PowerPC™ Microprocessor Common Hardware Reference Platform: A System Architecture

Personal Use Copy - Not for Reproduction

LICENSE INFORMATION

To the extent that Apple Computer, Inc., International Business Machines Corporation, and Motorola, Inc. (referred to as "the creators") own licensable copyrights in the *PowerPC Microprocessor Common Hardware Reference Platform: A System Architecture* (including accompanying source code samples), the creators grant you a copyright license to copy and distribute portions of this document (including accompanying source code samples) in any form, without payment to the creators, for the purpose of developing original documents, code, or equipment (except integrated circuit processors) which conform to the requirements in this document and for the purpose of using, reproducing, marketing, and distributing such code or equipment. This authorization applies to the content of this specification only and not to the referenced material. This authorization does not give you the right to copy and distribute this document in its entirety.

In consideration you agree to include for each reproduction of any portion of these documents or any derivative works the copyright notice as displayed below.

You are responsible for payment of any taxes, including personal property taxes, resulting from this authorization. If you fail to comply with the above terms, your authorization terminates.

The creators and others may have patents or pending patent applications, or other intellectual property rights covering the subject matter described herein. This document neither grants or implies a license or immunity under any of the creators or third party patents, patent applications or other intellectual property rights other than as expressly provided in the above copyright license. The creators assume no responsibility for any infringement of third party rights resulting from your use of the subject matter disclosed in, or from the manufacturing, use, lease, or sale of products described in, this document.

Licenses under utility patents of IBM® in the field of information handling systems are available on reasonable and non-discriminatory terms. IBM does not grant licenses to its appearance design patents. Direct your licensing inquiries in writing to the IBM Director of Licensing, International Business Machines Corporation, 500 Columbus Avenue, Thornwood, NY 10594.

Licenses under utility patents of Apple Computer, Inc., that are necessary to implement the specification set forth in this document are available on reasonable and non-discriminatory terms. Apple Computer, Inc. does not grant licenses to its appearance design patents. Direct your licensing inquires in writing to Mac OS Licensing Department, Apple Computer, Inc., 1 Infinite Loop, MS 305-1DS, Cupertino, CA 95014.

© Copyright Apple Computer, Inc., International Business Machines Corporation, Motorola, Inc. 1995. All rights reserved.

Note to U.S. Government Users—Documentation related to restricted rights— Use, duplication, or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Personal Use Copy - Not for Reproduction

NOTICES

The following paragraph does not apply to the United Kingdom or any country where such provisions are inconsistent with local law. In such countries, the minimum country warranties will apply.

THE CREATORS PROVIDE THIS DOCUMENT (INCLUDING ACCOMPANYING SOURCE CODE EXAMPLES) "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. THE DISCLAIMER OF WARRANTY APPLIES NOT ONLY TO THE DOCUMENT (INCLUDING ACCOMPANYING SOURCE CODE EXAMPLES) BUT ALSO TO ANY COMBINATIONS, INCORPORATIONS, OR OTHER USES OF THE DOCUMENT (INCLUDING ACCOMPANYING SOURCE CODE EXAMPLES) UPON WHICH A CLAIM COULD BE BASED.

Some states do not allow disclaimers of express or implied warranties in certain transactions; therefore, this statement may not apply to you.

These materials could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the document. The creators may make improvements and/or changes in the product(s) and/or the program(s) described in, or accompanying, this document at any time.

It is possible that this document may contain reference to, or information about, products (machines and programs), programming, or services of the creators that are not announced in your country. Such reference or information must not be construed to mean that the creators intend to announce such products, programming, or services in your country.

Requests for copies of this document or for technical information about products described herein should be directed to the creators. Refer to "Obtaining Additional Information" on page 300 for a description of information available and telephone numbers.

Version 1.0 (November 1995)

TRADEMARKS AND SERVICE MARKS

Trademarks or service marks in the United States or other countries are denoted by a registered symbol (®) or a trademark symbol(™) on their first occurrence in this document. See the section entitled "Trademark Information" on page 295 for a complete listing of all referenced trademarks and the companies that own them.

Personal Use Copy - Not for Reproduction

PowerPC™ Microprocessor Common Hardware Reference Platform: A System Architecture

Developed by Apple Computer, Inc., International Business Machines Corporation, and Motorola, Inc.

MORGAN KAUFMANN PUBLISHERS, INC. SAN FRANCISCO, CALIFORNIA

To order copies of this book, please contact the publisher at (800)745-7323 or your Apple, IBM, or Motorola contacts given in Section , “Sources for Documents,” on page 299

Sponsoring Editor: Jennifer Mann
Production Manager: Yonie Overton
Production Editor: Julie Pabst
Editorial Assitant: Jane Elliott
Cover Design: Carron Design
Printer: Courier Corporation

Morgan Kaufmann Publishers, Inc.
Editorial and Sales Office:
340 Pine Street, Sixth Floor
San Francisco, CA 94104-3205
USA
Telephone: 415/392-2665
Fasimile: 415/982-2665
Internet: mkp@mkp.com
Order toll free: 800/745-7323

This book was typeset by the authors, using FrameMaker cross-platform.

Printed in the United States of America

00 99 98 97 96 5 4 3 2 1

Library of Congress Cataloging-in-Publication Data is available for this book.

ISBN 1-55860-394-8

Personal Use Copy - Not for Reproduction

Foreword

Personal, business, and technical computing environments continue to demand ever-increasing levels of function and performance. The PowerPC™ microprocessor has been jointly developed by IBM®, Motorola®, and Apple® to meet the computer performance growth requirements of the 90s and enable a new generation of computer platform and applications with industry-leading capabilities. A prerequisite for these new computer systems to gain wide acceptance by the application and operating system development communities is a "system architecture" that provides consistent interfaces between hardware and software. The architecture maintains application software compatibility across platforms manufactured by different system vendors, and enables scalability to meet future demands. This specification satisfies these prerequisite by documenting a PowerPC-based *open system architecture* that can be used by system and software companies for the development of compatible PowerPC computer systems, subsystems, and software. The architecture is intended to support a range of PowerPC system implementations including portable, desktop, and server computer systems.

Our engineering teams, in conjunction with other industry participants, have endeavored to develop a leading-edge computer architecture that can satisfy manufacturer and customer needs into the next century. They have also expended significant effort to accommodate legacy IBM PC and Apple Macintosh® hardware and software issues, while supporting future system evolution. This focus on the future, with an eye on the past, is a key attribute of this architecture. Another key attribute is the ability, through a combination of hardware and software, for system manufacturers to differentiate their systems while maintaining application compatibility. This attribute creates greater opportunity for value to be added by system manufacturers.

We want to thank our teams and those in the computer industry who have contributed to development of this specification. Each of our companies supports making this architecture a success, and we invite each of you to join in the opportunity that this open industry architecture creates. For us, this is the beginning of an exciting new era in personal computing.

David Nagel
Senior Vice President, Apple Computer, Inc.

Robert M. Stephenson
IBM Senior Vice President and Group Executive, Personal Systems Group

Joe Guglielmi
Corporate Vice President and General Manager of Motorola Computer Group

Personal Use Copy - Not for Reproduction

Contents

Foreword vii

List of Figures xiii

List of Tables xv

About this Document xvii
- Goals of the Specification xviii
- Audience for this Document xix
- Organization of this Document xx
- Suggested Reading xxii
- Conventions Used in this Document xxiii
- Acknowledgments xxiv
- Comments on this Document xxiv

Chapter 1 Introduction 1
1.1 Platform Topology 2

Chapter 2 System Requirements 7
2.1 System Operation 7
- 2.1.1 Control Flow 7
- 2.1.2 POST 8
- 2.1.3 Boot Phase 8
- 2.1.4 Transfer Phase 11
- 2.1.5 Run-Time 12
- 2.1.6 Termination 12

2.2 Firmware 13
2.3 Bi-Endian Support 14
2.4 64-Bit Addressing Support 14
2.5 Minimum System Requirements 16
- 2.5.1 Table Description 17

2.6 Options and Extensions 20

Chapter 3 System Address Map 23
3.1 Address Areas 23
3.2 Address Decoding and Translation 28
3.2.1 Peripheral I/O Address Translation 37
3.2.2 Translation of 32-Bit DMA Addresses in 64-Bit Addressing Systems 38
3.3 PC Emulation Option 44

Chapter 4 Processor and Memory 49
4.1 Processor Architecture 49
4.1.1 Processor Architecture Compliance 50
4.1.2 PowerPC Microprocessor Differences 51
4.1.3 Processor Interface Variations 53
4.1.4 PowerPC Architecture Features Deserving Comment 53
4.2 Memory Architecture 56
4.2.1 System Memory 56
4.2.2 Storage Ordering Models 57
4.2.3 Memory Controllers 64
4.2.4 Cache Memory 65

Chapter 5 I/O Bridges 69
5.1 PCI Host Bridge (PHB) Architecture 69
5.1.1 PHB Implementation Options 70
5.1.2 Data Buffering and Instruction Queuing 70
5.1.3 Byte Ordering Conventions 74
5.1.4 PCI Bus Protocols 77
5.1.5 Programming Model 78
5.2 I/O Bus to I/O Bus Bridges 78
5.2.1 What Must Talk to What 79
5.2.2 PCI to PCI Bridges 81
5.2.3 PCI to ISA Bridges 82
5.2.4 16-Bit PC Card (PCMCIA) and Cardbus PC Card Bridges 83

Chapter 6 Interrupt Controller 85
6.1 Interrupt Controller Architecture 85
6.2 Distributed Implementation — A Proposal 86

Chapter 7 Run-Time Abstraction Services 91
7.1 RTAS Introduction 91
7.2 RTAS Environment 92
7.2.1 Machine State 93
7.2.2 Register Usage 94
7.2.3 RTAS Critical Regions 95
7.2.4 Resource Allocation and Use 96
7.2.5 Instantiating RTAS 97
7.2.6 RTAS Device Tree Properties 98
7.2.7 Calling Mechanism and Conventions 101
7.2.8 Return Codes 103
7.3 RTAS Call Function Definition 103

Personal Use Copy - Not for Reproduction

7.3.1 restart-rtas 103
7.3.2 NVRAM Access Functions 104
7.3.3 Time of Day 106
7.3.4 Error and Event Reporting 109
7.3.5 PCI Configuration Space 114
7.3.6 Operator Interfaces and Platform Control 117
7.3.7 Power Management 123
7.3.8 Suspend and Hibernate 128
7.3.9 Reboot 135
7.3.10 Caches 136
7.3.11 SMP Support 137

Chapter 8 Non-Volatile Memory 141
8.1 System Requirements 141
8.2 Structure 142
8.3 Signatures 142
8.4 Architected Partitions 144
8.4.1 Open Firmware (0x50) 144
8.4.2 Hardware (0x52) 145
8.4.3 System (0x70) 145
8.4.4 Configuration (0x71) 145
8.4.5 Error Log (0x72) 147
8.4.6 Multi-Boot (0x73) 147
8.4.7 Free Space (0x7F) 148
8.5 NVRAM Space Management 149

Chapter 9 I/O Devices 151
9.1 PCI Devices 151
9.1.1 Resource Locking 152
9.1.2 PCI Expansion ROMs 152
9.1.3 Assignment of Interrupts to PCI Devices 152
9.1.4 PCI Devices with Required Register Definitions 153
9.1.5 PCI-PCI Bridge Devices 154
9.1.6 Graphics Controller and Monitor Requirements for Clients 154
9.2 ISA Devices 155

Chapter 10 Error and Event Notification 157
10.1 Introduction 157
10.2 RTAS Error and Event Classes 158
10.2.1 Internal Error Indications 160
10.2.2 Environmental and Power Warnings 165
10.2.3 Power Management Events 167
10.3 RTAS Error and Event Information Reporting 167
10.3.1 Introduction 168
10.3.2 RTAS Error/Event Return Format 168

Chapter 11 Power Management 185
11.1 Power Management Concepts 185
11.1.1 Power Management Policy Versus Mechanism 186
11.1.2 Device Power States 187

Personal Use Copy - Not for Reproduction

11.1.3 System Power Management States 187
11.1.4 System Power Transitory States 190
11.1.5 Power Domains and Domain Control Points 191
11.1.6 Power Sources 195
11.1.7 Batteries 195
11.1.8 Power Management Events 195
11.1.9 Explicit Transfer of Power Management Policy 196
11.1.10 EPA Energy Star Compliance 197
11.2 Power-Managed Platform Requirements 197
11.2.1 Definition of Power Management Related Parameters Utilized by RTAS 198
11.2.2 Open Firmware Device Tree Properties 201
11.2.3 General Hardware Requirements 205
11.3 Operating System Requirements 210
11.3.1 General Requirements 212

Chapter 12 The Symmetric Multiprocessor Option 215
12.1 SMP System Organization 216
12.2 An SMP Boot Process 218
12.2.1 SMP-Safe Boot 219
12.2.2 Finding the Processor Configuration 220
12.2.3 SMP-Efficient Boot 222
12.2.4 Use of a Service Processor 222

Appendix A Operating System Information 223

Appendix B Requirements Summary 225

Appendix C Bi-Endian Designs 265
C.1 Little-Endian Address and Data Translation 265
C.2 Conforming Bi-Endian Designs 268
C.2.1 Processor and I/O Mode Control 268
C.2.2 Approach #1—Bi-Endian Memory and Bi-Endian I/O Design 269
C.2.3 Approach #2—Bi-Endian I/O Design 273
C.3 Software Support for Bi-Endian Operation 276
C.4 Bi-Modal Devices 276
C.5 Future Directions in Bi-Endian Architecture 279

Appendix D Architecture Migration Notes 281

Glossary 285

Trademark Information 295

Bibliography 297
Sources for Documents 299
Obtaining Additional Information 300

Index 303

Personal Use Copy - Not for Reproduction

Figures

1. Typical Desktop Topology 4
2. General Platform Topology 5
3. Phases of Operation (example) 8
4. Example of an Address Map for a 32-Bit Addressing System with One PHB 33
5. Example of an Address Map for a 32-Bit Addressing System with Two HBs 34
6. Example of an Address Map for a 64-Bit Addressing System with One PHB 35
7. Models for Load and Store Instructions to Peripheral I/O Space 38
8. I/O Device DMA Address Translation 43
9. Example Address Map with the PC Emulation Option Enabled 46
10. Example System Diagram Showing the PowerPC Ordering Domain 61
11. System Memory Map Showing Mapping of the IDU and an ISU 88
12. Layout of extended error log format from RTAS. 175
13. Example Domain and Device Dependency Relationships 193
14. Battery Condition Cycle State Transition Diagram 209
15. Example Power Management Software Structure 212
16. Bi-Endian Platform with Bi-Endian Memory and I/O 271
17. Bi-Endian Platform with Bi-Endian I/O 274
18. Bi-Endian Apertures for the Graphics Subsystem 278
19. Design with a Full Bi-Endian Processor 280

Tables

i. Typographical Conventions xxiii
1. I/O Device Reset States 10
2. Summary of Minimum Platform Requirements 19
3. Valid Hardware Combinations of Compatibility Holes and Initial Memory Alias Spaces 26
4. Map Legend 26
5. Processor Bus Address Space Decoding and Translation 32
6. DMA Address Decoding and Translation (I/O Bus Memory Space) 32
7. TCE Definition 44
8. Fixed Real Storage Locations Having Defined Uses 52
9. *Load* and *Store* Programming Considerations 76
10. Which I/O Devices Must Be Able to Access Which Address Spaces 79
11. *instantiate-rtas* Argument Call Buffer 98
12. RTAS Tokens for Functions 99
13. Open Firmware Device Tree Properties 100
14. RTAS Argument Call Buffer 101
15. RTAS Status Word Values 103
16. *restart-rtas* Argument Call Buffer 104
17. *nvram-fetch* Argument Call Buffer 105
18. *nvram-store* Argument Call Buffer 106
19. *get-time-of-day* Argument Call Buffer 107
20. *set-time-of-day* Argument Call Buffer 108
21. *set-time-for-power-on* Argument Call Buffer 109
22. *event-scan* Argument Call Buffer 111
23. *check-exception* Argument Call Buffer 113
24. Additional Information Provided to *check-exception* call 113
25. *read-pci-config* Argument Call Buffer 115
26. *write-pci-config* Argument Call Buffer 116
27. *display-character* Argument Call Buffer 118
28. *set-indicator* Argument Call Buffer 119
29. Defined Indicators 120

30. *get-sensor-state* Argument Call Buffer 121
31. Defined Sensors 121
32. *set-power-level* Argument Call Buffer 124
33. *get-power-level* Argument Call Buffer 125
34. *assume-power-management* Argument Call Buffer 126
35. *relinquish-power-management* Argument Call Buffer 127
36. *power-off* Argument Call Buffer 128
37. *suspend* Argument Call Buffer 129
38. *hibernate* Argument Call Buffer 132
39. Format of Block List 132
40. *system-reboot* Argument Call Buffer 135
41. *cache-control* Argument Call Buffer 136
42. *Cache-control* states 137
43. *freeze-time-base* Argument Call Buffer 138
44. *thaw-time-base* Argument Call Buffer 138
45. *stop-self* Argument Call Buffer 139
46. *start-cpu* Argument Call Buffer 140
47. NVRAM Structure 143
48. NVRAM Signatures 144
49. Multi-Boot String Definitions 148
50. Error and Event Classes with RTAS Function Call Mask 159
51. Error Indications for System Operations 162
52. EPOW Action Codes 166
53. RTAS Error Return Format (Fixed Part) 172
54. RTAS General Extended Error Log Format 176
55. Error Log Detail for CPU-Detected Error 178
56. Error Log Detail for Memory Controller-Detected Error 178
57. Error Log Detail for I/O-Detected Error 180
58. Error Log Detail for Power-On Self Test-Detected Error 182
59. Event Log Detail for Environmental and Power Warnings 183
60. Event Log Detail for Power Management Events 183
61. Defined Power Levels 198
62. Defined Power Management Event Types 199
63. Sources of Operating Systems Information 223
64. Address Modification for Little-Endian Mode 266
65. Bytes Accessed Versus Endian Mode 267
66. Rules for Byte Reversal and Address Modification 269
67. Endian Mode Data Byte Reversal 270
68. Byte Reversal, Unequal Bus Widths 270
69. PowerPC Reference Platform Specification Evolution 282

Personal Use Copy - Not for Reproduction

About this Document

The purpose of this document is to define an open architecture and minimum system requirements that enable the development and manufacture of industry-standard computer systems based on PowerPC microprocessors. These requirements are intended to be precise enough to assure application software compatibility for several operating system environments, broad enough to cover portables through server platforms in single or multiprocessor configurations, and forward-looking enough to allow evolution, including 64-bit addressing.

These requirements were developed by Apple Computer, Inc., International Business Machines Corporation, and Motorola, Inc. to define a system which is intended to become the pervasive open industry standard for single-user portable through multi-user server configurations. Systems built to these requirements will have PowerPC microprocessor(s) and will share components with the Apple Macintosh family and IBM compatible personal computers. These systems will be capable of running various native operating systems. The set of operating systems anticipated to be available includes Apple Mac™ OS, IBM AIX™, and PowerPC™ editions of IBM OS/2™ Warp Connect, Microsoft Windows NT™ Workstation, Novell NetWare™, and SunSoft Solaris™. Most applications for these operating environments may be run on these systems either in native mode or through some emulation capability. With the appropriate operating system and x86 emulation support, these systems will be capable of running Windows™ and DOS applications.

Within the context of this document, “architecture” is defined as the specification of the interface between the hardware platform and the operating systems and applications. Device drivers also use this architecture, but require additional definition of the device interfaces to the hardware and operating system interfaces within the software.

To the extent that firmware abstracts the hardware interface, it becomes part of the hardware. The firmware to operating system interface is defined in this architecture. Two types of firmware are discussed here. Open Firmware is the initialization or boot code that controls the platform prior to the transfer of control to the operating system. Run-Time Abstraction Services (RTAS) is the run-time firmware which provides abstractions to the executing operating system. Interfaces within the software or within the hardware are not defined in this document. Where necessary, reference will be made to documents where those definitions can be found.

Within the context of this document the term “the architecture” or “the CHRP architecture” is used to refer to the requirements contained in this document, *PowerPC Microprocessor Common Hardware Reference Platform: A System Architecture*.

Goals of the Specification

The specific goals of this specification are as follows:

- To create an open industry standard to be used for the implementation of PowerPC based systems. The architecture document is available to the industry and can be used by any hardware or software vendor to develop compliant products.
- To allow compatible differentiation through the use of abstracted hardware interfaces, defined minimum hardware, and extension mechanisms.
- To leverage existing and future industry-standard buses and interfaces. Existing bus architectures have a proven level of performance and functionality. Established industry-standard interfaces—for example SCSI, IDE, LocalTalk®, Ethernet™, etc.— and newer bus architectures, interfaces and protocols —for example PCI, PC Card, IrDA, etc.— provide higher levels of performance or utility not achievable by the older standards. The architecture allows platform and system designers to determine which buses, interfaces, and protocols best suit their target environment.
- To provide a flexible address map. Another key attribute of this specification is the relocatability of devices and subsystems within the PowerPC address space. Subsystem address information, which defines where I/O devices reside, is detected by the Open Firmware and passed to the operating systems in the device tree. The architecture accommodates the use of multiple identical buses and adapters in the same platform without address conflicts.

Personal Use Copy - Not for Reproduction

- To build upon the Open Firmware boot environment defined in IEEE 1275, *IEEE Standard for Boot (Initialization Configuration) Firmware, Core Requirements and Practices* [9]. Currently the abstraction approach for some operating systems uses platform description information discovered by a legacy boot process and passed to the operating system in data structures. With these systems, operating systems and platforms will migrate to the Open Firmware boot process and device tree.
- To architect the control of power management by different operating system. It is important that the combination of hardware and software be allowed to minimize power consumption through automatic or programmed power-saving methods. Power management of systems will reduce the operational cost for the user and reduce the impact of the system on the environment.
- To provide an architecture which can evolve as technology changes. The creators of the architecture invite industry participation in evolving future versions of it.
- To minimize the support cost for multiple operating systems through the definition of common platform abstraction techniques. Common and compatible approaches to the abstraction of hardware will reduce the burden on hardware vendors who produce differentiated machines.
- To architect a mechanism for error handling, error reporting, and fault isolation. The architecture provides for the implementation of more robust systems if desired by the system developers.

Audience for this Document

This document defines the Common Hardware Reference Platform (CHRP™) architecture and system requirements for building standard systems. This document is the primary source of information that a hardware platform, operating system, or hardware component developer would need to create compatible products. Additional requirements are defined by the industry standards referenced in this document.

This document must be used by those building CHRP hardware platforms. The document describes the hardware to operating system interface which must be provided in these platforms. Platform designers must assemble components and firmware which match this interface. Also, the document defines minimum system configuration requirements. Platform designers must meet or exceed these minimums to build a standard platform. The operating systems which are expected to support this architecture are listed in Appendix A,

Personal Use Copy - Not for Reproduction

"Operating System Information," on page 223. Each operating system has configuration requirements which are described in documents maintained by the owners of the operating systems. Using these requirements, a hardware platform designer may target a configuration to support multiple operating systems.

This document must be used by those building compatible software including operating systems, boot software, or firmware. If a function is supported, software developers must provide support for the interfaces described in this document. This software must provide the mandatory functions and capabilities as described in the requirements in this document. However, this document does not limit this software from going beyond the specification through software tailored to specific hardware. For example, an operating system must implement the interface to the required abstracted interfaces to hardware, but the operating system may also implement fast paths to specific hardware that it recognizes.

Sample implementations of CHRP platforms will be developed and their specifications will be published by several companies. Platform designers may use these descriptions as a reference for developing clone designs, or may use these descriptions in conjunction with this document to develop unique compliant platform designs.

Organization of this Document

The following chapters provide the detailed requirements for system developers to build compliant systems. The chapters cover all needed topics from minimum system requirements to power management. Where appropriate, references are made to other industry standards or readily available reference documentation for the required architecture information.

Following is a summary and brief description of the chapters and appendices of this document:

- Chapter 1, "Introduction," on page 1 describes the advantages of the architecture and shows typical platform topologies.
- Chapter 2, "System Requirements," on page 7 describes the system operation and 64 Bit Addressing support. Firmware, Bi-Endian, and minimum platform requirements are documented. The minimum platform requirements are given for three types of platforms. The Bi-Endian capability of these platforms is one key feature which allows the multitude of operating systems.

Personal Use Copy - Not for Reproduction

- Chapter 3, “System Address Map,” on page 23 describes the address map and the requirements for the address map. The chapter describes some address map extensions to support PC emulation.
- Chapter 4, “Processor and Memory,” on page 49 defines the required PowerPC microprocessor attributes for processors included in these platforms and provides specific requirements for the memory subsystem.
- Chapter 5, “I/O Bridges,” on page 69 defines the requirements for host bridges and I/O bridges. “Host Bridges” are the interface to the processor bus and the I/O bus and “I/O bridges” are any additional interfaces to other I/O buses.
- Chapter 6, “Interrupt Controller,” on page 85 defines the required interrupt controller for standard platforms.
- Chapter 7, “Run-Time Abstraction Services,” on page 91 defines the services which provide an abstract interface to some hardware components. These services provide to the operating systems a platform-independent interface for hardware components.
- Chapter 8, “Non-Volatile Memory,” on page 141 defines the NVRAM content and structure to be used on these platforms. NVRAM contains information such as the platform configuration and error logs that must persist across platform power cycles.
- Chapter 9, “I/O Devices,” on page 151 defines the requirements for I/O devices and references another document that defines the specific registers used to address these devices.
- Chapter 10, “Error and Event Notification,” on page 157 defines platform error reporting requirements and describes the use of RTAS services for obtaining error and event information on these platforms.
- Chapter 11, “Power Management,” on page 185 describes the anticipated power management approach. For systems which implement power management, this chapter defines the requirements that software and hardware must meet.
- Chapter 12, “The Symmetric Multiprocessor Option,” on page 215 gives those requirements specifically for symmetric multiprocessor platforms and describes the boot process for symmetric multiprocessors.
- Appendix A, “Operating System Information,” on page 223 lists sources of additional information for targeted operating systems.

Personal Use Copy - Not for Reproduction

- Appendix B, “Requirements Summary,” on page 225 is a complete list of all requirements imbedded in a chapter level table of contents. This appendix is a useful summary or check list for those implementing standard platforms.
- Appendix C, “Bi-Endian Designs,” on page 265 provides some explanation of endianess and the design approaches for implementing Bi-Endian platforms.
- Appendix D, “Architecture Migration Notes,” on page 281 provides information describing creation of this architecture from the legacy systems.

Suggested Reading

The “Bibliography” on page 297 provides a full list of references and ordering information for these references. Within this document, the number of the reference in the bibliography is placed after the citation in brackets, “[nn]”.

This document assumes the reader has an understanding of computer architecture, the PowerPC microprocessor architecture, the current PowerPC processors, Apple and IBM compatible personal computers, Peripheral Component Interconnect (PCI) local bus, and Open Firmware. Some understanding of the current personal computer and workstation architectures is also useful. A list of suggested background reading includes:

- *Computer Architecture: A Quantitative Approach* [2]
- *The PowerPC Architecture: A Specification for a New Family of RISC Processors* [1]. Note that this specification is referred to as *The PowerPC Architecture* in the body of this document.
- *PowerPC 603 RISC Microprocessor User’s Manual* [4]
- *PowerPC 603 RISC Microprocessor Technical Summary* [5]
- *PowerPC 604 RISC Microprocessor User’s Manual* [6]
- *PowerPC 604 RISC Microprocessor Technical Summary* [7]
- *Technical Introduction to the Macintosh Family* [26]
- *PowerPC Reference Platform Specification,* Version 1.1 [28]
- *PCI Local Bus Specification*, Revision 2.1 [14]
- IEEE 1275, *IEEE Standard for Boot (Initialization Configuration) Firmware, Core Requirements and Practices* [9] and relevant bindings

Personal Use Copy - Not for Reproduction

Conventions Used in this Document

Within the body of this document lists of requirements are clearly defined. The convention used is to head each requirements list with the word "**Requirements:**" in bold face type. Following this heading are one or more requirements. These requirements may point to other standards documents, or figures or tables which conveniently show the requirement. The referenced material becomes part of the requirements in this document. Users of this document must comply with these requirements to build a standard platform. Other material in this document is supportive description of these requirements, architecture notes, or implementation notes. Architecture or implementation notes are flagged with a descriptive phrase—for example, "Hardware Implementation Note"— and followed by indented paragraphs. The descriptive material and notes provide no additional requirements and may be used for their information content.

Big-Endian numbering of bytes and bits is used in this document, unless indicated otherwise.

Typographical conventions used in this document are described in Table on page xxiii.

Table 1. Typographical Conventions

Text Element	Description of Use
Italics	Used for emphasis such as the first time a new term is used. Indicates a book title. Indicates PowerPC instruction mnemonics. Indicates Open Firmware properties, methods, and configuration variables. Indicates RTAS function and parameter names
0xnnnn	Prefix to denote hexadecimal numbers.
0bnnnn	Prefix to denote binary numbers.
nnnnn	Numbers without a prefix are decimal numbers.
0xF. . . FFF100	This hexadecimal notation represents a replication of the hexadecimal character to the right of the ellipsis to fill out the field width. For example, the address 0xF . . . FFF100 would be 0xFFFFF100 on a processor with a 32-bit address bus or 0xFFFFFFFFFFFFF100 on a processor with a 64-bit address bus.
0:9	Ranges of bits are specified by two numbers separated by a colon. The range includes the first number, all numbers in between, and the last number.
0xm-0xn	A range of addresses or values within the document is always inclusive, from m up to and including n.

Personal Use Copy - Not for Reproduction

Acknowledgments

This document has come together through the work of many people in several companies. The primary responsibility for writing sections fell on employees of Apple, IBM, and Motorola. Industry review was solicited and comments were received from such companies as Advanced Micro Devices, Canon, FirePower, National Semiconductor, and Toshiba.

Not everyone who worked on this document can be mentioned here due to space limitations. The contributions of some key individuals are worth mentioning. The document was brought together by a team of engineers who merged existing material with new material and spent countless hours reviewing and discussing this document. This team included Art Adkins, Richard Arndt, Stanford Au, Don Banks, Jerry Barrett, Rob Baxter, Rich Bealkowski, Mike Bell, Doug Bossen, Jim Brennan, John Bruner, Steve Bunch, Pat Carr, Leo Clark, Ron Clark, Bob Coffin, Clayton Cole, Scott Comer, Gary De Angelis, Sanjay Deshpande, Brad Frey, Jim Gable, Bill Galcher, Brian Hansche, Ron Hochsprung, Kohichi Kii, John Kingman, Steve MacKenzie, Don McCauley, Andy McLaughlin, Todd Moore, Dan Neal, Luan Nguyen, Jim Nicholson, John O'Quin, Mike Paczan, Ray Pedersen, Charlie Perkins, Patrick Perrino, Dave Peterson, Greg Pfister, Craig Prouse, Andy Rawson, Paul Resch, Joe St. Clair, Kanti Shah, Fred Strietelmeier, Howard Tanner, M. Teodorovich, Don Thorson, Steve Thurber, Dave Tjon, Abraham Torres, George Towner, Ted Toyokawa, Tom Tyson, and Lee Wilson.

Comments on this Document

Comments on this document are welcome. Because of time and resource constraints, we will not always be able to provide responses to general industry comments. We will review your comments for possible incorporation in future versions of the specification. All comments become the property of Apple, IBM, and Motorola and may be used for any purpose whatsoever. For this reason, comments must not contain any proprietary data. Please preface your comments with the statement "These comments do not contain confidential or proprietary information and may be used for any purpose". Comments may be addressed to:

info-hrp@austin.ibm.com

Personal Use Copy - Not for Reproduction