1. Record Nr.	UNINA9910483387903321
Autore	Kovalev Mikhail
Titolo	A Pipelined Multi-core MIPS Machine : Hardware Implementation and Correctness Proof / / by Mikhail Kovalev, Silvia M. Müller, Wolfgang J. Paul
Pubbl/distr/stampa	Cham : , : Springer International Publishing : , : Imprint : Springer, , 2014
ISBN	3-319-13906-1
Edizione	[1st ed. 2014.]
Descrizione fisica	1 online resource (XII, 352 p. 147 illus.)
Collana	Theoretical Computer Science and General Issues, , 2512-2029 ; ; 9000
Disciplina	004.165
Soggetti	Algorithms Software engineering Computers Microprocessors Computer architecture Computer networks Compilers (Computer programs) Software Engineering Computer Hardware Processor Architectures Computer Communication Networks Compilers and Interpreters
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Bibliographic Level Mode of Issuance: Monograph
Nota di contenuto	Introduction -- Motivation -- Overview -- Number Formats and Boolean Algebra -- Basics -- Numbers, Sets and Logical Connectives -- Sequences and Bit-Strings -- Modulo Computation -- Geometric Sums -- Binary Numbers -- Two’s Complement Numbers -- Boolean Algebra -- Identities -- Solving Equations -- Disjunctive Normal Form -- Hardware -- Digital Gates and Circuits -- Some Basic Circuits -- Clocked Circuits -- Digital Clocked Circuits -- The Detailed Hardware Model -- Timing Analysis -- Registers -- Drivers and Main Memory -- Open Collector Drivers and Active Low Signal -- Tristate Drivers and

Bus Contention -- The Incomplete Digital Model for Drivers -- Self Destructing Hardware -- Clean Operation of Tristate Buses -- Specification of Main Memory -- Operation of Main Memory via a Tristate Bus -- Finite State Transducers -- Realization of Moore Automata -- Precomputing Outputs of Moore Automata -- Realization of Mealy Automata -- Precomputing Outputs of Mealy Automata -- Nine Shades of RAM -- Basic Random Access Memory -- Single-Port RAM Designs -- Read Only Memory (ROM) -- Multi-bank RAM -- Cache State RAM -- SPR RAM -- Multi-port RAM Designs -- 3-port RAM for General Purpose Registers -- General 2-port RAM -- 2-port Multi-bank RAM-ROM -- 2-port Cache State RAM -- Arithmetic Circuits -- Adder and Incrementer -- Arithmetic Unit -- Arithmetic Logic Unit (ALU) -- Shift Unit -- Branch Condition Evaluation Unit -- A Basic Sequential MIPS Machine -- Tables -- I-type -- R-type -- J-type -- MIPS ISA -- Configuration and Instruction Fields -- Instruction Decoding -- ALU-Operations -- Shift Unit Operations -- Branch and Jump -- Sequences of Consecutive Memory Bytes -- Loads and Stores -- ISA Summary -- A Sequential Processor Design -- Software Conditions -- Hardware Configurations and Computations -- Memory Embedding -- Defining Correctness for the Processor Design -- Stages of Instruction Execution -- Initialization -- Instruction Fetch -- Instruction Decoder -- Reading from General Purpose Registers -- Next PC Environment -- ALU Environment -- Shift Unit Environment -- Jump and Link -- Collecting Results -- Effective Address -- Shift for Store Environment -- Memory Stage -- Shifter for Load -- Writing to the General Purpose Register File -- Pipelining.-MIPS ISA and Basic Implementation Revisited -- Delayed PC -- Implementing the Delayed PC -- Pipeline Stages and Visible Registers -- Basic Pipelined Processor Design -- Transforming the Sequential Design -- Scheduling Functions -- Use of Invisible Registers -- Software Condition SC-1 -- Correctness Statement -- Correctness Proof of the Basic Pipelined Design -- Forwarding -- Hits -- Forwarding Circuits -- Software Condition SC-2 -- Scheduling Functions Revisited -- Correctness Proof -- Stalling Stall Engine -- Hazard Signals -- Correctness Statement -- Scheduling Functions -- Correctness Proof -- Liveness -- Caches and Shared Memory -- Concrete and Abstract Caches -- Abstract Caches and Cache Coherence -- Direct Mapped Caches -- k-way Associative Caches -- Fully Associative Caches -- Notation -- Parameters -- Memory and Memory Systems -- Accesses and Access Sequences -- Sequential Memory Semantics -- Sequentially Consistent Memory Systems -- Memory System Hardware Configurations -- Atomic MOESI Protocol -- Invariants -- Defining the Protocol by Tables -- Translating the Tables into Switching Functions -- Algebraic Specification -- Properties of the Atomic Protocol -- Gate Level Design of a Shared Memory System -- Specification of Interfaces -- Data Paths of Caches -- Cache Protocol Automata -- Automata Transitions and Control Signals -- Bus Arbiter -- Initialization -- Correctness Proof -- Arbitration -- Silent Slaves and Silent Masters -- Automata Synchronization -- Control of Tristate Drivers -- Protocol Data Transmission -- Data Transmission -- Accesses of the Hardware Computation -- Relation with the Atomic Protocol -- Ordering Hardware Accesses Sequentially -- Sequential Consistency -- Liveness -- A Multi-core Processor -- Compare-and-Swap Instruction -- Introducing CAS to the ISA -- Introducing CAS to the Sequential Processor -- Multi-core ISA and Reference Implementation -- Multi-core ISA Specification -- Sequential Reference Implementation -- Simulation Relation -- Local Configurations and Computations -- Accesses of the Reference Computation -- Shared Memory in the Multi-core System -- Notation -- Invisible Registers and

Sommario/riassunto

This monograph is based on the third author's lectures on computer architecture, given in the summer semester 2013 at Saarland University, Germany. It contains a gate level construction of a multi-core machine with pipelined MIPS processor cores and a sequentially consistent shared memory. The book contains the first correctness proofs for both the gate level implementation of a multi-core processor and also of a cache based sequentially consistent shared memory. This opens the way to the formal verification of synthesizable hardware for multi-core processors in the future. Constructions are in a gate level hardware model and thus deterministic. In contrast the reference models against which correctness is shown are nondeterministic. The development of the additional machinery for these proofs and the correctness proof of the shared memory at the gate level are the main technical contributions of this work.