1. Record Nr. UNINA9910784235303321

Autore	Alexander Perry
Titolo	System-level design with Rosetta [[electronic resource] /] / Perry Alexander
Pubbl/distr/stampa	San Francisco, : Elsevier/Morgan Kaufman, c2007
ISBN	1-281-01500-8 9786611015008 0-08-049837-X
Descrizione fisica	1 online resource (375 p.)
Collana	Morgan Kaufmann series in systems on silicon
Disciplina	621.3815
Soggetti	Integrated circuits - Computer-aided design Microelectronics
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Description based upon print version of record.
Nota di bibliografia	Includes bibliographical references (p. 339-341) and index.
Nota di contenuto	Front Cover; System-Level Design with Rosetta; Copyright Page; Contents; Acknowledgments; Foreword; Preface; Part I: Introduction; Chapter 1. Introduction; 1.1 What is System-Level Specification?; 1.2 Rosetta's Design Goals; 1.3 Anatomy of a Specification; 1.4 Learning Rosetta; Part II: The Expression Language; Chapter 2. Items, Values, Types, and Declarations; 2.1 Labels, Values, and Types; 2.2 Item Declarations and Type Assertions; 2.3 Universal Operations; Chapter 3. Expressions; 3.1 Atomic Expressions; 3.2 Function Application; 3.3 Operator Application; 3.4 If Expressions 3.5 Case Expressions3.6 Let Expressions; 3.7 Compound Expressions; Chapter 4. Elemental Types; 4.1 The Boolean Type; 4.2 The Number Types; 4.3 The Character Type; 4.4 The Element Type; 4.5 The Top and Bottom Types; 4.6 Element Literals; 4.7 Operator Result Types; Chapter 5. Composite Types; 5.1 Type Formers; 5.2 Set Types; 5.3 Multiset Types; 5.4 Sequence Types; Chapter 6. Functions; 6.1 Direct Function Definition; 6.2 Function Values and Function Types; 6.3 Evaluating Functions; 6.4 Universally Quantified Parameters; Chapter 7. Higher-Order Functions; 7.1 Domain, Range, and Return Functions 7.2 Alternate Higher-Order Function Notation7.3 Minimum and Maximum; 7.4 Quantifiers and Comprehension; 7.5 Sequences and Higher-Order Functions; 7.6 Function Inclusion and Composition;

Chapter 8. User-Defined Types; 8.1 Defining New Types; 8.2 Defining Types By Extension; 8.3 Defining Types By Comprehension; 8.4 Defining Constructed Types; 8.5 Functions as Type Definition Tools; Part III: The Facet Language; Chapter 9. Facet Basics; 9.1 A First Model - An AM Modulator; 9.2 Composing Models - Adding Constraints; 9.3 Combinational Circuits - A Simple Adder; 9.4 Defining State - A 2-bit Counter
9.5 Defining Structure - A 2-bit Adder9.6 Specification Reuse - Using Packages; 9.7 Abstract Specification - Architecture Definition; Chapter 10. Defining Facets; 10.1 Direct Facet Definition; 10.2 Separable Definitions; 10.3 Facets and Hardware Description Languages; 10.4 Facet Styles; 10.5 Scoping Rules; 10.6 Basics of Facet Semantics; Chapter 11. Packages, Libraries, and Components; 11.1 Packages; 11.2 Libraries; 11.3 Components; Part IV: Domains and Interactions; Chapter 12. Domains; 12.1 Elements of a Domain; 12.2 The Standard Domains; 12.3 Domains and Facet Types; Chapter 13. Reflection
13.1 Template Expressions and AST Structures13.2 Interpreting AST Structures; 13.3 Domain Declarations; 13.4 Defining Engineering Domains; 13.5 Defining New Model-of-Computation Domains; 13.6 Defining New Unit-of-Semantics Domains; 13.7 Defining Ticked and Dereferencing Expressions; 13.8 Consistent Domain Extension; Chapter 14. The Facet Algebra; 14.1 Facet Products and Sums; 14.2 Facet Homomorphism and Isomorphism; 14.3 Conditional Expressions; 14.4 Let Expressions; 14.5 Higher-Order Facets; Chapter 15. Domain Interactions; 15.1 Projection Functions, Functors, and Combinators
15.2 Defining Interactions

Sommario/riassunto

The steady and unabated increase in the capacity of silicon has brought the semiconductor industry to a watershed challenge. Now a single chip can integrate a radio transceiver, a network interface, multimedia functions, all the ""glue"" needed to hold it together as well as a design that allows the hardware and software to be reconfigured for future applications. Such complex heterogeneous systems demand a different design methodology. A consortium of industrial and government labs have created a new language and a new design methodology to support this effort. Rosetta permits designers t

2. Record Nr. UNISA996691662603316

Autore	Li Chao
Titolo	Advanced Parallel Processing Technologies : 16th International Symposium, APPT 2025, Athens, Greece, July 13-16, 2025, Proceedings / / edited by Chao Li, Xuehai Qian, Dimitris Gizopoulos, Boris Grot
Pubbl/distr/stampa	Singapore : , : Springer Nature Singapore : , : Imprint : Springer, , 2026
ISBN	9789819510214 9789819510207
Edizione	[1st ed. 2026.]
Descrizione fisica	1 online resource (598 pages)
Collana	Lecture Notes in Computer Science, , 1611-3349 ; ; 16062
Disciplina	004.35
Soggetti	Software engineering Operating systems (Computers) Computer systems Computers, Special purpose Artificial intelligence Software Engineering Operating Systems Computer System Implementation Special Purpose and Application-Based Systems Artificial Intelligence
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Nota di contenuto	-- Best Paper Candidates -- DACO: Unlocking Latent Dataflow Opportunities in Edge-side SIMT Accelerators -- ATLAS: Efficient Dynamic GNN System through Abstraction-Driven Incremental Execution -- Segmentation-Aware Optimization of Collective for Waferscale Chips -- Area-Efficient Automated Logic Design with Monte-Carlo Tree Search -- Chip and Accelerators -- NFMap: Node Fusion Optimization for Efficient CGRA Mapping with Reinforcement Learning -- A Unified Synthesis Framework for Dataflow Accelerators through Multi-Level Software and Hardware Intermediate Representations -- Defect-aware Task Scheduling and Mapping for Redundancy-Enhanced Spatial Accelerators -- Irregular Sparsity-

Enabled Search-In-Memory Engine for Accelerating Spiking Neural Networks -- Memory and Storage -- QRAMsim: Efficiently Simulating, Analyzing, and Optimizing Large-scale Quantum Random Access Memory -- CeDMA: Enhancing Memory Efficiency of Heterogeneous Accelerator Systems Through Central DMA Controlling -- PAMM: Adaptive Memory Management for CXL-/UB-Based Heterogeneous Memory Pooling Systems -- STAMP: Accelerating Second-order DNN Training Via ReRAM-based Processing-in-Memory Architecture -- Cloud and Networking -- Cochain: Architectural Support Mechanism for Blockchain-based Task Scheduling -- DyQNet: Optimizing Dynamic Entanglement Routing with Online Request in Quantum Network -- Veyth: Adaptive Container Placement for Optimizing Cross-Server Network Traffic of Microservice Applications -- Design for LLM and ML/AI -- Unifying Two Operators with One PIM: Leveraging Hybrid Bonding for Efficient LLM Inference -- AsymServe: Demystifying and Optimizing LLM Serving Efficiency on CPU Acceleration Units -- SparseTem: Boosting the Efficiency of CNN-Based Video Encoders by Exploiting Temporal Continuity -- TokenSim: Enabling Hardware and Software Exploration for Large Language Model Inference Systems -- Big Data and Graph Processing -- Achieving Efficient Temporal Graph Transformation on the GPU -- GASgraph: A GPU-accelerated Streaming Graph Processing System based on SubHPMAs -- Accelerating Large-Scale Out-of-GPU-Core GNN Training with Two-Level Historical Caching -- Understand Data Preprocessing for Effective End-to-End Training of DNN -- Secure and Dependable System -- TwinStore: Secure Key-Value Stores Made Faster with Hybrid Trusted/Untrusted Storage -- The Future of Fully Homomorphic Encryption: from a Storage I/O Perspective -- LASM: A Lightweight and General TEE Secure Monitor Framework -- Identifying Potential Anomalous Operations in Graph Neural Network Training -- APPT Posters -- DraEC: A Decentralized Routing Algorithm in Erasure-Coded Deduplication System -- Spatial-Aware Orchestration of LLM Attention on Waferscale Chips -- ACLP: Towards More Accurate Loop Prediction for High-Performance Processors -- DSL-SGD: Distributed Local Stochastic Gradient Descent with Delayed Synchronization -- Exploiting Large Language Models for Software-Defined Solid-State Drives Design -- Comber: QoS-aware and Efficient Deployment for Co-located Microservices and Best-Effort Tasks in Disaggregated Datacenters -- NISA-DV: Verification Framework for Neuromorphic Processors with Customized ISA -- Lembda: Optimizing LLM Inference on Embedded Platforms via CPU/FPGA Co-Processing -- QDLoRA: Enhanced LoRA Fine-Tuning on Quantized LLMs via Integrated Low-Rank Decomposition.

Sommario/riassunto

This book constitutes the refereed proceedings of the 16th International Symposium on Advanced Parallel Processing Technologies, APPT 2025, held in Athens, Greece, during July 13–16, 2025. The 17 full papers and 10 short papers included in this book were carefully reviewed and selected from 74 submissions. They were organized in topical sections as follows: Chip and Accelerators, Memory and Storage, Cloud and Networking, Design for LLM and ML/AI, Big Data and Graph Processing, and Secure and Dependable System.