1.

Record Nr.	UNINA9910457323303321
Autore	Katz David J
Titolo	Embedded media processing [[electronic resource] /] / by David Katz and Rick Gentile
Pubbl/distr/stampa	Boston, : Elsevier/Newnes, c2005
ISBN	1-280-64264-5 9786610642649 0-08-045888-2
Descrizione fisica	1 online resource (425 p.)
Collana	Embedded Technology
Altri autori (Persone)	GentileRick
Disciplina	004.16
Soggetti	Signal processing - Digital techniques Embedded computer systems Electronic books.
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Includes index.
Nota di contenuto	Front cover; Embedded Media Processing; Copyright page; Table of Contents; Preface; History of This Book; Chapter Overviews; Chapter 1: Embedded Media Processing; Chapter 2: Memory Systems; Chapter 3: Direct Memory Access (DMA); Chapter 4: System Resource Partitioning and Code Optimization; Chapter 5: Basics of Embedded Audio Processing; Chapter 6: Basics of Embedded Video and Image Processing; Chapter 7: Media Processing Frameworks; Chapter 8: Power Management for Embedded Systems; Chapter 9: Application Examples; Acknowledgments; Acronyms; About the Authors What's on the (Companion website)?Chapter 1: Embedded Media Processing; Why Are You Reading This Book?; So What's All the Excitement About Embedded Multimedia Systems?; A Simplified Look at a Media Processing System; Core Processing; Input/Output Subsystems-Peripheral Interfaces; Subsystem Control; Storage; Connectivity; Data Movement; Memory Subsystem; Laying the Groundwork for an EMP Application; What Kind(s) of Media Am I Dealing With?; What Do I Need to Do With the Data?; Are My System Needs Likely to Change Over Time, or Will This Be a Static System?; Is This a Portable Application?

Does my Application Require a Fixed-Point or Floating-Point Device? How Does the Data Get Into and/or Out of the Chip?; How Do I Develop on the Processor?; Do I Need an Operating System?; What Are the Different Ways to Benchmark a Processor?; How Much Am I Willing to Spend?; OK, So What Processor Choices Do I Have?; A Look Inside the Blackfin Processor; System View; Computational Units; Memory Model; DMA; Instruction Flow; Event Handler; Protection of Resources; Programming Model; Power Management; What's Next?; Chapter 2: Memory Systems; Introduction; Memory Spaces; L1 Instruction Memory L1 Data MemoryCache Overview; What Is Cache?; More Cache Details; External Memory; Synchronous Memory; Asynchronous Memory; What's Next?; Chapter 3: Direct Memory Access; Introduction; DMA Controller Overview; More on the DMA Controller; Programming the DMA Controller; DMA Classifications; Advanced DMA Features; System Performance Tuning; External DMA; What's Next?; Chapter 4: System Resource Partitioning and Code Optimization; Introduction; Event Generation and Handling; System Interrupts; Programming Methodology; Architectural Features for Efficient Programming Multiple Operations per CycleHardware Loop Constructs; Specialized Addressing Modes; Interlocked Instruction Pipelines; Compiler Considerations for Efficient Programming; System and Core Synchronization; Load/Store Synchronization; Ordering; Atomic Operations; Memory Architecture-The Need for Management; Memory Access Tradeoffs; Instruction Memory Management-To Cache or To DMA?; Data Memory Management; System Guidelines for Choosing Between DMA and Cache; Memory Management Unit (MMU); Physics of Data Movement; 1. Grouping Like Transfers to Minimize Memory Bus Turnarounds; Example 4.4
2. Understanding Core and DMA SDRAM Accesses

Sommario/riassunto

A key technology enabling fast-paced embedded media processing developments is the high-performance, low-power, small-footprint convergent processor, a specialized device that combines the real-time control of a traditional microcontroller with the signal processing power of a DSP. This practical guide is your one-stop shop for understanding how to implement this cutting-edge technology.You will learn how to:* Choose the proper processor for an application.* Architect your system to avoid problems at the outset.* Manage your data flows and memory accesses so that they line