bhattacharyya SS, Murthy PK, Lee EA (1999) Synthesis of embedded software from synchronous dataflow specifications. J VLSI Signal Process Syst 21(2):151–166. http://www.dx.doi.org/10.1023/A:1008052406396
24. Bhattacharyya SS, Deprettere EF, Keinert J (2010) Dynamic and multidimensional dataflow graphs. In: Bhattacharyya SS, Deprettere EF, Leupers R, Takala J (eds) Handbook of signal processing systems. Springer, Berlin, pp 899–930
25. Bhattacharyya SS, Deprettere EF, Leupers R, Takala J (2010a) Handbook of signal processing systems, Part II. Springer, Berlin
26. Bhattacharyya SS, Deprettere EF, Leupers R, Takala J (2010b) Handbook of signal processing systems, Part IV. Springer, Berlin
27. Bijlsma T, Bekooij MJG, Smit GJM, Jansen PG (2007) Omphale: streamlining the communication for jobs in a multi processor system on chip. Technical Report TR-CTIT-07-44. Centre for Telematics and Information Technology University of Twente, Enschede
28. Bilsen G, Engels M, Lauwereins R, Peperstraete J (1996) Cyclo-static dataflow. IEEE Trans Signal Process 44(2):397–408. doi:10.1109/78.485935
29. Biscondi E, Flanagan T, Fruth F, Lin Z, Moerman F (2012) Maximizing multicore efficiency with navigator runtime. White Pap. www.ti.com/lit/wp/spry190/spry190.pdf
30. Blair G, Coupaye T, Stefani JB (2009) Component-based architecture: the fractal initiative. Ann Telecommun 64:1–4, doi:10.1007/s12243-009-0086-1. http://dx.doi.org/10.1007/s12243-009-0086-1
31. Borlenghi F, Witte EM, Ascheid G, Meyr H, Burg AP (2011) A 772 Mbit/s 8.81 bit/nJ 90 nm CMOS soft-input soft-output sphere decoder. In: IEEE Asian solid state circuits conference (A-SSCC)
32. Bridges MJ, Vachharajani N, Zhang Y, Jablin T, August DI (2007) Revisiting the sequential programming model for multi-core. In: Proceedings of the 40th annual IEEE/ACM international symposium on microarchitecture. IEEE computer society, Washington, DC, USA, MICRO 40, pp 69–84, Doi:10.1109/MICRO.2007.35. http://liberty.princeton.edu/Publications/micro40_scale.pdf

33. Buck JT (1993) Scheduling dynamic dataflow graphs with bounded memory using the token flow model. PhD thesis, EECS Department. University of California, Berkeley
34. Burger D, Austin TM (1997) The simplescalar tool set, Version 2.0. SIGARCH Comput Archit News 25(3):13–25, doi:10.1145/268806.268810. http://doi.acm.org/10.1145/268806.268810
35. C Meenderinck, A Azevedo, M Alvarez, B Juurlink, and A Ramirez (2008) Parallel scalability of H.264. In: Workshop on programmability issues for multi-core computers (MULTIPROG)
36. Canada CRC (2010) Scari software suite. http://www.crc.gc.ca/en/html/crc/home/research/satcom/rars/sdr/products/scari_suite/scari_suite
37. Carloni L, Bernardinis F, Pinello C, Sangiovanni-Vincentelli A, Sgroi M (2005) Platform-based design for embedded systems. In: Zurawski R (ed) The embedded systems handbook. CRC Press, Boca Raton, p 36
38. Carro L, Rutzig MB (2012) Multi-core systems on chip. In: Bhattacharyya SS, Deprettere EF, Leupers R, Takala J (eds) Handbook of signal processing systems, 2nd edn. Springer, Berlin
39. Carta S, Alimonda A, Pisano A, Acquaviva A, Benini L (2007) A control theoretic approach to energy-efficient pipelined computation in MPSoCs. ACM Trans Embed Comput Syst 6, doi:10.1145/1274858.1274865. url:http://doi.acm.org/10.1145/1274858.1274865
40. Castrillon J, Velasquez R, Stulova A, Sheng W, Ceng J, Leupers R, Ascheid G, Meyr H (2010) Trace-based KPN composability analysis for mapping simultaneous applications to MPSoC platforms. In: Proceedings of the design, automation and test in europe conference and exhibition, Dresden, pp 753–758
41. Castrillon J, Schürmans S, Stulova A, Sheng W, Kempf T, Leupers R, Ascheid G, Meyr H (2011) Component-based waveform development: the nucleus tool flow for efficient and portable software defined radio. Analog Integr Circuits Signal Process 69(2):173–190, doi:10.1007/s10470-011-9670-1. http://dx.doi.org/10.1007/s10470-011-9670-1
42. Castrillon J, Shah A, Murillo L, Leupers R, Ascheid G (2011) Backend for virtual platforms with hardware scheduler in the MAPS framework. In: Proceedings of the 2nd IEEE Latin American symposium on circuits and systems LASCAS'11, IEEE, pp 1–4
43. Castrillon J, Sheng W, Leupers R (2011) Trends in embedded software synthesis. In: Carro L, Pimentel AD (eds) International conference On embedded computer systems: architecture, modeling, and, simulation (SAMOS'11), pp 347–354
44. Castrillon J, Tretter A, Leupers R, Ascheid G (2012) Communication-aware mapping of KPN applications onto heterogeneous MPSoCs. In: Proceedings of the 49th annual conference on design automation (DAC '12)
45. Castrillon J, Leupers R, Ascheid G (2013) MAPS: mapping concurrent dataflow applications to heterogeneous MPSoCs. IEEE Trans. Industr Inf 9(1):527–545, doi:10.1109/TII.2011.2173941. (Online since 10.2011)
46. Castrillon J, Schürmans S, Stulova A, Sheng W, Kempf T, Ishaque A, Leupers R, Ascheid G, Meyr H, (2010) Component-based waveform development: the nucleus tool flow for efficient and portable SDR. In; (2010) Wireless innovation conference and product exposition (SDR'10). Wireless innovation forum, Washington, DC, USA
47. Ceng J (2011) A methodology for efficient multiprocessor system-on-chip software development. PhD thesis, RWTH Aachen Univeristy. Institute for Communication Technologies and Embedded Systems (ICE), Aachen
48. Ceng J, Castrillon J, Sheng W, Scharwächter H, Leupers R, Ascheid G, Meyr H, Isshiki T, Kunieda H (2008) MAPS: an integrated framework for MPSoC application parallelization. In: Proceedings of the 45th annual conference on design automation (DAC '08). ACM, New York, pp 754–759. http://doi.acm.org/10.1145/1391469.1391663
49. Ceng J, Sheng W, Castrillon J, Stulova A, Leupers R, Ascheid G, Meyr H (2009) A high-level virtual platform for early MPSoC software development. In: Proceedings of the 7th IEEE/ACM international conference on hardware/software codesign and system synthesis (CODES+ISSS '09). ACM, New York, pp 11–20. http://doi.acm.org/10.1145/1629435.1629438

50. Chakraborty S, Kunzli S, Thiele L (2003) A general framework for analysing system properties in platform-based embedded system designs. In: Proceedings of the conference on design, automation and test in europe (DATE '03). IEEE Computer Society, Washington, DC, USA, pp 190–195. doi:10.1109/DATE.2003.1253607
51. Chandraiah P, Domer R (2008) Code and data structure partitioning for parallel and flexible MPSoC specification using designer-controlled recoding. IEEE Trans Comput Aided Des Integr Circuits Syst 27(6):1078–1090. doi:10.1109/TCAD.2008.923244
52. Cheung E, Hsieh H, Balarin F (2007) Automatic buffer sizing for rate-constrained KPN applications on multiprocessor system-on-chip. In: Proceedings of the 2007 IEEE international high level design validation and test workshop. IEEE Computer Society, Washington, DC, USA, pp 37–44, doi:10.1109/HLDVT.2007.4392782. http://portal.acm.org/citation.cfm?id=1546679.1546842
53. Choi J, Oh H, Kim S, Ha S (2012) Executing synchronous dataflow graphs on a SPM-based multicore architecture. In: Proceedings of the 49th annual design automation conference (DAC '12). ACM, New York, pp 664–671, doi:10.1145/2228360.2228480. http://doi.acm.org/10.1145/2228360.2228480
54. Clarke P (2011) IHS: embedded wireless market to grow 35 %. http://www.eetimes.com/electronics-news/4215209/IHS-2011-embedded-wireless-market-to-grow-35-, in EETimes
55. Coherent Logix (2013) HyperX development system. http://www.coherentlogix.com/
56. Compaan Design BV (2012) HotSpot parallelizer for C. http://www.compaandesign.com/
57. Cordes D, Marwedel P (2012) Multi-objective aware extraction of task-level parallelism using genetic algorithms. In: Proceedings of design, automation and test in europe (DATE 2012), Dresden
58. Cordes D, Marwedel P, Mallik A (2010) Automatic parallelization of embedded software using hierarchical task graphs and integer linear programming. In: Proceedings of the eighth IEEE/ACM/IFIP international conference on hardware/software codesign and system synthesis (CODES/ISSS '10). ACM, New York, pp 267–276, doi:10.1145/1878961.1879009. http://doi.acm.org/10.1145/1878961.1879009
59. Coussy P, Gajski D, Meredith M, Takach A (2009) An introduction to high-level synthesis. IEEE Des Test Comput 26(4):8–17. doi:10.1109/MDT.2009.69
60. CriticalBlue (2012) Prism: simplifying multicore programming. http://criticalblue.com/prism/index.php
61. Cui Y, Wang Y, Chen Y, Shi Y (2011) Experience on comparison of operating systems scalability on the multi-core architecture. In: 2011 IEEE international conference on cluster computing (CLUSTER), pp 205–215. doi:10.1109/CLUSTER.2011.31
62. Cumming P (2003) The TI OMAP platform approach to SoC. In: Martin G, Chang H (eds) Winning the SoC revolution: experiences in real design, Kluwer, chap 5
63. Davare A (2007) Automated mapping for heterogeneous multiprocessor embedded systems. PhD thesis, EECS Department. University of California, Berkeley. http://www.eecs.berkeley.edu/Pubs/TechRpts/2007/EECS-2007-115.html
64. Davis RI, Burns A (2011) A survey of hard real-time scheduling for multiprocessor systems. ACM Comput Surv (CSUR) 43(4):35:1–35:44, doi:10.1145/1978802.1978814. http://doi.acm.org/10.1145/1978802.1978814
65. Demers A, Keshav S, Shenker S (1989) Analysis and simulation of a fair queueing algorithm. SIGCOMM Comput Commun Rev 19(4):1–12, doi:10.1145/75247.75248. http://doi.acm.org/10.1145/75247.75248
66. Derin O, Diken E, Fiorin L (2011) A Middleware approach to achieving fault-tolerance of Kahn process networks on networks-on-chips. Int J Reconfigurable Comput 2011(Article ID 295385):15pages http://www.downloads.hindawi.com/journals/ijrc/2011/295385.pdf, selected papers from the international workshop on reconfigurable communication-centric systems on chips (ReCoSoC' 2010)
67. Design and Reuse (2010) Blue wonder communications' BWC200 passed first IOT tests against ZTE environment. http://www.designreuse.com/news/23192/lte-ip-iot-tests.html

68. Dongarra JJ, Du Croz J, Hammarling S, Duff IS (1990) A set of level 3 basic linear algebra subprograms. ACM Trans Math Softw 16(1):1–17, doi:10.1145/77626.79170. http://doi.acm.org/10.1145/77626.79170
69. van Dongen SM (2000) Graph clustering by flow simulation. PhD thesis, Faculteit Wiskunde en Informatica. Universiteit Utrecht, Utrecht
70. Dubey P (2005) A platform 2015 workload model: recognition, mining and synthesis moves computers to the Era of Tera. White Pap
71. Dunkels A, Schmidt O, Voigt T, Ali M (2006) Protothreads: simplifying event-driven programming of memory-constrained embedded systems. In: Proceedings of the 4th international conference on embedded networked sensor systems (SenSys '06). ACM, New York, pp 29–42. http://doi.acm.org/10.1145/1182807.1182811
72. Ecker W, Müller W, Dömer R (2008) Hardware-dependent software—principles and practice. In: Ecker W, Müller W, Dömer R (eds) Hardware-dependent software—introduction and overview. Springer, Berlin
73. Eclipse Foundation (2012) Eclipse— an open development platform. http://www.eclipse.org
74. Edwards S, Tardieu O (2006) SHIM: a deterministic model for heterogeneous embedded systems. IEEE Trans VLSI Syst 14(8):854–867. doi:10.1109/TVLSI.2006.878473
75. EE Times (2007) Embedded software Stuck at C. http://www.eetimes.com/news/design/$
76. Eker J, Janneck J, Lee E, Liu J, Liu X, Ludvig J, Neuendorffer S, Sachs S, Xiong Y (2003) Taming heterogeneity—the ptolemy approach. Proc IEEE 91(1):127–144. doi:10.1109/JPROC.2002.805829
77. Electronic Systems Group (2012) CoMPSoC project. http://compsoc.eu/
78. Electronic Systems Group (2013) SDF^3: SDF for free. http://www.es.ele.tue.nl/sdf3/
79. Erbas C, Cerav-Erbas S, Pimentel A (2006) Multiobjective optimization and evolutionary algorithms for the application mapping problem in multiprocessor system-on-chip design. IEEE Trans Evol Comput 10(3):358–374. doi:10.1109/TEVC.2005.860766
80. Ester M, peter Kriegel H, S J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of 2nd international conference on knowledge discovery and data mining. AAAI Press, New York, pp 226–231
81. EURETILE Consortium (2013) European reference tiled architecture experiment (euretile). http://euretile.roma1.infn.it/mediawiki/index.php/Main_Page
82. Falk J, Keinert J, Haubelt C, Teich J, Bhattacharyya SS (2008) A generalized static data flow clustering algorithm for MPSoC scheduling of multimedia applications. In: Proceedings of the 8th ACM international conference on embedded software (EMSOFT '08). ACM, New York, pp 189–198, doi:10.1145/1450058.1450084. http://doi.acm.org/10.1145/1450058.1450084
83. Falk J, Zebelein C, Haubelt C, Teich J (2011) A rule-based static dataflow clustering algorithm for efficient embedded software synthesis. In: Proceedings of IEEE computer society (DATE'11), pp 14–18
84. Fiorin L, Ferrante A, Padarnitsas K, Carucci S (2010) Hardware-assisted security enhanced linux in embedded systems: a Proposal. In: Proceedings of the 5th workshop on embedded systems security (WESS '10). ACM, New York, pp 3:1–3:7, doi:10.1145/1873548.1873551. http://doi.acm.org/10.1145/1873548.1873551
85. Fisher J, Farabosch P, Young C (2005) Embedded computing: a VLIW approach to architecture compilers and tools. Morgan-Kaufmann (Elsevier), Burlington
86. Fisher JA, Faraboschi P, Young C (2004) Embedded computing : a VLIW approach to architecture compilers and tools. Morgan Kaufmann, Burlington
87. Flynn MJ (1972) Some computer organizations and their effectiveness. IEEE Trans Comput C-21(9):948–960. doi:10.1109/TC.1972.5009071
88. Gao L, Huang J, Ceng J, Leupers R, Ascheid G, Meyr H (2009) TotalProf: a fast and accurate retargetable source code profiler. In: Proceedings of the 7th IEEE/ACM international conference on hardware/software codesign and system synthesis (CODES+ISSS '09). ACM, New York, pp 305–314. http://doi.acm.org/10.1145/1629435.1629477
89. Garey MR, Johnson DS (1990) Computers and intractability. A guide to the theory of NP-completeness. W. H. Freeman and Co. New York

90. Gartner (2011) Gartner says western europe PC market declined 19 percent in second quarter of 2011. http://www.gartner.com/it/page.jsp?id=1769215
91. Geilen M, Basten T (2003) Requirements on the execution of Kahn process networks. In: Proceedings of the 12th european symposium on programming (ESOP 2003). Springer Verlag, Berlin, pp 319–334
92. Geilen M, Tripakis S, Wiggers M (2010) The earlier the better: a theory of timed actor interfaces. Technical Report UCB/EECS-2010-130, EECS Department. University of California, Berkeley. http://www.eecs.berkeley.edu/Pubs/TechRpts/2010/EECS-2010-130.html
93. Gelernter D, Carriero N (1992) Coordination languages and their significance. Commun ACM 35(2):97–107, doi:10.1145/129630.129635. http://doi.acm.org/10.1145/129630.129635
94. Geuns S, Bekooij M, Bijlsma T, Corporaal H (2011) Parallelization of while loops in nested loop programs for shared-memory multiprocessor systems. In: Design automation test in europe conference exhibition (DATE), pp 1–6
95. Gheorghita S, T Basten Corpoaal H (2006) An overview of application scenario usage in streaming-oriented embedded system design
96. Ghosal A, Limaye R, Ravindran K, Tripakis S, Prasad A, Wang G, Tran TN, Andrade H (2012) Static dataflow with access patterns: semantics and analysis. In: Proceedings of the 49th annual design automation conference (DAC '12). ACM, New York, pp 656–663, doi:10.1145/2228360.2228479. http://doi.acm.org/10.1145/2228360.2228479
97. Girkar M, Polychronopoulos C (1992) Automatic extraction of functional parallelism from ordinary programs. IEEE Trans Parallel Distrib Syst 3(2):166–178. doi:10.1109/71.127258
98. Girkar M, Polychronopoulos CD (1995) Extracting task-level parallelism. ACM Trans Program Lang Syst 17(4):600–634. http://doi.acm.org/10.1145/210184.210189
99. GNU Project (2012) GDB: The GNU project debugger. http://www.gnu.org/software/gdb/
100. Goyal P, Guo X, Vin HM (1996) A hierarchical CPU scheduler for multimedia operating systems. In: Proceedings, 2nd symposium on operating systems design and implementations (OSDI'96). Seattle, Washington, pp 107–122
101. Graham SL, Kessler PB, Mckusick MK (1982) Gprof: a call graph execution profiler. SIGPLAN Not 17(6):120–126, doi:10.1145/872726.806987. http://doi.acm.org/10.1145/872726.806987
102. Group OS (2013) Open64. http://www.open64.net/
103. Haene S (2008) VLSI circuits for MIMO-OFDM physical layer. ETH Zurich, Zurich
104. Haid W, Keller M, Huang K, Bacivarov I, Thiele L (2009) Generation and calibration of compositional performance analysis models for multi-processor systems. In: Proceedings of SAMOS'09. IEEE Press, Thessaloniki, pp 92–99
105. Hall MH, Amarasinghe SP, Murphy BR, Liao SW, Lam MS (1995) Detecting coarse-grain parallelism using an interprocedural parallelizing compiler. In: Proceedings of the 1995 ACM/IEEE conference on Supercomputing (CDROM) Supercomputing '95. ACM, New York, doi:10.1145/224170.224337. http://doi.acm.org/10.1145/224170.224337
106. Hankins RA, Chinya GN, Collins JD, Wang PH, Rakvic R, Wang H, Shen JP (2006) Multiple instruction stream processor. SIGARCH Comput Archit News 34(2):114–127. http://doi.acm.org/10.1145/1150019.1136495
107. Hansson A, Goossens K, Bekooij M, Huisken J (2009) CoMPSoC: a template for composable and predictable multi-processor system on chips. ACM Trans Des Autom Electron Syst 14(1):1–24. http://doi.acm.org/10.1145/1455229.1455231
108. Harriss T, Walke R, Kienhuis B, Deprettere E (2002) Compilation from matlab to process networks realized in FPGA. Des Autom Embed Syst 7(4):656-679. http://doi.acm.org/10.1145/1146909.1146981
109. Haubelt C, Falk J, Keinert J, Schlichter T, Streubühr M, Deyhle A, Hadert A, Teich J (2007) A systemC-based design methodology for digital signal processing systems. EURASIP J Embed Syst 2007(1):22. http://dx.doi.org/10.1155/2007/47580
110. Hennessy J (1999) The future of systems research. Computer 32(8):27–33. doi:10.1109/2.781631

111. Hind M (2001) Pointer analysis: haven't we solved this problem Yet? In: Proceedings of the 2001 ACM SIGPLAN-SIGSOFT workshop on program analysis for software tools and engineering (PASTE '01). ACM, New York, pp 54–61. http://doi.acm.org/10.1145/379605.379665
112. Hoare CAR (1978) Communicating sequential processes. Commun ACM 21(8):666–677. doi:10.1145/359576.359585
113. Hoare CAR (2004) Communicating sequential processes. Prentice Hall Int, Upper Saddle River
114. Howard J, Dighe S, Hoskote Y, Vangal S, Finan D, Ruhl G, Jenkins D, Wilson H, Borkar N, Schrom G, Pailet F, Jain S, Jacob T, Yada S, Marella S, Salihundam P, Erraguntla V, Konow M, Riepen M, Droege G, Lindemann J, Gries M, Apel T, Henriss K, Lund-Larsen T, Steibl S, Borkar S, De V, Van Der Wijngaart R, Mattson T (2010) A 48-core IA-32 message-passing processor with DVFS in 45nm CMOS. In: 2010 IEEE international on solid-state circuits conference digest of technical papers (ISSCC), pp 108–109. doi:10.1109/ISSCC.2010.5434077
115. Huang K, Bacivarov I, Liu J, Haid W (2009) A modular fast simulation framework for stream-oriented MPSoC. In: IEEE symposium on industrial embedded systems (SIES), IEEE, Ecole Polytechnique Fédérale de Lausanne. Switzerland, pp 74–81. doi:10.1109/SIES.2009.5196198
116. Huang K, Haid W, Bacivarov I, Keller M, Thiele L (2012) Embedding formal performance analysis into the design cycle of MPSoCs for real-time streaming applications. ACM Transactions in Embedded Computing Systems (TECS), New York
117. Hwu W-M, Ryoo S, Ueng SZ, Kelm JH, Gelado I, Stone SS, Kidd RE, Baghsorkhi SS, Mahesri AA, Tsao SC, Navarro N, Lumetta SS, Frank MI, Patel SJ (2007) Implicitly parallel programming models for thousand-core microprocessors. In: Proceedings of the 44th annual conference on design automation (DAC '07). ACM, New York, pp 754–759. http://doi.acm.org/10.1145/1278480.1278669
118. IEEE (2005) Open systemC language reference manual
119. IEEE (2010) IEEE standard for IP-XACT, standard structure for packaging, integrating, and reusing IP within tool flows. In: IEEE computer society and the IEEE standards association corporate advisory group
120. Ienne P, Leupers R (2006) Customizable embedded processors: design technologies and applications (Systems on silicon). Morgan Kaufmann Publishers Inc, Burlington
121. IMEC (2012) CleanC analysis tools. http://www.imec.be/cleanc/Welcome.html
122. International Technology Roadmap for Semiconductors (ITRS) (2005) Design. http://www.itrs.net/
123. International Technology Roadmap for Semiconductors (ITRS) (2011) Design. http://www.itrs.net/
124. International Technology Roadmap for Semiconductors (ITRS) (2011) System drivers. http://www.itrs.net/
125. International Technology Roadmap for Semiconductors (ITRS) (2013). http://www.itrs.net/
126. ISO/IEC (1996) Information technology—open systems interconnection—basic reference model: the basic model. International standard ISO-IEC 7498–1
127. Jain AK, Murty MN, Flynn PJ (1999) Data clustering: a review. ACM Comput Surv 31(3):264–323. http://doi.acm.org/10.1145/331499.331504
128. Jantsch A, Sander I (2005) Models of computation and languages for embedded system design. IEE Proc Comput Digital Tech 152(2):114–129. doi:10.1049/ip-cdt:20045098
129. Java Community Process (2013) Real-time specification for Java. http://www.rtsj.org/
130. Javaid H, Shafique M, Parameswaran S, Henkel J (2011) Low-power adaptive pipelined MPSoCs for multimedia: an H.264 video encoder case study. In: 48th ACM/EDAC/IEEE design automation conference (DAC), pp 1032–sa1037
131. Johnson R, Pingali K (1993) Dependence-based program analysis. In: Conference on programming language design and implementation, pp 78–89. http://citeseer.ist.psu.edu/johnson93dependencebased.html

132. Joint Tactical Networking Center (JTNC) (2013) JTRS standards. http://jtnc.mil/sca/Pages/sca1.aspx
133. Jones D, Topham N (2009) High speed CPU simulation using LTU dynamic binary translation. In: Proceedings of the 4th international conference on high performance embedded architectures and compilers (HiPEAC '09). Springer-Verlag, Berlin, pp 50–64, doi:10.1007/978-3-540-92990-1_6. http://dx.doi.org/10.1007/978-3-540-92990-1_6
134. JTRS Standards Joint Program Executive Office (JPEO) Joint Tactical Radio System (JTRS) (2011) Software communications architecture specification. Version Next (Draft), San Diego. http://www.public.navy.mil/jpeojtrs/sca/Pages/scanext.aspx
135. Kahn G (1974) The semantics of a simple language for parallel programming. In: Rosenfeld JL (ed) Information processing '74: proceedings of the IFIP congress, New York, pp 471–475
136. Kanda W, Yumura Y, Kinebuchi Y, Makijima K, Nakajima T (2008) SPUMONE: lightweight CPU virtualization layer for embedded systems. In: IEEE/IFIP international conference on embedded and ubiquitous computing (EUC '08), vol 1. pp 144–151. doi:10.1109/EUC.2008.157
137. Karam L, AlKamal I, Gatherer A, Frantz G, Anderson D, Evans B (2009) Trends in multicore DSP platforms. IEEE Signal Process Mag 26(6):38–49. doi:10.1109/MSP.2009.934113
138. Karkowski I, Corporaal H (1997) Design of heterogenous multi-processor embedded systems: applying functional pipelining. In: Proceedings of the 1997 international conference on parallel architectures and compilation techniques (PACT '97). IEEE Computer Society, Washington, DC, USA, p 156
139. Karp RM, Miller RE (1966) Properties of a model for parallel computations: determinacy, termination, queuing. SIAM J Appl Math 14(6)
140. Karuri K, Al Faruque MA, Kraemer S, Leupers R, Ascheid G, Meyr H (2005) Fine-grained application source code profiling for ASIP design. In: Proceedings of the 42nd annual design automation conference. ACM, New York, pp 329–334. http://doi.acm.org/10.1145/1065579.1065666
141. Keinert J, Streubühr M, Schlichter T, Falk J, Gladigau J, Haubelt C, Teich J, Meredith M (2009) SystemCodesigner—an automatic ESL synthesis approach by design space exploration and behavioral synthesis for streaming applications. ACM Trans Des Autom Electron Syst 14:1:1–1:23, doi:10.1145/1455229.1455230. http://www.doi.acm.org/10.1145/1455229.1455230
142. Kempf T, Guenther D, Ishaque A, Ascheid G (2011) MIMO OFDM transceiver for a many-core computing fabric—a nucleus based implementation. In: The wireless innovation forum conference on communications technologies and software defined radio (SDR'11). Washington, DC, USA
143. Kempf T, Witte EM, Ramakrishnan V, Ascheid G, Adrat M, Antweiler M, (2008) A practical view on SDR baseband processing portability. In: 2008 Software defined radio technical conference (SDR'08). Washington DC, USA
144. Kennedy K, Allen JR (2002) Optimizing compilers for modern architectures: a dependence-based approach. Morgan Kaufmann Publishers Inc. San Francisco
145. Khronos Group (2013) OpenCL—the open standard for parallel programming of heterogeneous systems. http://www.khronos.org/opencl/
146. Kienhuis B, Deprettere E, Vissers K, Van Der Wolf P (1997) An approach for quantitative analysis of application-specific dataflow architectures. In: Proceedings IEEE international conference on application-specific systems, architectures and processors, pp 338–349. doi:10.1109/ASAP.1997.606839
147. Kogel T, Leupers R, Meyr H (2006) Integrated system-level modeling of network-on-chip enabled multi-processor platforms. Springer-Verlag New York Inc. Secaucus
148. Kogel T, Meyr H (2004) Heterogeneous MP-SoC—the solution to energy-efficient signal processing. In: Design automation conference (DAC). San Diego
149. Kohout P, Ganesh B, Jacob B (2003) Hardware support for real-time operating systems. In: Proceedings of the 1st IEEE/ACM/IFIP international conference on Hardware/software codesign and system synthesis (CODES+ISSS '03). ACM, New York, pp 45–51. http://www.doi.acm.org/10.1145/944645.944656

150. Kopetz H (2011) Real-time systems: design principles for distributed embedded applications, 2nd edn. Springer, New York
151. Kreher DL, Stinson DR (1998) Combinatorial algorithms: generation, enumeration, and search (Discrete Mathematics and Its Applications). 1 edn. CRC Press, ISBN-13: 978–0849339882
152. Krishnan V, Torrellas J (1999) A chip-multiprocessor architecture with speculative multithreading. IEEE Trans Comput 48(9):866–880
153. Kulkarni M, Pingali K, Walter B, Ramanarayanan G, Bala K, Chew LP (2007) Optimistic parallelism requires abstractions. SIGPLAN Not 42:211–222, doi:10.1145/1273442.1250759. http://www.doi.acm.org/10.1145/1273442.1250759
154. Kumar A, Fernando S, Ha Y, Mesman B, Corporaal H (2008) Multiprocessor systems synthesis for multiple use-cases of multiple applications on FPGA. ACM Trans Des Autom Electron Syst 13(3):1–27. http://doi.acm.org/10.1145/1367045.1367049
155. Kumar A, Mesman B, Theelen B, Corporaal H, Ha Y (2008) Analyzing composability of applications on MPSoC platforms. J Syst Archit 54(3–4):369–383. http://dx.doi.org/10.1016/j.sysarc.2007.10.002
156. Kumar S, Hughes CJ, Nguyen A (2007) Carbon: architectural support for fine-grained parallelism on chip multiprocessors. SIGARCH Comput Archit News 35(2):162–173. http://doi.acm.org/10.1145/1273440.1250683
157. Kwok YK, Ahmad I (1999) Static scheduling algorithms for allocating directed task graphs to multiprocessors. ACM Comput Surv 31(4):406–471. http://doi.acm.org/10.1145/344588.344618
158. Lab K (2013) METIS—serial graph partitioning and fill-reducing matrix ordering. http://glaros.dtc.umn.edu/gkhome/metis/metis/publications
159. Lam M (1988) Software pipelining: an effective scheduling technique for VLIW machines. SIGPLAN Not 23(7):318–328. http://doi.acm.org/10.1145/960116.54022
160. Lattner C (2008) LLVM and clang: next generation compiler technology. The BSD Conference, Ottawa
161. Lattner C, Adve V (2004) LLVM: a compilation framework for lifelong program analysis and transformation. In: Proceedings of the international symposium on code generation and optimization (CGO '04). IEEE Computer Society, Washington, DC, USA, p 75
162. Lattner C, Lenharth A, Adve V (2007) Making context-sensitive points-to analysis withheap cloning practicalf for the real world. In: Proceedings of the 2007 ACM SIGPLANconference on programming language design and implementation (PLDI'07), San Diego
163. Lee E, Parks T (1995) Dataflow process networks. Proc IEEE 83(5):773–801. doi:10.1109/5.381846
164. Lee EA (1991) Consistency in dataflow graphs. IEEE Trans Parallel Distrib Syst 2(2):223–235
165. Lee EA (2006) The problem with threads. Computer 39(5):33–42. http://portal.acm.org/citation.cfm?id=1137232.1137289
166. Lee EA, Messerschmitt DG (1987) Synchronous data flow. Proc IEEE 75(9):1235–1245
167. Lee EA, Sangiovanni-Vincentelli A (1996) Comparing models of computation. In: Proceedings of the IEEE/ACM international conference on computer-aided design (ICCAD '96). IEEE Computer Society, Washington, DC, USA, pp 234–241
168. Lee EA, Sangiovanni-Vincentelli A (1998) A framework for comparing models of computation. IEEE Trans Comput Aided Design Integr Circuits Syst 17(12):1217–1229
169. Lee J, Mooney VJ III, Daleby A, Ingström K, Klevin T, Lindh L (2003) A Comparison of the RTU hardware RTOS with a hardware/software RTOS. In: Proceedings of the 2003 conference on Asia south pacific design automation (ASPDAC). ACM, New York, pp 683–688. http://doi.acm.org/10.1145/1119772.1119925
170. Lehtoranta O, Hamalainen T (2003) Complexity analysis of spatially scalable MPEG-4 encoder. In: Proceedings international symposium on system-on-chip, pp 57–60. doi:10.1109/ISSOC.2003.1267717
171. Leupers R (2000) Code selection for media processors with simd instructions. In: DATE '00. ACM, New York, pp 4–8. http://www.doi.acm.org/10.1145/343647.343679

172. Leupers R (2011) LANCE retargetable C compiler. http://www.lancecompiler.com
173. Leupers R, Castrillon J (2010) MPSoC programming using the MAPS compiler. In: Proceedings of the 15th Asia and south pacific design automation conference (ASP-DAC '10), pp 897–902
174. Levy HM (1984) Capability-based computer systems. Butterworth-Heinemann, Chap Intel iAPX 432:159–186
175. Liao CH, Lai IW, Nikitopoulos K, Borlenghi F, Kammler D, Witte M, Zhang D, Chiueh TD, Ascheid G, Meyr H (2009) Combining orthogonalized partial metrics: efficient enumeration for soft-input sphere decoder. In: IEEE 20th international symposium on personal, indoor and mobile radio communications, pp 1287–1291. doi:10.1109/PIMRC.2009.5450104
176. Limberg T, Ristau B, Fettweis G (2008) A real-time programming model for heterogeneous MPSoCs. Chap embedded computer systems: architectures, modeling, and, simulation. Springer, Berlin, pp 75–84
177. Limberg T, Winter M, Bimberg M, Klemm R, Matus E, Tavares M, Fettweis G, Ahlendorf H, Robelly P (2008) A fully programmable 40 GOPS SDR single chip baseband for LTE/WiMAX terminals. In: 34th European solid-state circuits conference (ESSCIRC 2008), pp 466–469. doi:10.1109/ESSCIRC.2008.4681893
178. Lippett M (2004) An IP core based approach to the on-chip management of heterogeneous SoCs. IP/SOC 2004
179. Liu CL, Layland JW (1973) Scheduling algorithms for multiprogramming in a hard-real-time environment. J ACM 20:46–61
180. Lyrtech Inc (2013) SFF SDR DP API guide. 1st edn. http://www.ceanet.com.au/Products/Lyrtech/SFFSDRDevelopment.aspx
181. Ma Z, Marchal P, Scarpazza DP, Yang P, Wong C, Gmez JI, Himpe S, Ykman-Couvreur C, Catthoor F (2007) Systematic methodology for real-time cost-effective mapping of dynamic concurrent task-based systems on heterogenous platforms. Springer Publishing Company Incorporated, New York
182. Martin G (2005) ESL requirements for configurable processor-based embedded system design. In: IP-SoC 2005, pp 15–20
183. Martin G (2006) Overview of the MPSoC design challenge. In: 43rd ACM/IEEE design automation conference, pp 274–279. doi:10.1109/DAC.2006.229245
184. MathWorks (2013) Simulink—simulation and model-based design. http://www.mathworks.com/products/simulink/
185. McKeown M (2010) FFT implementation on the TMS320VC5505, TMS320C5505, and TMS320C5515 DSPs. Application report. Texas instruments, (SPRABB6A)
186. Meijer S, Nikolov H, Stefanov T (2010) Combining process splitting and merging transformations for polyhedral process networks. In: 8th IEEE workshop on embedded systems for real-time multimedia (ESTIMedia), pp 97–106. doi:10.1109/ESTMED.2010.5666985
187. Meijer S, Nikolov H, Stefanov T (2010) Throughput modeling to evaluate process merging transformations in polyhedral process networks. In: 13th International conference design, automation and test in, europe (DATE'10), pp 747–752
188. Meyr H, Moeneclaey M, Fechtel S (1997) Digital communication receivers: synchronization, channel estimation, and signal processing. Wiley, New York
189. Mignolet JY, Baert R, Ashby TJ, Avasare P, Jang HO, Son JC (2009) MPA: parallelizing an application onto a multicore platform made easy. IEEE Micro 29(3):31–39. http://www.doi.ieeecomputersociety.org/10.1109/MM.2009.46
190. Miyamori T (2007) MPSoC architecture trade-offs for multimedia applications. In: 7th International forum on application-specific multi-processor SoC (MPSoC'07), pp 336–337
191. Moorby P, Kelf D, Lin Y, (2010) Open vector radio, a C dialect standard proposal for high performance software baseband coding. In: 2010 Wireless innovation conference and product exposition (SDR'10). Wireless Innovation Forum, Washington, DC, USA
192. Moreira O, Valente F, Bekooij M (2007) Scheduling multiple independent hard-real-time jobs on a heterogeneous multiprocessor. In: Proceedings of the 7th ACM and IEEE international conference on embedded software (EMSOFT '07). ACM, New York, pp 57–66. http://www.doi.acm.org/10.1145/1289927.1289941

193. Muchnick SS (1997) Advanced compiler design and implementation. Morgan Kaufmann Publishers Inc, San Francisco
194. Murtaza Z, Khan S, Rafique A, Bajwa K, Zaman U (2006) Silicon real time operating system for embedded DSPs. In: Proceedings of the international conference on emerging technologies (ICET '06), pp 188–191. doi:10.1109/ICET.2006.336032
195. Nácul AC, Regazzoni F, Lajolo M (2007) Hardware scheduling support in SMP architectures. In Proceedings of the conference on design, automation and test in Europe, EDA consortium (DATE '07). San Jose, pp 642–647
196. Nadezhkin D, Stefanov T (2011) Automatic derivation of polyhedral process networks from while-loop affine programs. In: 9th IEEE symposium on embedded systems for real-time multimedia (ESTIMedia), pp 102–111. doi:10.1109/ESTIMedia.2011.6088516
197. Nadezhkin D, Nikolov H, Stefanov T (2010) Translating affine nested-loop programs with dynamic loop bounds into polyhedral process networks. In: 8th IEEE Workshop on embedded systems for real-time multimedia (ESTIMedia), pp 21–30. doi:10.1109/ESTMED.2010.5666977
198. Nakano T, Utama A, Itabashi M, Shiomi A, Imai M (1995) Hardware implementation of a real-time operating system. In: Proceedings of the 12th TRON project international symposium (TRON '95). IEEE Computer Society, Washington, DC, USA, p 34
199. Nass R (2008) An Insider's view of the 2008 embedded market study. http://eetimes.com/design/embedded/4007664/An-insider-s-view-of-the-2008-Embedd ed-Market-Study, in EETimes
200. National Instruments (2013) LabVIEW system design software. http://www.ni.com/labview/
201. Nichols B, Buttlar D, Farrell JP (1996) Pthreads programming. O'Reilly and Associates Inc. Sebastopol
202. Nikolov H (2009) System-level design methodology for streaming multi-processor embedded systems. PhD thesis Universiteit Leiden
203. Nohl A, Braun G, Schliebusch O, Leupers R, Meyr H, Hoffmann A (2002) A universal technique for fast and flexible instruction-set architecture simulation. In: Proceedings of the 39th annual design automation conference (DAC '02). ACM, New York, pp 22–27, doi:10.1145/513918.513927. http://doi.acm.org/10.1145/513918.513927
204. NVIDIA (2013) CUDA: parallel programming made easey. http://www.nvidia.com/object/cuda_home_new.html
205. Odendahl M, Sheng W, Aguilar M, Leupers R, Ascheid G (2012) Automated code generation of streaming applications for C6000 multicore DSPs. In: 5th European DSP education and research conference
206. Olivier BE (2010) Embedded C for Digital Signal Processing. In: Bhattacharyya SS, Deprettere EF, Leupers R, Takala J (eds) . Handbook of signal processing systems. Springer, Berlin, pp 769–787
207. Ottoni G, Rangan R, Stoler A, August DI (2005) Automatic thread extraction with decoupled software pipelining. In: Proceedings of the 38th annual IEEE/ACM international symposium on microarchitecture (MICRO 38). IEEE Computer Society, Washington, DC, USA, pp 105–118
208. Owens JD, Luebke D, Govindaraju N, Harris M, Krüger J, Lefohn AE, Purcell TJ (2007) A survey of general-purpose computation on graphics hardware. Comput Graph Forum 26(1):80–113
209. Park S, Hong DS, Chae SI (2008) A hardware operating system kernel for multi-processor systems. IEICE Electrons Express 5(9):296–302
210. Parks TM (1995) Bounded scheduling of process networks. PhD thesis, EECS Department, University of California, Berkeley
211. Patel NK, Shelby KA, Dalio BA, (2010) Radio waveform development system providing an integrated approach to SDR waveform design and implementation. In: Wireless innovation conference and product exposition (SDR'10). Wireless Innovation Forum, Washington, DC, USA

212. Paulin P (2011) Programming challenges and solutions for multi-processor SoCs: an industrial perspective. In: 48th ACM/EDAC/IEEE design automation conference (DAC), pp 262–267
213. Paulin P, Pilkington C, Langevin M, Bensoudane E, Lyonnard D, Benny O, Lavigueur B, Lo D, Beltrame G, Gagne V, Nicolescu G (2006) Parallel programming models for a multiprocessor SoC platform applied to networking and multimedia. Very large scale integration (VLSI) systems. IEEE Trans 14(7):667–680. doi:10.1109/TVLSI.2006.878259
214. Pham D, Asano S, Bolliger M, Day M, Hofstee H, Johns C, Kahle J, Kameyama A, Keaty J, Masubuchi Y, Riley M, Shippy D, Stasiak D, Suzuoki M, Wang M, Warnock J, Weitzel S, Wendel D, Yamazaki T, Yazawa K (2005) The design and implementation of a first-generation CELL processor. In: IEEE international solid-state circuits conference on digest of technical papers (ISSCC), vol 1. pp 184–592. doi:10.1109/ISSCC.2005.1493930
215. Pimentel A, Erbas C, Polstra S (2006) A systematic approach to exploring embedded system architectures at multiple abstraction levels. IEEE Trans Comput 55(2):99–112. http://dx.doi.org/10.1109/TC.2006.16
216. Plishker W, Sane N, Kiemb M, Anand K, Bhattacharyya S (2008) Functional DIF for rapid prototyping. In: The 19th IEEE/IFIP international symposium on rapid system prototyping (RSP '08), pp 17–23. doi:10.1109/RSP.2008.32
217. Plishker W, Sane N, Bhattacharyya SS (2009) A generalized scheduling approach for dynamic dataflow applications. In: Proceedings of DATE'09, pp 111–116
218. PrismTech Ltd (2011) Spectra CX—the SCA development tool. www.prismtechnologies.com
219. Püschel M, Moura J, Johnson J, Padua D, Veloso M, Singer B, Xiong J, Franchetti F, Gacic A, Voronenko Y, Chen K, Johnson R, Rizzolo N (2005) SPIRAL: code generation for DSP transforms. Proc IEEE 93(2):232–275. doi:10.1109/JPROC.2004.840306
220. Qualcomm (2011) Snapdragon S4 processors: system on chip solutions for a new mobile age. White Pap. https://developer.qualcomm.com/download
221. Qualcomm (2013) Qualcomm snapdragon processors. http://www.qualcomm.eu/products/snapdragon
222. Ramakrishnan, V, Witte, EM, Kempf, T, Kammler, D, Ascheid, G and H Meyr, Adrat, M and M Antweiler (2009) Efficient and portable SDR waveform development: the nucleus concept. In: IEEE military communications conference (MILCOM 2009). Boston
223. Ramey C (2011) TILE-Gx100 manyCore processor: acceleration interfaces and architecture. Presented at HotChips 23
224. Reid AD, Flautner K, Grimley-Evans E, Lin Y (2008) SoC-C: efficient programming abstractions for heterogeneous multicore systems on chip. In: Proceedings of the 2008 international conference on compilers, architectures and synthesis for embedded systems (CASES '08). ACM, New York, pp 95–104, doi:10.1145/1450095.1450112. http://www.doi.acm.org/10.1145/1450095.1450112
225. Rik Myslewski (2011) ARM vet: the CPU's future is threatened. http://www.theregister.co.uk/2011/08/20/microprocessors_may_face_trouble_ahead/
226. Robin Saxby (2006) Don't just survive, thrive. Presentation: 9th Annual Semico Summit, Scottsdale
227. Rul S, Vandierendonck H, De Bosschere K (2010) A profile-based tool for finding pipeline parallelism in sequential programs. Parallel Comput 36:531–551, doi:10.1016/j.parco.2010.05.006. http://www.dx.doi.org/10.1016/j.parco.2010.05.006
228. Sangiovanni-Vincentelli A, Martin G (2001) Platform-based design and software design methodology for embedded systems. IEEE Design Test Comput 18(6):23–33. doi:10.1109/54.970421
229. Sarkar V (1991) Automatic partitioning of a program dependence graph into parallel tasks. IBM J Res Dev 35(5.6):779–804. doi:10.1147/rd.355.0779
230. Schaeffer SE (2007) Graph clustering. Comput Sci Rev 1(1):27–64, doi:10.1016/j.cosrev.2007.05.001. http://www.sciencedirect.com/science/article/pii/S1574013707000020
231. Seidel H (2006) A task-level programmable processor. PhD thesis, Duisburg
232. Shabbir A, Kumar A, Stuijk S, Mesman B, Corporaal H (2010) CA-MPSoC: an automated design flow for predictable multi-processor architectures for multiple applications. J Syst

Archit EUROMICRO J 56(7):265–277, doi:10.1016/j.sysarc.2010.03.007. http://dx.doi.org/10.1016/j.sysarc.2010.03.007
233. Shen CC, Wu HH, Sane N, Plishker W, Bhattacharyya SS (2011) A design tool for efficient mapping of multimedia applications onto heterogeneous platforms. In: IEEE International conference on multimedia and expo (ICME), pp 1–6. doi:10.1109/ICME.2011.6011952
234. Shen CC, Plishker W, Wu HH, Bhattacharyya SS, (2010) A lightweight dataflow approach for design and implementation of SDR systems. In Wireless innovation conference and product exposition (SDR'10), Washington, DC, USA
235. Sheng W, Wiebe A, Stulova A, Leupers R, Kienhuis B, Walters J, Ascheid G (2012) FIFO exploration in mapping streaming applications onto the TI OMAP3530 platform: case study and optimizations. In: IEEE 6th international symposium on embedded eulticore SoCs (MCSoC-12)
236. Sheng W, Schürmans S, Odendahl M, Leupers R, Ascheid G (2013) Automatic calibration of streaming applications for software mapping exploration. IEEE Des Test Comput
237. Sih GC, Lee EA (1993) A compile-time scheduling heuristic for interconnection-constrained heterogeneous processor architectures. IEEE Trans Parallel Distrib Syst 4(2):175–187. http://dx.doi.org/10.1109/71.207593
238. Snir M, Otto S (1998) MPI-the complete reference: the MPI core. MIT Press, Cambridge
239. Sriram S, Bhattacharyya SS (2009) Embedded multiprocessors: scheduling and synchronization, 2nd edn. Marcel Dekker, Inc, New York
240. Standard for information technology—portable operating system interface (POSIX) Shell and utilities IEEE Std 10031–2004, The Open Group Base Specifications Issue 6, section 29 (2004) IEEE and The Open Group
241. Stefanov T (2004) Converting weakly dynamic programs to equivalent process network specifications. PhD thesis, Leiden University, The Netherlands
242. STMicroelectronics and CEA (2010) Platform 2012: a many-core programmable accelerator for ultra-efficient embedded computing in nanometer technology. White Pap. http://www.2parma.eu/documents/publications.html
243. Straumann T (2001) Open source real time operating systems overview. CoRR cs.OS/0111035
244. Stuijk S, Geilen M, Basten T (2006) SDF^3: SDF for free. In: Proceedings of the 6th international conference on application of concurrency to system design (ACSD 2006). IEEE Computer Society Press, Los Alamitos, pp 276–278, doi:10.1109/ACSD.2006.23. http://www.es.ele.tue.nl/sdf3
245. Stuijk S, Basten T, Geilen MCW, Corporaal H (2007) Multiprocessor resource allocation for throughput-constrained synchronous dataflow graphs. In: Proceedings of the 44th annual design automation conference (DAC '07). ACM, New York, pp 777–782. http://doi.acm.org/10.1145/1278480.1278674
246. Stulova A, Leupers R, Ascheid G (2012) Throughput driven transformations of synchronous data flows for mapping to heterogeneous MPSoCs. In: International conference on embedded computer systems: architectures, modeling and simulation (SAMOS XII)
247. Sun Y, Amiri K, Brogioli M, Cavallaro JR (2012) Application-specific accelerators for communications. In: Bhattacharyya SS, Deprettere EF, Leupers R, Takala J (eds) Handbook of signal processing systems, 2nd edn. Springer, Berlin
248. Suzanne Deffree (2009) ASIC design starts to drop 22reports http://www.edn.com/article/459021-ASIC_design_starts_to_drop_22_in_2009_Gartner_reports.php
249. Synopsys (2012) Design compiler graphical. http://www.synopsys.com/Tools/Implementation/RTLSynthesis/DCGraphical
250. Synopsys (2012) Platform architect. http://www.synopsys.com/Tools/SLD/VirtualPrototyping/Pages/PlatformArchitect.aspx
251. Synopsys (2012) Prime time. http://www.synopsys.com/tools/implementation/signoff/pages/primetime.aspx
252. Synopsys (2012) Processor designer. http://www.synopsys.com/Tools/SLD/ProcessorDev/Pages/default.aspx

253. Synopsys (2012) Signal processing worksystem (SPW). http://www.synopsys.com/systems/blockdesign/digitalsignalprocessing/pages/signa l-processing.aspx
254. Synopsys (2012) System studio. http://www.synopsys.com/Systems/BlockDesign/DigitalSignalProcessing/Pages/Syste mStudio.aspx
255. Target (2013) IP designer. http://www.retarget.com/index.php
256. Tensilica (2013) Xtensa customizable processors. http://www.tensilica.com/products/xtensa-customizable
257. Texas Instruments (2013) Keystone device architecture. http://processors.wiki.ti.com/index.php/Keystone
258. Texas Instruments (2013) OMAP mobile processors. http://www.ti.com/lsds/ti/omap-applications-processors/features.page
259. The Object Management Group (OMG) (2013) CORBA 3.1. http://www.omg.org/spec/CORBA/3.1/
260. The OpenMP Architecture Review Board (2009) The openMP specification for parallel programming. http://www.openmp.org
261. Thiele L (2013) Distributed Application Layer (DAL). http://www.tik.ee.ethz.ch/euretile/dal.php
262. Thiele L, Wilhelm R (2004) Design for timing predictability. Real Time Syst 28(2–3):157–177, doi:10.1023/B:TIME.0000045316.66276.6e. http://dx.doi.org/10.1023/B:TIME.0000045316.66276.6e
263. Thiele L, Bacivarov I, Haid W, Huang K (2007) Mapping applications to tiled multiprocessor embedded systems. In: Proceedings of the ACSD '07, IEEE Computer Society, New Yor. http://dx.doi.org/10.1109/ACSD.2007.53
264. Thies W, Chandrasekhar V, Amarasinghe S (2007) A practical approach to exploiting coarse-grained pipeline parallelism in C programs. In: Proceedings of the 40th annual IEEE/ACM international symposium on microarchitecture (MICRO 40). IEEE Computer Society, New York, pp 356–369, doi:10.1109/MICRO.2007.7. http://dx.doi.org/10.1109/MICRO.2007.7
265. Tobias Grosser (2012) Polly: polyhedral optimizations for LLVM. http://www.polly.llvm.org/
266. Tournavitis G, Wang Z, Franke B, O'Boyle M (2009) Towards a holistic approach to auto-parallelization—integrating profile-driven parallelism detection and machine-learning based mapping. In: Proceedings of the programming language design and implementation conference (PLDI 09), Dublin, 15–20 June
267. Tsang E (1993) Foundations of constraint satisfaction. Academic press limited, department of computer science. University of Essex Colchester, Essex
268. Tudor D, Macariu G, Jebelean C, Cretu V (2009) Towards a load balancer architecture for multi-core mobile communication systems. In: 5th International symposium on applied computational intelligence and informatics (SACI '09), pp 391–396. doi:10.1109/SACI.2009.5136280
269. UBM Electronics (2012) 2012 Embedded market survey. http://www.embedded.com/electronics-blogs/embedded-market-surveys/4405646/2012- Embedded-Market-Survey
270. Ullman JD (1975) NP-complete scheduling problems. J Comput Syst Sci 10(3):384–393, doi:10.1016/S0022-0000(75),80008--0. http://dx.doi.org/10.1016/S0022-0000(75)80008--0
271. University RA (2012) Ultra-high speed mobile information and communication (UMIC). http://www.umic.rwth-aachen.de
272. Urfianto MZ, Isshiki T, Khan AU, Li D, Kunieda H (2006) A multiprocessor system-on-chip architecture with enhanced compiler support and efficient interconnect. In: IP-SOC 2006, Design and reuse
273. Vasudevan N, Edwards SA (2009) Celling SHIM: compiling deterministic concurrency to a heterogeneous multicore. In: Proceedings of the ACM symposium on applied computing (SAC), Honolulu
274. VectorFabrics (2012) vfEmbedded and vfThreaded-x86. http://www.vectorfabrics.com/
275. Verdoolaege S, Nikolov H, Stefanov T (2007) PN: a tool for improved derivation of process networks. EURASIP J Embed Syst 2007(1):13. http://www.dx.doi.org/10.1155/2007/75947

276. Voronenko Y, Arbatov V, Berger CR, Peng R, Püschel M, Franchetti F, (2010) Computer generation of platform-adapted physical layer software. In: Wireless innovation conference and product exposition (SDR'10). Wireless Innovation Forum, Washington, DC, USA
277. Wallace G (1992) The JPEG still picture compression standard. Consumer electronics, IEEE Trans 38(1):18–34. doi:10.1109/30.125072
278. Wei YC, Cheng CK (1991) Ratio cut partitioning for hierarchical design. IEEE Trans Comput Aided Des 10(7):911–921
279. Weng RN, Wolf T (2009) Analytic modeling of network processors for parallel workload mapping. ACM Trans Embed Comput Syst 8:18:1–18:29
280. Whittier R (1996) Push/Pull: PC technology/End user demand. In: Symposium on VLSI technology. Digest of technical papers, pp 2–5. doi:10.1109/VLSIT.1996.507773
281. Wiggers MH (2009) Aperiodic multiprocessor scheduling for real-time stream processing applications. PhD thesis, University of Twente, 978–90-365-2850-4
282. Wiggers MH, Bekooij MJG, Smit GJM (2008) Buffer capacity computation for throughput constrained streaming applications with data-dependent inter-task communication. In: Proceedings of the 2008 IEEE real-time and embedded technology and applications symposium (RTAS '08). IEEE Computer Society, Washington, DC, USA, pp 183–194. http://dx.doi.org/10.1109/RTAS.2008.10
283. Wiggers MH, Bekooij MJG, Smit GJM (2011) Buffer capacity computation for throughput-constrained modal task graphs. ACM Trans Embed Comput Syst 10(2):17:1–17:59, doi:10.1145/1880050.1880053. http://doi.acm.org/10.1145/1880050.1880053
284. van der Wijngaart RF, Mattson TG, Haas W (2011) Light-weight communications on intel's single-chip cloud computer processor. SIGOPS Oper Syst Rev 45:73–83, doi:10.1145/1945023.1945033. http://doi.acm.org/10.1145/1945023.1945033
285. Wikipedia (2013) Embedded system. http://en.wikipedia.org/wiki/Embedded_system
286. Wikipedia (2013) List of concurrent and parallel programming languages. http://en.wikipedia.org/wiki/List_of_concurrent_and_parallel_programming_langua ges
287. Wikipedia (2013) OMAP. http://en.wikipedia.org/wiki/OMAP
288. Wikipedia (2013) Snapdragon (system on chip). http://en.wikipedia.org/wiki/Snapdragon_(System_on_Chip)
289. Wilhelm R, Engblom J, Ermedahl A, Holsti N, Thesing S, Whalley D, Bernat G, Ferdinand C, Heckmann R, Mitra T, Mueller F, Puaut I, Puschner P, Staschulat J, Stenström P (2008) The worst-case execution-time problem—overview of methods and survey of tools. ACM Trans Embed Comput Syst 7(3):1–53. http://doi.acm.org/10.1145/1347375.1347389
290. Wilson RP, French RS, Wilson CS, Amarasinghe SP, Anderson JM, Tjiang SWK, Liao SW, Tseng CW, Hall MW, Lam MS, Hennessy JL (1994) SUIF: an infrastructure for research on parallelizing and optimizing compilers. SIGPLAN Not 29:31–37, doi:10.1145/193209.193217. http://doi.acm.org/10.1145/193209.193217
291. Witte EM (2012) Efficient and flexibility tradeoff of soft-input soft-output sphere decoding architecture. PhD thesis, RWTH Aachen Univeristy. Institute for Communication Technologies and Embedded Systems (ICE), Aachen
292. Witte EM, Borlenghi F, Ascheid G, Leupers R, Meyr H (2010) A Scalable VLSI-architecture for soft-input soft-output single tree-search sphere decoding. IEEE Trans Circuits Syst Part II: Express Briefs (57):706–710
293. Wolf W (2004) The future of multiprocessor systems-on-chips. In: Proceedings of the 41st annual conference on design automation (DAC '04). ACM Press, New York, pp 681–685. http://doi.acm.org/10.1145/996566.996753
294. Xilinx (2011) LogiCORE IP fast fourier transform v7.1. Product specification. Xilinx Inc, San Jose (DS260)
295. Xilinx (2012) IP documentation. http://www.xilinx.com/support/documentation/
296. Yehia S, Girbal S, Berry H, Temam O (2009) Reconciling specialization and flexibility through compound circuits. In: IEEE 15th International symposium on high performance computer architecture (HPCA 2009), pp 277–288. doi:10.1109/HPCA.2009.4798263

297. Yoon JS, Kim JH, Kim HE, Lee WY, Kim SH, Chung K, Park JS, Kim LS (2010) A graphics and vision unified processor with 0.89 uW/fps pose estimation engine for augmented reality. In: IEEE international solid-state circuits conference digest of technical papers (ISSCC), pp 336–337. doi:10.1109/ISSCC.2010.5433907
298. Zeligsoft (2012) Zeligsoft CE 2.4. http://www.zeligsoft.com/tools/zeligsoft-ce
299. Zhang D, Nikitopoulos K, Lai IW, Ascheid G, Meyr H (2010) Iterative channel estimation control for MIMO-OFDM systems. In: 44th Annual conference on information sciences and systems (CISS), pp 1–6. doi:10.1109/CISS.2010.5464953
300. Zhang D, Zhang H, Castrillon J, Kempf T, Vanthournout B, Ascheid G, Leupers R (2011) Optimized communication architecture of MPSoCs with a hardware scheduler: a system-level analysis. Int J Embed Real Time Commun Syst 2(3):1–20. doi:10.4018/jertcs.2011070101
301. Zhang D, Lu L, Castrillon J, Kempf T, Ascheid G, Leupers R, Vanthournout B (2013) Application-aware spinlock control using a hardware scheduler in MPSoC platforms. Inte J Embed Real Time Commun Syst (to appear)

Glossary

Notation (General, Multi-application Flow)

Of	big O notation for asymptotic runtime complexity order
$\wp(S)$	power set of set S
$\mathscr{A}$	Set of all applications, $\mathscr{A} = \mathscr{A}^{\text{seq}} \sqcup \mathscr{A}^{\text{kpn}} \sqcup \mathscr{A}^{\text{sdr}}$
ACG	application concurrency graph
ACS	application concurrency set
$\mathscr{A}^{\text{hrt}}$	set of applications with hard real-time constraints
$\mathscr{A}^{\text{seq}}$	set of sequential applications
$\mathscr{A}^{\text{kpn}}$	set of parallel KPN applications
$\mathscr{A}^{\text{srt}}$	set applications with soft real time constraints
$\mathscr{A}^{\text{nrt}}$	set of applications with no real time constraints
$\mathscr{A}^{\text{sdr}}$	set of applications for SDR
A	generic application, $A \in \mathscr{A}$
$\mathscr{CM}^{CP}$	cost model of a communication primitive
ζ^{CP}	cost function associated with communication primitive CP
$\mathscr{CP}$	set of all communication channels in the target MPSoC
$\mathscr{CM}^{PT}$	cost model of a processor type
ζ^{PT}	cost function associated with processor type PT
$\zeta^{PT,\text{dy}}$	dynamic cost function associated with processor type PT
$\zeta^{PT,\text{st}}$	static cost function associated with processor type PT
CP	generic communication channel, $CP \in \mathscr{CP}$
$\mathscr{CR}$	set of all communication resources in the target MPSoC
CR	generic communication resource, $CR \in \mathscr{CR}$
D_v^A	domain of variable v of application A
D_v^{PT}	domain of variable v of processor type PT
dst	generic function that returns the target node of an edge within a graph
$\mathscr{K}^A$	application constraints
K_i^A	i-th constraint of application A
$\mathscr{M}^A$	application model
μ_a	assignment of platform and application variables
μ_c	mapping of communication to communication primitives
μ_p	mapping of processes to processing elements
$\mathscr{PE}$	set of all processing elements in the target MPSoC
PE	generic processing element, $PE \in \mathscr{PE}$

J. Castrillón Mazo and R. Leupers, *Programming Heterogeneous MPSoCs*, DOI: 10.1007/978-3-319-00675-8,

$\mathscr{PE}^v$	shorthand notation for the set of all processing elements of type v
PE_i^v	shorthand notation for the i-th processing element of processor type v, $PE_i^v \in PE$, $v \in \mathscr{PT}$
$\mathscr{PT}$	set of all processor type in the target MPSoC
PT	generic processor type, $PT \in \mathscr{PT}$
RC^A	runtime configuration for application A
$\mathscr{RC}^A$	set of runtime configurations for application A
$\mathscr{RC}^{UC}$	set of runtime configurations for each application in use case UC
res	function that maps a communication primitive with its resources, res : $\mathscr{CP} \rightarrow \wp(\mathscr{CR})$
SOC	graph model of a given MPSoC, $SOC = (\mathscr{PE}, \mathscr{CP})$
src	generic function that returns the source node of an edge within a graph
UC	use case, subset of applications that may run concurrently and associated weight
$\vartheta_{PE}^{RC^A}$	utilization function of an application A on a processor PE according to a runtime configuration RC^A
$\mathscr{U}_{SOC}^{RC^A}$	set of all utilization functions of application A on platform SOC according to a runtime configuration RC^A
var_{size}	function that returns the size of a variable associated with a data flow edge
V^A	application variables
V^{PT}	set of variables of a processor type
v_{SP}^{PT}	variable of a processor type that represents the scheduling policy
$\omega_{\text{dc}}^{\mathscr{RC}^{UC}}$	multi-application score for use case UC with use case runtime configuration $\mathscr{RC}^{UC}$ according to the displacement-criterion
$\omega_{\text{mc}}^{\mathscr{RC}^{UC}}$	multi-application score for use case UC with use case runtime configuration $\mathscr{RC}^{UC}$ according to the mean-criterion
x_{CH}^{CR}	amount of channels that can be implemented over a communication resource CR
x_{MEM}^{CR}	memory size of communication resource CR
X^{PT}	attribute set of a processor type
x_{cs}^{PT}	attribute of a processor type that models the time spent in a context switch
x_{tasks}^{PT}	attribute of a processor type that defines the maximum amount of tasks that can executed
π^{P^A}	sequential profile of a process P^A
BB^A	generic basic block of application A
$\mathscr{BB}^{f^A}$	set of basic blocks of function f^A in application A
$\mathscr{BB}^A$	set of basic blocks of application A
CB^f	coupled block in a function f
$CDFG^{f^A}$	control-data flow graph of function f^A
CFG^{f^A}	control flow graph of function f^A
CG^A	call graph of application A
$\mathscr{C}^G$	clustering of a graph G
$CDFG_{\text{par}}^{f^A}$	parallel annotated graph for function f^A

Notation (Sequential Flow)

CG_{par}^A	call graph of application A with function graphs that are parallel-annotated
CG_{pi}^A	call graph of application A with function graphs for which a parallel implementation has been determined
δ^c	control dependence relation
δ^f	forward data dependence relation
δ^a	anti-data dependence relation

δ^o	output data dependence relation
DFG^{f^A}	data flow graph of function f^A
dom	dominance relationship of nodes in a control flow graph
$D_v^{PA^n}$	domain of variable v of a parallel annotation PA^n
$E_c^{f^A}$	set of control flow edges of function f^A
E_c^A	set of control flow edges of all functions in application$A$
E_{cg}^A	set of call graph edges of application A
$E_d^{f^A}$	set of data flow edges of function f^A
$E_d^{f^A*}$	set of data flow edges of function f^A defined over basic blocks
E_d^A	set of data flow edges of all functions in application A
f^A	generic function of application A
IR^A	intermediate representation of application A
$\mathscr{M}_{\text{par}}^A$	parallel-annotated application model for application A
PA^n	parallel annotation for node n of a given graph
$\mathscr{PI}^A$	parallel implementation option for a sequential application A
$\mathscr{PA}^V$	set of parallel annotations for a set of nodes V of a given graph
pdom	post-dominance relationship of nodes in a control flow graph
pred	function that returns the control flow predecessors of a node
π^A	sequential profile of application A
s^A	generic IR-statement of application A
σ^A	function that provides information about function call sites of application A
$\mathscr{SE}^A$	set of all elements in the model of a sequential application A
S_{f}^A	set of all functions of application A
$S_{\text{stmt}}^{f^A}$	set of all statements in function f^A
S_{stmt}^A	set of all statements of application A
succ	function that returns the control flow successors of a node
V^{PA^n}	set of variables of a parallel annotation for node n
X^{PA^n}	attribute set of a parallel annotation for node n

Notation (Parallel Flow)

$b_{C^A}^A$	variable that represents the size of channel C^A of application A
β	assignment of buffer size variables
C^A	fifo channel of a KPN application A
CG^{P^A}	call graph of process P^A
$\mathscr{C}^A$	set of all fifo channel of KPN application A
$I_i^{P^A}$	i-th process iteration of process P^A
$\mathscr{I}^{P^A}$	sequence of process iterations of process P^A
KPN^A	KPN graph model of an application
$N_{\text{it}}^{P^A}$	number of iterations of process P^A
P^A	process of application A
$\mathscr{PAE}^A$	set of all elements in the model of a parallel application A
$\mathscr{P}^A$	set of processes of application A
RE^A	set with all read events of a KPN application A
$S_i^{P^A}$	i-th segment of process P^A
$\mathscr{S}^A$	set of all segments in application A
$\mathscr{SE}^{P^A}$	set of all elements in the sequential model of a process P^A
$\mathscr{TG}^A$	trace graph of application A
T^{P^A}	trace of process P^A

$\mathscr{T}^A$	set of all process traces of application A
V^A_{size}	set of channel size variables of application A
WE^A	set with all write events of a KPN application A

Notation (SDR Flow)

$\mathscr{CM}^{F^{SOC}}$	cost model of flavor F^{SOC}
$\zeta^{F^{SOC}}$	cost function associated with a flavor F^{SOC}
$D_v^{F^{SOC}}$	domain of variable v of a flavor F^{SOC}
$D_v^{N^A}$	domain of variable v of a nucleus N^A
F^{SOC}	a flavor in platform SOC
$\mathscr{F}^{SOC}$	set of all flavors in platform SOC
$IN^{F^{SOC}}$	set of of input ports of a flavor F^{SOC}
IN^{N^A}	set of of input ports of a nucleus N^A
$K_i^{F^{SOC}}$	i-th constraint of flavor F^{SOC}
$\mathscr{K}^{F^{SOC}}$	set of constraints of a flavor F^{SOC}
$K_i^{N^A}$	i-th constraint of nucleus N^A
$\mathscr{K}^{N^A}$	set of constraints of a nucleus N^A
μ_f	assignment of flavor variables
μ_n	mapping of nucleus to flavors
$\equiv_{\text{IF}}$	relation that describes that two interfaces match
N^A	a nucleus of application A
$\mathscr{N}^A$	set of all nuclei in application A
NC^A	nucleus configuration for an application A, mapping of nucleus to flavors and matching flavor configuration
$\mathscr{N}$	set of all nuclei (for example defined in a library)
$OUT^{F^{SOC}}$	set of output ports of a flavor F^{SOC}
OUT^{N^A}	set of output ports of a nucleus N^A
$\mathscr{PE}^{F^{SOC}}$	set of PEs that contain the flavor F^{SOC}
P^{N^A}	process description associated with a nucleus N^A
$\mathscr{SI}^A$	SDR implementation of application A
$V^F_{C^A}$	set of variables of a flavor F that describe the interface of the port connected over a channel C^A
$V^{F^{SOC}}$	set of variables of a flavor F^{SOC}
V^{N^A}	set of variables of a nucleus N^A

Index

J. Castrillón Mazo and R. Leupers, *Programming Heterogeneous MPSoCs*, DOI: 10.1007/978-3-319-00675-8,