OPEN PLATFORM MANAGEMENT ARCHITECTURE

SPECIFICATION LICENSE AGREEMENT

This Open Platform Management Architecture Specification License Agreement ("Agreement") is made by Advanced Micro Devices, Inc., a Delaware corporation, having its principal place of business at One AMD Place, Sunnyvale, California, 94088 (“AMD”), and the entity named on the signature page of this Agreement (“Adopter”). This Agreement is effective as of the date a fully executed original has been received by AMD (“Effective Date”).

BACKGROUND

A. AMD has developed a platform architecture which is designed to modularize the platform hardware manageability subsystem for servers, the Open Platform Management Architecture. AMD desires to provide the OPMA (defined below) specification to various vendors and suppliers in a manner that enables compatibility between different vendors’ hardware and software designs implementing OPMA.

B. AMD is the owner or licensee of certain intellectual property rights covering OPMA and the OPMA Specification and is willing to license such rights to Adopter under the terms and conditions set forth in this Agreement.

NOW, THEREFORE, in consideration of the mutual covenants and conditions contained herein and for other good and valuable consideration, the receipt and sufficiency of which are hereby acknowledged, and intending to be legally bound hereby, the parties hereto agree as follows:

AGREEMENT

Article 1. Ownership and License Grants

1.1 Ownership. Adopter hereby acknowledges and agrees that AMD, AMD Affiliates, and AMD’s licensors shall retain ownership of all worldwide rights, titles, and interests in and to OPMA and the OPMA Specification, and all related intellectual property rights embodied therein, subject to the licenses granted to Adopter in Section 1.2 below.

1.2 Licenses. As of the Effective Date, the following licenses are granted by Adopter to AMD and all Fellow Adopters (as provided in Section 1.2.2), and the grants of AMD and all Fellow Adopters shall extend to Adopter. In each case, the party (AMD, Adopter, or Fellow Adopter) granting the license is referred to as the “Grantor,” and the party (AMD, Adopter, or Fellow Adopter) receiving the license is referred to as the “Grantee.” “Fellow Adopters” are all other entities which have executed, at any time, an agreement substantially similar to this Agreement and delivered it to AMD.

1.2.1 OPMA. Subject to the terms and conditions of this Agreement, Grantor hereby grants to Grantees a worldwide, non-exclusive, royalty free, non-transferable license under Grantor’s Necessary Claims, to make, have made, use, import, sell, offer to sell, lease, and otherwise dispose of Compliant Portions, provided that such license shall not extend to any part or function of a product that is itself not part of a Compliant Portion.

1.2.2 Extension to Fellow Adopters. AMD may extend the licenses set forth in this Section 1.2 to any third party Fellow Adopter subject to the terms and conditions set forth in this Agreement, including but not limited to those set forth in Section 1.3 below.

1.3 No Implied Licenses. Except as expressly provided in this Agreement, no other rights are granted by Grantor hereunder by implication, estoppel or otherwise. All rights not expressly granted by Grantor are reserved to Grantor.

Article 2. Usage of Adopter Name

2.1 Use of Adopter Name. AMD shall have the right to include Adopter's name in any lists, published by AMD, of entities licensing OPMA.

Article 3. Confidentiality

3.1 Nondisclosure. The parties acknowledge that it may be necessary for AMD to disclose to Adopter certain confidential and/or proprietary information ("Confidential Information") to effectuate the purpose of this Agreement. Confidential Information shall include (a) information provided by AMD to Adopter hereunder that AMD identifies in writing as confidential or proprietary information, (b) any information provided hereunder, whether orally or in writing, which Adopter knows or has reason to know is Confidential Information of AMD, and (c) until made publicly available by AMD, the OPMA Specification and this Agreement. Adopter acknowledges that all rights to Confidential Information are reserved by AMD, and unless otherwise provided in this Agreement, Adopter may not disclose or disseminate such Confidential Information to anyone other than its employees or contractors with a need to know such Confidential Information for purposes of this Agreement. Adopter will protect the Confidential Information by using the same degree of care, but no less than a reasonable degree of care, to prevent the unauthorized use, dissemination, or disclosure of the Confidential Information as it uses to protect its own confidential information of a like nature.

3.2 Feedback. Adopter may, but is not obligated to, provide AMD with feedback, suggestions, and improvements regarding the OPMA Specification and drafts thereof for possible incorporation by AMD or its designee into future versions of the OPMA Specification. Such feedback, suggestions, and improvements provided to AMD or its designee in writing or through an Electronic Feedback Forum (defined below in Section 3.3) shall be referred to herein as "Feedback." AMD or its designee shall be free to incorporate any and all Feedback into the OPMA Specification, and reserves the right to determine, in its sole discretion, the contents of the OPMA Specification. Adopter hereby grants to AMD, and AMD accepts, a non-exclusive, irrevocable, perpetual, worldwide, transferable, royalty-free license, with the right to sublicense, under Adopter's intellectual property rights in and to any and all Feedback provided by Adopter to AMD hereunder and incorporated into the OPMA Specification (i) to use, copy, create derivative works of, publicly display, publicly perform, and distribute such Feedback as part of the OPMA Specification; and (ii) to use, copy, create derivative works of, publicly display, publicly perform, distribute, make, have made, sell, have sold, import and otherwise dispose of such Feedback in Compliant Portions, without attribution or reference to source. The licenses granted under this Section 3.2 shall survive any termination of this Agreement.

3.3 Information Exchange Forums. In order to promote widespread adoption of OPMA and facilitate communication among various vendors and suppliers licensed thereto, AMD, may, but is not obligated to, develop or maintain one or more forums for the electronic interchange of information between Adopter, AMD and other Fellow Adopters ("Electronic Forums"). Electronic Forums may include, but are not limited to, users groups, web sites, and mailing lists. One or more of such forums may be dedicated for the purpose of receiving Feedback ("Electronic Feedback Forums"). Access to and use of the Electronic Forums may be restricted and subject to additional terms and conditions provided by AMD. Until such time and to the extent that AMD deems otherwise, all information disclosed in the Electronic Forums shall be considered Confidential Information.

3.4 Exceptions. The obligations of confidentiality set forth in Article 3 of this Agreement shall not apply to information that: (a) was in Adopter's possession without confidentiality restriction prior to disclosure hereunder; (b) has become publicly known through no wrongful act of Adopter; (c) has come into the possession of Adopter without confidentiality restrictions from a third party and such third party is under no obligation to AMD to maintain the confidentiality of such information; (d) was developed by Adopter independently of and without reference to any Confidential Information disclosed by AMD hereunder; or (e) has been approved for release by written authorization of AMD.

3.5 Disclosure to Employees, Third Parties. Adopter shall have obtained the execution of proprietary non-disclosure agreements with its employees having access to Confidential Information, which agreements shall contain obligations at least as restrictive as the restrictions contained herein, and shall diligently enforce such agreements. Prior to its publication by AMD, Adopter may provide a copy of the OPMA Specification and an unsigned copy of this Agreement to a third party only if such third party has executed a written agreement with Adopter that prevents disclosure and unauthorized use of Confidential Information by that third party, and obligates such third party to protect such Confidential Information as set forth in Section 3.1.

3.6 Duration. The obligation of confidentiality will survive any termination of this Agreement and will expire five (5) years following receipt of the Confidential Information.

3.7 Remedies. If Adopter breaches any of its obligations with respect to the non-disclosure or unauthorized use of AMD's Confidential Information, AMD shall be entitled to seek equitable relief to protect its interest therein, including but not limited to injunctive relief, as well as money damages.

Article 4. Disclaimer of Warranties

4.1 Disclaimer of Warranties. ALL MATERIAL, INFORMATION, AND LICENSES PROVIDED BY AMD AND FELLOW ADOPTERS TO ADOPTER HEREUNDER (INCLUDING, WITHOUT LIMITATION, THE OPMA SPECIFICATION), AND ALL LICENSES PROVIDED BY ADOPTER TO AMD AND FELLOW ADOPTERS HEREUNDER, ARE PROVIDED ON AN "AS IS" BASIS, WITH NO WARRANTIES WHATSOEVER, WHETHER EXPRESS, IMPLIED, STATUTORY, CONTRACTUAL OR OTHERWISE, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, OR NON-INFRINGEMENT, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION, OR SAMPLE. FURTHERMORE, NO WARRANTY OR REPRESENTATION IS MADE OR IMPLIED RELATIVE TO THE VALIDITY OR ENFORCEABILITY OF ANY PATENT LICENSED HEREUNDER, OR RELATIVE TO FREEDOM FROM INFRINGEMENT OF ANY THIRD PARTY PATENTS.

Article 5. Exclusion of Damages; Limitations of Liability

5.1 Exclusion of Damages. IN NO EVENT WILL EITHER PARTY OR ITS AFFILIATES BE LIABLE TO THE OTHER OR TO ANY THIRD PARTY FOR ANY INDIRECT, PUNITIVE, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH OR ARISING OUT OF THIS AGREEMENT (INCLUDING LOSS OF PROFITS, USE, DATA OR OTHER ECONOMIC ADVANTAGE), HOWEVER IT ARISES, WHETHER FOR BREACH OF THIS AGREEMENT, INCLUDING BREACH OF WARRANTY, OR IN TORT (INCLUDING NEGLIGENCE), EVEN IF THAT PARTY HAS BEEN PREVIOUSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. The foregoing shall not apply, however, to waive any remedy otherwise available to AMD for injury suffered or to be suffered by AMD as a result of Adopter's breach of Article 3 of this Agreement.

5.2 Limitation of Liability of AMD. IF, AT ANY TIME, AMD OR ANY OF ITS AFFILIATES SHALL HAVE ANY LIABILITY ARISING FROM OR BY VIRTUE OF THIS AGREEMENT, AND THE PROVISIONS FOR EXCLUSION OF DAMAGES UNDER SECTION 5.1 OF THIS AGREEMENT DO NOT APPLY, AND WHETHER SUCH LIABILITY IS DUE TO AMD'S OR ITS AFFILIATE'S NEGLIGENCE, BREACH OF ITS OBLIGATIONS UNDER THIS AGREEMENT, OR OTHERWISE, ADOPTER AGREES THAT IN NO EVENT WILL THE TOTAL AGGREGATE LIABILITY OF AMD AND ITS AFFILIATES FOR ANY CLAIMS, LOSSES, OR DAMAGES INCURRED BY ADOPTER OR ANY FELLOW ADOPTER EXCEED $1,000. THIS LIMITATION OF LIABILITY IS COMPLETE AND EXCLUSIVE, SHALL APPLY EVEN IF AMD OR ITS AFFILIATES HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH POTENTIAL CLAIMS, LOSSES, OR DAMAGES, AND SHALL APPLY REGARDLESS OF THE SUCCESS OR EFFECTIVENESS OF ANY OTHER REMEDIES POSSESSED BY ADOPTER, ITS CUSTOMERS, OR ANY THIRD PARTIES. THIS LIMITATION OF LIABILITY REFLECTS AN AGREED ALLOCATION OF RISK BETWEEN AMD AND ADOPTER IN VIEW OF THE NATURE OF THIS TRANSACTION.

Article 6. Termination

6.1 Term. This Agreement shall commence upon the Effective Date, and shall continue until terminated as provided herein.

6.2 Termination By Adopter. Adopter may terminate this Agreement at any time upon giving AMD written notice of termination. Termination under this Section 6.2 shall be effective as of the date that AMD receives such written notice of termination from Adopter. After the effective date of termination, the licenses granted by Adopter to AMD under Sections 1.2 and 3.2 shall terminate, except as provided in Section 6.4 below or elsewhere in this Agreement.

6.3 Termination By AMD. AMD may terminate this Agreement upon providing Adopter with written notice of termination if Adopter is in material breach of this Agreement and Adopter fails to cure such breach within thirty (30) days after receiving notice to Adopter of such breach and AMD's intention to terminate. After the effective date of termination, the license granted by AMD under Section 1.2 shall terminate, except as provided in Section 6.4 of this Agreement.

6.4 Effect of Termination. Notwithstanding termination of this Agreement by either party for any reason, and subject to the limitations set forth in Section 1.3, the parties acknowledge that the licenses granted under Sections 1.2 and 3.2 shall remain in full force and effect: (a) for the version of the OPMA Specification set forth in Exhibit A and any versions of the OPMA Specification published by AMD more than sixty (60) days prior to the effective date of termination and (b) for any Feedback provided by Adopter prior to the effective date of termination.

6.5 Survival. All rights and obligations of the parties hereunder shall cease upon termination or expiration of this Agreement, except as provided in Section 6.4, and except the obligations in Articles 3, 4, 5, 6 and 7, which shall survive any termination or expiration of this Agreement. No termination, other than a termination for cause, shall relieve either party from the performance of any of its responsibilities or obligations that should have been performed prior to such termination.

Article 7. Miscellaneous

7.1 Governing Law; Jurisdiction; Venue. The laws of the State of California will govern this Agreement without reference to conflicts of law principles. Jurisdiction and venue for all disputes relating to this Agreement shall lie with the state and federal courts located in Santa Clara County, California.

7.2 No Obligation to Enforce. Nothing contained in this Agreement shall be construed as imposing on either party any obligation to institute any suit or action for infringement of any of its intellectual property rights, or to defend any suit or action brought by a third party which challenges or concerns the validity of any of its intellectual property rights licensed under this Agreement, or to file any patent application or to secure any patent or maintain any patent in force.

7.3 Press Release. AMD may issue press releases regarding the parties' relationship and the nature of this Agreement with Adopter's prior written approval, which shall not be unreasonably withheld, at any time following the execution of this Agreement.

7.4 Compliance With Export Laws. The parties each agree to comply with all U.S. export laws in connection with the marketing, sale and distribution of products licensed from the other party hereunder, including without limitation the Export Administration Regulations administered by the U.S. Department of Commerce and the International Traffic in Arm Regulations administered by the U.S. Department of State.

7.5 No Support. Adopter acknowledges and agrees that other than providing Adopter with the OPMA Specification, AMD is under no obligation to provide additional materials or support to Adopter.

7.6 Relationship of the Parties. The parties are independent contractors under this Agreement and no other relationship is intended, including a partnership, franchise, joint venture, agency, employer/employee, fiduciary, master/servant relationship, or other special relationship. Neither party shall act in a manner that expresses or implies a relationship other than that of independent contractor, nor bind the other party.

7.7 Notices. Unless otherwise provided herein, all notices which shall be given by either party under the provisions of this Agreement shall be in writing and be hand delivered, sent by facsimile transmission followed with written confirmation by mail, sent by commercial overnight delivery, or sent by certified U.S. mail (return receipt requested). Notices shall be deemed given on the date of actual receipt (or refusal of delivery) when hand delivered, upon confirmed transmission when sent by facsimile, one day after having been sent when sent by commercial overnight delivery, and three days after having been mailed when sent by certified U.S. mail. Notwithstanding anything to the contrary in this Section 7.7, any written notice will be effective no later than the date actually received. Notices shall be addressed as set forth in the signature page hereto.

7.8 Notice of Publication. AMD shall, at its sole discretion, provide Adopter with notice of publication for each future version of the OPMA Specification. Such notice may be sent to Adopter at the address, and in the manner, described in Section 7.7. Alternatively, AMD may provide email notification of publication to Adopter at the email address provided on the signature page hereto.

7.9 Entire Agreement; Amendment. This Agreement (including any attached exhibits) constitutes the final and entire agreement between the parties, and supercedes all prior written and oral agreements, understandings, or communications with respect to the subject matter of this Agreement (including without limitation any memorandums of understanding, written proposals, and term sheets). Notwithstanding the foregoing, nothing in this Agreement shall be deemed to limit the scope of any license granted in any prior agreement executed between the parties that is of a broader scope than the licenses granted hereunder. This Agreement may not be modified except in writing signed by a duly authorized representative of each party. It is expressly understood and agreed that no employee, agent, or other representative of AMD has any authority to bind AMD with respect to any statement, representations, warranty, or other expression unless the same is specifically set forth in this Agreement. It is also understood and agreed that no usage of trade or other regular practice or method of dealing between the parties

hereto shall be used to modify, interpret, supplement, or alter in any manner the terms of this Agreement.

7.10 No Waiver. The waiver by either party of any breach of any provision of this Agreement shall not operate or be construed as a waiver of any other or a subsequent breach of the same or a different provision.

7.11 Assignment. Neither party shall assign, transfer, or otherwise delegate any of its rights, duties, or obligations under this Agreement in whole or in part to any individual, firm or corporation without the prior written consent of the other party, which consent shall not be unreasonably withheld; provided however, that (a) AMD may assign this Agreement, in whole or in part, to one or more of its Affiliates or (b) AMD may assign this Agreement to a trade association or similar entity. Any attempt to assign, transfer or otherwise delegate any of the rights, duties, or obligations under this Agreement without the prior written consent of the other party shall be void. Notwithstanding the foregoing, either party may assign its rights, duties, and obligations hereunder without approval of the other party to a party that succeeds to all or substantially all of its assets (whether by sale, merger, operation of law or otherwise), provided that such assignee or transferee agrees in writing to be bound by the terms and conditions of this Agreement. The rights and liabilities of the parties under this Agreement will bind and inure to the benefit of the parties' permitted assigns and successors.

7.12 Captions. The captions appearing in this Agreement have been inserted as a matter of convenience and in no way define, limit or enlarge the scope of this Agreement or any of the Sections thereto.

7.13 Severability. In the event that any one or more of the provisions of this Agreement is determined by a court of competent jurisdiction to be invalid, unenforceable or illegal, such invalidity, unenforceability or illegality shall not affect any other provisions of this Agreement, and the Agreement shall be construed as if the challenged provision had never been contained herein. The parties further agree that in the event such provision is an essential part of this Agreement, they will immediately begin negotiations for a suitable replacement provision.

7.14 Force Majeure. Neither party will be deemed in default of this Agreement to the extent that performance of its obligations or attempts to cure any breach are delayed or prevented by reason of any act of God, fire, natural disaster, accident, act of government, shortages of material or supplies or any other cause beyond the control of such party, provided that such party gives the other party written notice thereof promptly and, in any event, within thirty (30) days of discovery thereof and uses good faith efforts to so perform or cure. In the event of such a Force Majeure, the time for performance or cure will be extended for a period equal to the duration of the Force Majeure but not in excess of six (6) months.

7.15 Binding. This Agreement shall be binding on the parties, their Affiliates, subsidiaries, successors, and assigns (if any), and they each warrant that the signatories hereto are authorized to execute this Agreement on behalf of the respective party.

7.16 No Bias. This Agreement shall be interpreted as written and negotiated jointly by the parties. It shall not be strictly construed against either party, regardless of the actual drafter of the Agreement.

7.17 Costs, Attorneys' Fees, and Experts' Fees. In the event any obligation of this Agreement must be enforced, through litigation or otherwise, the prevailing party will be entitled to recover reasonable costs and expenses incurred in enforcing the obligation, including costs, reasonable attorneys' fees and experts' fees.

7.18 Counterparts and Facsimile. This Agreement may be executed in duplicate and either copy or both copies are considered originals, but all of which together constitute one and the same instrument. This Agreement may be executed by facsimile signature.

7.19 Expenses. Each of the parties shall bear its own costs and expenses incurred hereunder, including, without limitation, travel, employee compensation, and incidental expenses.

Article 8. Glossary

8.1 **"Affiliate"** means an entity that directly or indirectly Controls or is Controlled by, or is under common Control with another entity, so long as such Control exists. For purposes of this Section 8.1, "Control" means control or ownership of (a) more than fifty percent (50%) of an entity's outstanding shares or stock entitled to vote for the election of directors or similar managing authority of that entity, or (b) in the case of an entity not having outstanding shares or securities, more than fifty (50%) of the right to make the decisions for that entity.

8.2 **"Compliant Portion"** means only those specific portions of products (hardware, software or combinations thereof) that: (a) implement and are compliant with all relevant portions of an OPMA Specification, and (b) are within the bounds of the Scope.

8.3 **"Feedback"** has the meaning given in Section 3.2 of this Agreement.

8.4 **"OPMA"** means the electrical, mechanical, and firmware interfaces as described in the OPMA Specification.

8.5 **"OPMA Specification"** means the specification attached hereto as Exhibit A setting forth the description and requirements (including, but not limited to, logical and electrical specifications) for OPMA, and any future version of such specification published by AMD under the title "Open Platform Management Architecture Specification."

8.6 **"Necessary Claims"** of a party means those claims of all patents and patent applications throughout the world to which such party or its Affiliates has the right at any time during the term of this Agreement to grant licenses without such grant resulting in payment of royalties or other consideration to third parties (except for payments to Affiliates or employees), which claims are (a) necessarily infringed by an implementation of the OPMA Specification and (b) within the bounds of the Scope. Necessary Claims do not include any claims other than those set forth above even if contained in the same patent as Necessary Claims.

8.7 **"Scope"** means the protocols, electrical signaling characteristics, commands, and clocking signals solely to the extent disclosed with particularity in the OPMA Specification where the sole purpose of such disclosure is to enable products to interoperate, interconnect, or communicate as defined within the OPMA Specification. Notwithstanding the foregoing, the Scope shall not include (a) any enabling technologies that may be necessary to make or use any product or portion thereof that complies with the OPMA Specification, but are not themselves expressly set forth in the OPMA Specification (e.g., semiconductor manufacturing technology, X86 architecture, and processor microarchitecture); and (b) the implementation of other published specifications not developed by or for AMD but referred to in the body of the OPMA Specification.

Executed by the undersigned authorized representatives of AMD and Adopter, respectively, to be effective as of the Effective Date.

Adopter Company Name

Signature

Representative Name (Printed)

Title

Date

Advanced Micro Devices, Inc.

Brian Spross
AMD Senior Technology Counsel

Date

AMD Address for Legal Notice:
Advanced Micro Devices, Inc.
5204 E. Ben White Blvd., M/S 562
Austin, Texas 78741
ATTENTION: CPG Legal
Facsimile: 512-602-4932

Adopter Address for Legal Notice:
_____
_____
_____
_____
Facsimile: _____

AMD Technical Contact:
Dave Tobias
800-538-8450
E-Mail: dave.tobias@amd.com

Adopter Technical Contact:
Name: _____
Phone: _____
E-Mail: _____

Adopter: Please have an authorized representative sign on behalf of Adopter and fax the Agreement to AMD Legal, Attention: Sarah Blankenship, Fax No. 512-602-4932. AMD Legal will return a counter-signed Agreement to Adopter's Legal contact as listed above.

Open Platform Management Architecture Specification

Publication #	**32200** Revision: **1.3**
Issue Date:	**January 2008**

Advanced Micro Devices

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.

The contents of this document are provided in connection with Advanced Micro Devices, Inc. ("AMD") products. AMD makes no representations or warranties with respect to the accuracy or completeness of the contents of this publication and reserves the right to make changes to specifications and product descriptions at any time without notice. The information contained herein may be of a preliminary or advance nature and is subject to change without notice. No license, whether express, implied, arising by estoppel, or otherwise, to any intellectual property rights are granted by this publication. Except as set forth in AMD's Standard Terms and Conditions of Sale, AMD assumes no liability whatsoever, and disclaims any express or implied warranty, relating to its products including, but not limited to, the implied warranty of merchantability, fitness for a particular purpose, or infringement of any intellectual property right.

AMD's products are not designed, intended, authorized or warranted for use as components in systems intended for surgical implant into the body, or in other applications intended to support or sustain life, or in any other application in which the failure of AMD's product could create a situation where personal injury, death, or severe property or environmental damage may occur. AMD reserves the right to discontinue or make changes to its products at any time without notice.

Trademarks

AMD, the AMD Arrow logo, AMD Opteron, and combinations thereof are trademarks of Advanced Micro Devices, Inc.

Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement
entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Contents

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

List of Figures

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

List of Tables

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Revision History

Revision bars indicate changes from the last revision.

Date	Revision	Description
January 2008	1.3	Added Network Controller Sideband Interface (NC-SI) support to the specification. Introduced CPU interface I^2C/SMBus. Provided additional clarifications in other sections.
June 2007	1.2	Updated mechanical connector specification to reflect SO-DIMM DDR2 only.
May 2006	1.1	Clarifications to SO-DIMM connector keying and mechanical drawing, mCard power consumption, PCI reset signal, SEEPROM and I^2C/SMBus device addressing, motherboard I/O terminations, firmware debug header and fan types sections. Added Appendix D. *Technical Implications of Changes in OPMA Specification R 1.1*.
May 2005	1.02	Updates to fan mux select signal polarity (pages 26, 37, and 42).
February 2005	1.01	Removed Appendix D.
December 2004	1.00	Initial public release.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Chapter 1 Introduction

This chapter describes the Open Platform Management Architecture (OPMA) goals and background information.

1.1 Overview of OPMA Goals

OPMA:

- describes the motherboard and subsystem card aspects of a modular manageability architecture for servers and workstations and explains the benefits of this approach. Management subsystems feature cards that are OPMA compliant, which are known as *mCards*.
- describes three tiers of mCard functionality that drive three different price ranges for mCards. These feature tiers are provided as examples; the market decides what feature sets are actually offered in the OPMA-defined mCard form factor both initially and over time. The example tiers were chosen to show the range of manageability solutions that were considered in the definition of the OPMA hardware interface, required hardware resources, and firmware extensions.
- provides a publicly available specification for a manageability subsystem electrical interface, mechanical form factor, and firmware interfaces for creating OPMA-compliant motherboards and OPMA feature cards (mCards) for each of the described functionality tiers.

1.2 Background

In the past, platform hardware manageability in servers was treated as a premium, OEM-specific value-added feature. As the marketplace matures, hardware management is evolving into a standards-based, “must have” feature found on the vast majority of enterprise class servers. Today, the BMC (Baseboard Management Controller) firmware and IPMI (Intelligent Platform Management Interface) command interfaces to the host and remote systems are clearly defined and widely accepted.

Despite the advances made in standardizing manageability firmware and software interfaces, manageability hardware subsystems are still proprietary and are often soldered to the baseboard. Because of this practice, manageability hardware subsystems:

- are reinvented for many server platforms, often times even within a given server motherboard/system manufacturer’s product lines.
- may be inefficient to develop due to lack of re-use across designs, thus adding significantly to time to market, system development costs and risk.
- may not provide the proper flexibility of cost/benefit trade-offs for the server customer.
- may not provide a customer driven, cost-effective upgrade path for servers as manageability requirements change.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled “OPMA SPECIFICATION LICENSE AGREEMENT.”

- do not enable the building of an efficient, competition-based infrastructure ecosystem for hardware manageability.

To address these issues, AMD has developed the OPMA specification.

1.3 Basic Goals for OPMA

As server volumes rise, server prices fall. All areas of server design must be more closely scrutinized for efficiency that drives cost savings while retaining or increasing customer value, features, and flexibility. Industry standards bodies have made good strides in standardizing the external command interfaces to the management subsystem. OPMA extends the standardization concept to the management subsystem hardware interconnect and associated system architecture. The combination of a standardized external command interface with standardized, modular management subsystem architecture (OPMA) forms the basis of efficient, cost effective management subsystems. OPMA is primarily oriented at standardizing hardware interfaces. OPMA does not preclude the use of external command interfaces other than IPMI.

OPMA does not dictate a management subsystem feature set. Instead, it provides a robust hardware interface that allows management subsystems to be attached to server motherboards in a standardized way. The main goals of the OPMA specification are to:

- standardize the server management subsystem hardware interface architecture using a modular approach while continuing to allow intelligent innovation in management subsystems.
- reduce platform development risk, cost, and time to market.
- broaden motherboard applicability. Avoid missed system sales due to the management subsystem not meeting the customers' needs or cost targets.
- enable the evolution of manageability subsystem hardware into a COTS (commercial off the shelf) model that supports various tiers of capability/price.
- assist OEMs in moving to an outsourcing model for server design by enabling a multi-vendor approach for supplying the manageability subsystem. Give OEMs a build or buy decision that is not available today.
- increase customer satisfaction by providing more customer flexibility and choice at a reduced cost.
- enable cost reductions obtained by using COTS devices to drive enhanced manageability into markets that could not previously afford it.
- enable management subsystem infrastructure competition by providing a level playing field.
- free up the PCI slot commonly used for Keyboard, Video, and Mouse over Internet Protocol (KVMoIP) feature cards while increasing performance of the remote graphics console.
- achieve all of the above with minimal impact on BMC and BIOS firmware investments.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

1.4 Audience and Purpose of Document

This document is the single point of OPMA architecture definition. The audience is management, marketing, and engineering personnel of ISVs, IHVs, ODMs and OEMs who develop, test, and market server platforms and management subsystems.

1.5 Acronyms and Terminology

Table 1 lists the definitions of acronyms and terms used in this specification. In some of the definitions, the IPMI specification is referred to. For detailed information on the IPMI specification and other referenced specifications, refer to Table 3 on page 18.

Table 1. Acronyms and Terminology

Term	Definition
ACPI	Advanced Configuration and Power Interface. See ACPI specification for details.
BIOS	Basic Input Output System. This is the boot firmware on a standard PC system (including servers). It also provides some abstraction for system hardware.
BMC	Baseboard Management Controller. This is the main component of the OPMA subsystem that provides IPMI command processing, alerting, error logging, etc. in compliance with the IPMI specification.
BT	IPMI's Block Transfer command interface.
CLI	Command Line Interface. This is a text-based interface to a BMC.
CMOS	Complimentary-symmetry Metal Oxide Semiconductor. This term in this document refers to a battery-backed storage element found on legacy PC systems.
COTS	Commercial Off The Shelf. This refers to items that are commonly available commercially, not custom and not military.
DDC	Display Data Channel. Provides plug-and-play data to whatever device the monitor is plugged into.
DDWG	Digital Display Working Group—owns DVI specification.
DMTF	Distributed Management Task Force – www.dmtf.org
DVI	Digital Visual Interface. A set of buses for moving video data within a system. DVI-D digitally encodes the video data and then puts it out onto a bus. DVI-A is simply the same analog video signals seen on a standard VGA connector. DVI-I is the union of DVI-D and DVI-A signals to provide both methods of video information transmission over a single cable and/or connector.
FRU	Field Replaceable Unit. See the IPMI specification for details.
I^2C	Inter-Integrated Circuit bus. A simple two-wire bus often used to allow system-processing elements to read low cost sensor devices. An I^2C device interfaces to the system via the I^2C bus.
ICMB	Intelligent Chassis Management Bus. An IPMI-defined bus for connecting management processors that exist in separate physical chassis. See the IPMI specification for details.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Term	Definition
IHV	Independent Hardware Vendor. Used in this specification to refer to the manufacturer of mCard subsystem boards.
IPMI	Intelligent Platform Management Interface specification. See IPMI specification for details.
IPMB	Intelligent Platform Management Bus. An IPMI-defined bus for connecting management processors that exist in a single physical chassis. See IPMI specification for details.
ISV	Independent Software Vendor. Used in this specification to refer to the manufacturer of OPMA subsystem board firmware, OS-level drivers, and OS-level applications.
JTAG	Joint Test Action Group
KCS	IPMI's Keyboard Controller Style command interface
KVMoIP	Keyboard, Video, and Mouse over Internet Protocol. Used for implementing remote video consoles on headless (i.e., no local keyboard mouse or display) servers.
LCD	Liquid Crystal Display
LED	Light Emitting Diode
LPC	A reference to the low pin count bus specification. This is a common host system interface bus for BMCs.
LUN	IPMI Logical Unit Number
mCard	This is the generic name for an OPMA-compliant Management Card. It is a modular, connector based management subsystem which contains the BMC and associated hardware resources. In this document, "mCard" and "OPMA feature card" are used interchangeably.
NetFn Code	Net Function value parameter as defined in the IPMI specification document.
NIC	Network Interface Controller (a.k.a Ethernet controller). Typically includes an Ethernet Media Access Controller and may also include an integrated physical layer (PHY).
NC-SI	Network Controller Sideband Interface – A sideband NIC interface standard developed by the Distributed Management Task Force (DMTF).
ODM	Original Design Manufacturer. Used in this specification to refer to the developers of motherboards that are used in a completed server system.
OEM	Original Equipment Manufacturer. Used in this specification to refer to the server system manufacturer.
OOB	Out Of Band. A platform hardware management communications channel that connects a remote console directly to the BMC. This channel is working even when the managed platform is in a soft off state.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Term	Definition
OPMA	Open Platform Management Architecture. This acronym is used in several contexts as follows: OPMA architecture—This is the overall description of the way OPMA subsystem cards connect to and interact with the motherboard. It includes not only the OPMA connector and associated electrical specification, but required firmware interfaces as well as other resources located on the motherboard that are required to support the intended usage model. The OPMA architecture is described in this specification. OPMA subsystem—An implementation of the OPMA architecture on the motherboard taken together with an OPMA feature card. OPMA feature card—This is a feature board that contains the majority of the management subsystem hardware. The term "mCard" (management card) is often used for this in the interest of brevity. OPMA connector—This is the connector into which an OPMA feature card (mCard) is plugged to provide a given system hardware management feature set. OPMA interface—This refers collectively to all OPMA signals that are routed to and from the OPMA connector. The term "interface" as used in this document refers to an electrical interface as opposed to software or command interfaces.
OSPM	Operating System directed Power Management
PLD	Programmable Logic Device
POST	Power On Self Test. In this specification, POST refers to code run at boot time by the BIOS on a server.
PWM	Pulse Width Modulation
RxD	Receive Data
SCI	System Control Interrupt. This is a way to notify the ACPI subsystem that a device is requesting service.
SDR	Sensor Data Record. See the IPMI specification for details.
SDRR	Sensor Data Record Repository
SEEPROM	Serial Electrically Erasable Programmable Read-Only Memory
SEL	System Error Log. See the IPMI specification for details.
SERIRQ	Serialize Interrupt Request. See the LPC specification for details.
SMBus	An I^2C bus derivative.
SMI	System Management Interrupt.
SMIC	IPMI's Server Management Interface Chip command interface
SoL	Serial over LAN. The redirecting of text data to a remote text console using Ethernet as a transport. Redirected text is typically the BMC's CLI or system-generated text such as BIOS boot and setup screens.
SPD	Serial Presence Detect (used for DRAM auto sizing)
SSIF	IPMI's SMBus System InterFace command interface
TCO	Total Cost of Ownership

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Term	Definition
TxD	Transmit Data
UART	Universal Asynchronous Receiver / Transmitter
USB	Universal Serial Bus
Zero Impact	The ability to add an OPMA connector to an existing system for the purposes of upgrading the existing management subsystem without requiring changes in either the existing down solution firmware or in system BIOS.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

1.6 Conventions

Table 2 describes the conventions used in this specification.

Table 2. Conventions

Term	Definition
Down solution	This refers to a basic, BMC-based, IPMI 1.5-compliant management subsystem that is soldered to the motherboard.
OPMA	When OPMA is used as a noun, it should be taken by the reader to mean the OPMA specification.
Upgrade kit	This refers to an mCard which is used to add enhanced capabilities to a server that employs a down solution.
Hardware signal names	OPMA hardware signal names are presented in capital letters with underscores between words. If the name ends with "L" then the signal is low true. If the name does not end with "L", it is high true. Example: MCARD_CLR_CMOS_L
IPMI command names	The names of all IPMI commands are presented with no spaces between the words in the name. Only the first letter of each word is capitalized. Example: GetSystemTypeIdentifier
IPMI command parameters	The names of all IPMI command parameters are presented with spaces between the words in the name. Only the first letter of each word is capitalized unless the word is an acronym or abbreviation. Example: Interface ID
ms	Abbreviation for millisecond(s) (1/1,000th of a second)
µs	Abbreviation for microsecond(s) (1/1,000,000th of a second)
h	Suffix for a number expressed in hexadecimal. Every four digit set is separated by an underscore for readability. Example: 1234_5678h ***Note**: Decimal is the default radix.*
b	Prefix for a number expressed in binary. Every four digits set is separated by an underscore for readability. Example: 1010_1101b ***Note:** Decimal is the default radix.*
Bit fields [x:y]	Brackets designate bit fields. For example, [0] means bit zero. [3:0] refers to bits zero through three.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

1.7 Reference Documents

Table 3 lists the documents referenced in this OPMA specification or are otherwise related to the understanding of this specification.

Table 3. Reference Documents

Specification	Comment
IPMI 1.5	*Intelligent Platform Management Interface Specification*, Version 1.5, Revision 1.1
IPMI 2.0	*Intelligent Platform Management Interface Specification*, Version 2.0, Revision 1.0
DVI	*Digital Visual Interface Specification*, Version 1.0
I^2C	*Inter-Integrated Circuit Bus Specification*, Version 2.1
JTAG	IEEE 1149.1
LPC	*Low Pin Count Interface Specification*, Revision 1.1
USB	*Universal Serial Bus Specification*, Revision 2.0
NC-SI	*Network Controller Sideband Interface Specification Version 1.0.0a* (http://www.dmtf.org/standards/published_documents/DSP0222.pdf)

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Chapter 2 OPMA Functionality Tiers

For explanatory purposes, this specification defines and refers to three basic sets of functionality labeled M1, M2, and M3. Higher numerical designations imply more functionality and higher associated cost both on the platform and in supporting data center infrastructure. In addition, higher numerical designations are largely functional supersets of lower levels. These example tiers were selected to show the range of features that OPMA was designed to support and to provide terminology to express OPMA concepts. The capabilities of these exemplary functional tiers are presented in the following sections. *Again, these tiers are for explanatory purposes only. The market will decide what features are actually delivered on any given OPMA compliant management subsystem as long as said feature set is implemented within the confines of connector level interoperability as defined by this specification.*

2.1 M1—The Value Solution

Features of the M1 functional tier are:

- Minimal implementation cost, both for the mCard and reduced data center cabling infrastructure.
- No dedicated mCard manageability NIC
  - System NIC is shared through NC-SI for both server and manageability traffic

OPMA 1.2 and earlier dedicated an SMBus to the function of sideband communications with the system NIC for the purpose of implementing the Shared NIC feature of the M1 tier (AKA “IPMI pass through” and “NIC sideband”). In OPMA 1.3, the Shared NIC feature is implemented using the recently ratified Distributed Management Task Force (DMTF) Network Controller Sideband Interface (NC-SI). Prior versions of OPMA anticipated the advent of NC-SI by reserving pins for this purpose. These pins are now formally defined in OPMA 1.3 to support NC-SI implementations.

SMBus based NIC sideband was nonstandard and was not universally adopted. Since the new NC-SI functionality is far superior, the SMBus which was associated with NIC sideband in prior versions of OPMA have been repurposed to provide a CPU management SMBus. Thus, MCard implementations compliant with OPMA 1.3 and later will **not** provide an SMBus based NIC sideband feature. Refer to section 7.4 on page 62 for details on the backward compatibility scheme associated with this re-use.

Note: *For all references to NC-SI in this document, refer to the DMTF website for document number: DSP0222 (* http://www.dmtf.org/standards/published_documents/DSP0222.pdf *)*

Figure 1 on page 20 shows a system configuration using the M1 functionality tier.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled “OPMA SPECIFICATION LICENSE AGREEMENT.”

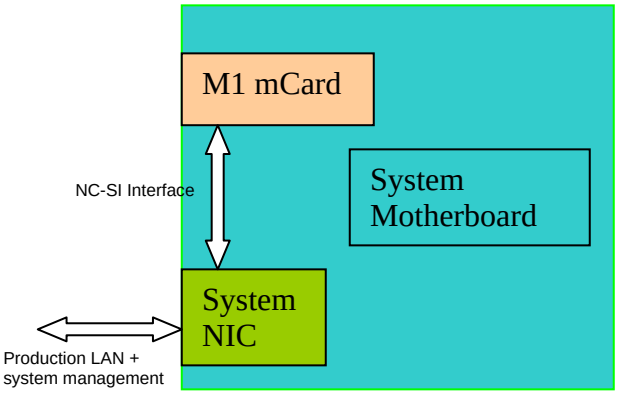

Figure 1. M1 mCard with NC-SI Shared NIC Configuration

2.2 M2—Reasonable Price, Security and Performance

Features of the M2 functional tier provide:

- All M1 base features plus a dedicated management NIC.
- High performance and optimal LAN security.
- Access to hardware health data through BMC-based web server interface.
- A more powerful BMC CPU core, larger address space, more FLASH and RAM.
- Remote virtual mass storage functionality.

Any mCards with more powerful processors and more FLASH/RAM will fare better in this space due to the performance advantages needed by more complex tasks such as serving web pages. Figure 2 shows a system configuration using the M2 functionality tier.

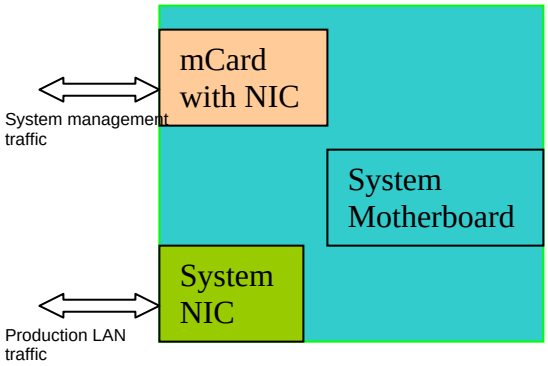

Figure 2. M2 mCard with Dedicated NIC

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

2.3 M3—High-End Solution with Graphics Console Redirection

Features of the M3 functional tier are M2 plus KVMoIP. M3 subsystems require video capture and compression hardware in order to provide useful frame rates for the end user. The M3 solution provides KVMoIP without consuming a PCI slot, giving it an advantage over some management solutions existing in today's market. Figure 3 shows system configuration using the M3 functionality tier.

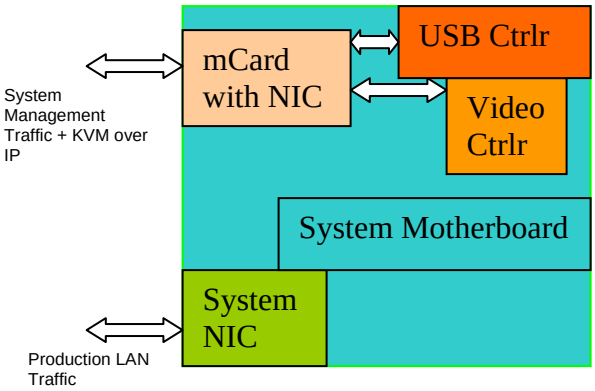

Figure 3. M3 with Dedicated NIC and KVMoIP Configuration

2.4 Mx—Upgrade Soldered-Down Management Subsystem

The motherboard designer may decide to solder a basic management solution down on the motherboard (a "down solution") while still needing a management system upgrade path. OPMA supports the following two scenarios associated with this usage model:

- As an upgrade path for cost sensitive server systems—The system can ship with a low cost solution (i.e., M1 feature set) soldered down to the motherboard, and then provide an upgrade path to M2 or M3 using an OPMA connector.
- As a transitional strategy for server product lines that will eventually use OPMA-only solutions—These products can leave their current management subsystems down and add the OPMA connector. When the OEM is comfortable with the maturity of mCard offerings, it can then depopulate the down components to achieve a cost savings. Future board spins can remove the down pads entirely.

The term upgrade is too broad to be left unbounded. As a result, OPMA places limits on its scope. The goal is to limit OPMA complexity and reduce the requirement for changes in non-OPMA-aware down solution firmware or in system BIOS in an upgrade scenario. An ability to provide an

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

upgrade path without requiring changes to existing BIOS and BMC firmware is driven by the usage models previously listed, and is referred to as the zero impact upgrade capability. Zero impact would allow OPMA-based manageability subsystem upgrades to be integrated into existing non-OPMA based product designs quickly and with low risk.

The main scope limitation for upgrades is that the down solution *must* use NC-SI as its OOB communications channel, and the upgrade kit *must* provide a dedicated management NIC as part of its feature set. In OPMA terminology, you can upgrade a down version of M1 to M2 or M3, but you can't upgrade a down solution which uses NC-SI to any mCard that also uses NC-SI. It is also impossible to upgrade a down solution which uses a dedicated manageability LAN connector. Imposing these limitations eliminates several cross-configuration compatibility issues.

When mCard is used as an upgrade kit, it leverages most of the down solution infrastructure and then provides additional capabilities. The down solution, having already been ported to the platform specifics is still used for sensor scanning. In-band IPMI accesses go to the down solution BMC, same as before the upgrade kit was installed. This means that the down solution maintains control of the IPMI-defined host interface. In order to reduce conflicts, the mCard must, upon power-up, determine whether it's being used in a stand-alone configuration or as an upgrade kit. It does this by detecting presence of another BMC over IPMB. If stand-alone, the mCard enables its host system interface. If used as an upgrade kit, the mCard leaves its IPMI host interface disabled to avoid contentions with the down solution. "IPMI host interface" refers to KCS, BT, SMIC, and SSIF interfaces are defined in the *IPMI Specification* versions 1.5 and 2.0. It does not refer to the LPC bus interface of the BMC, which should always remain enabled.

Upon detecting that it's being used as an upgrade kit, the mCard must change its internal IPMI device address. BMC devices normally use the address 20h. However, in an upgrade scenario, the down solution still uses 20h. Thus, the upgrade kit must use a different address such as 28h (or 48h if 28h is already used). No part of the down solution's operation changes as a result of adding the upgrade kit. The down solution still maintains its own in band and out of band channels as before the upgrade. The upgrade kit simply provides additional features such as virtual floppy disk, USB-based mass storage, KVMoIP, web-hosted interface to management data and control features, etc. The kit also adds a second out-of-band IPMI channel through the dedicated NIC. This channel is mainly used for transferring high bandwidth data associated with the upgrade kit (i.e., data associated with a mass storage device, KVMoIP traffic, etc.).

The upgrade kit can also return sensor readings through its out-of-band channel, but remote console software must obtain these readings using the upgrade kit's BMC device address (for example) 28h (this may also be another address such as 48h, etc.). Commands sent to 20h goes to the down solution.

A limitation of this arrangement is the firmware upgrade strategy for the upgrade kit. Since the mCard IPMI-defined host interface (KCS, BT, etc.) is disabled in the upgrade kit scenario, the mCard firmware must be upgraded remotely through the out-of-band LAN channel.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Chapter 3 OPMA Interface Signal Specification

This section provides details on the signals specified for the OPMA connector. These signals are collectively referred to as "the OPMA interface" in this document. For termination requirements, example motherboard required circuits etc., that are required to support the OPMA signals, refer to the appropriate sections of this document.

3.1 Background

Usage and cost models were both taken into account when defining the OPMA connector, OPMA signal list, mCard board mechanicals, and OPMA system resource requirements. Due consideration was given as to where certain resource devices should be located. The main partition is whether the device should go on the motherboard or on the mCard. OPMA places resources where they make the most sense to the end markets.

For example, the cost involved in of some of the standard components that are essential for all tiers of the mCard is borne by the motherboard. This design decision reduces the cost and size of the mCard. The size reduction provides more flexibility to the motherboard vendors for placement of the OPMA connector. However, certain resources are specific to more expensive manageability tiers. In those cases, costs specific to those manageability levels are borne by the mCard implementations of those levels.

3.2 Signal Callout Grouped by Functions

This section provides a description of the major buses and signal subgroups. Under each signal group, individual signals are referred to by a short descriptive name followed by the capitalized hardware signal name inside parentheses. For signals that are closely related, shortcuts have been taken when displaying the associated hardware signal names. For example, if there are four signals named SIG0, SIG1, SIG2, and SIG3, these are abbreviated as SIG0–SIG3. For full signal names, refer to Table 5 on page 37.

3.2.1 LED Control Signals

The LED control signals enable mCard-controlled LEDs to provide a visual indication of the hardware health and status. OPMA defines only two dedicated LED control signals at the connector. If the system level manageability solution requires a more detailed visual indication of faults with other LEDs or LCD indicators, they may be achieved through I^2C or SMBus-based indicators.

Any other system specific GPIOs should be implemented on the motherboard using I^2C or SMBus-based GPIO expanders (i.e., 8575 devices). OPMA does not require the implementation

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

of additional GPIOs. However, such GPIOs are likely to be implemented for a variety of purposes including GPIO based sensors and fault diagnostic LEDs. BMC firmware that is specific to a given motherboard must handle these system specific GPIOs.

Signal Group: LED Control

- Fault LED control (FAULT_LED_L)
  - Connect to a BMC GPIO for fault status indicator LED control.
  - Direct BMC connection allows control of this LED even under a hung I^2C or SMBus scenario.
- Chassis ID LED control (CHASSIS_ID_L)
  - Connect to a BMC GPIO for chassis ID LED control
  - Direct BMC connection simplifies blink rate control by the mCard firmware.

3.2.2 USB Interface Signals

The OPMA interface USB signals enable the mCard to emulate a USB keyboard and mouse, which are used for KVMoIP. This USB interface is also used by an mCard to emulate USB mass storage devices (i.e., virtual CD, virtual floppy disk, etc.) to the system.

Signal Group: USB

- USB interface (USB_P, USB_N)
  - One USB bus

Note: *The OPMA interface only supports one USB channel. If a USB hub device is required for a particular mCard implementation to emulate multiple USB channels, this device is expected to reside on the mCard itself.*

3.2.3 Push-Button Signals

The push-button signals are dedicated I/O control signals that are connected directly to the mCard. This allows the mCard to remotely/virtually control these signals. These signals are implemented as dedicated I/O control instead of I^2C or SMBus-based control signals because the supported functions are deemed mission critical. Refer to Section 3.2.14 on page 29 for descriptions of signals that monitor push-button signal status.

Signal Group: Buttons

- mCard host's system Power button control (MCARD_PWRBTN_L)
  - This output allows a remote console to control the mCard host's system Power button signal. This is used to power up or power down a server system remotely. It can also be used to force the server into a low power ACPI state or to wake it up from a low-power ACPI state.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

- mCard host's system Reset button control (MCARD_RSTBTN_L)
  - This output allows a remote console to control the mCard host's system reset button signal. This is used to recover a hung remote system.
- mCard NMI button control (MCARD_NMIBTN_L)
  - This output allows a remote console to control the mCard host's system NMI (core dump) signal. The core dump information can be used to debug system software issues.

3.2.4 Video Capture DVI-I Signals

The server's graphics chip sends the DVI-I (Digital Visual Interface) signals to the mCard, which uses these signals to capture the graphics console. The captured video data is compressed and transmitted over an Ethernet link to a remote console. The console operator can then view all graphical screens of the remote machine. This capability is part of the KVMoIP feature that is the primary differentiator of the M3 tier.

OPMA provides a total of 18 signals for this purpose. Signals in this group are defined by the Digital Display Working Group (DDWG) (*http://www.ddwg.org*). The DDWG specification describes a TMDS (Transition Minimized Differential Signaling) receiver chip for retrieving the serialized data from the TMDS channels. M3 mCard implementations require the receiver chip for recovering the video stream from the encoded TMDS signals, and M3 designers must design this chip into M3 mCards. M1 and M2 solutions are not burdened with the cost of this device.

Signal Group: Video Capture DVI-I

- TMDS differential data pairs (DVI_TXD0_H–DVI_TXD2_H, DVI_TXD0_L–DVI_TXD2_L)
  - These six signals form three communications channels that carry digitized video data to an M3 type mCard for graphics capture purposes. Channel 0 carries the blue data, channel 1 carries the green data, and channel 2 carries the red data.
- TMDS differential clock pair (DVI_TX_CLK_H, DVI_TX_CLK_L)
  - These two signals form a differential clock for digitized video data transfer.
- DVI interface DDC clock and data (DVI_DDC_DATA, DVI_DDC_CLK)
  - These two signals form the Display Data Channel for the DVI interface. An M3 mCard can communicate its capabilities to a host system's DVI-based video chip using this communications bus. For example, support of this feature can disallow a video subsystem from driving unsupported video resolutions, etc., to the M3 mCard video capture circuit.
- Analog video interface signals (ANALOG_R, G, B, H, V)
  - Five signals that provide analog video data (Red, Green, Blue, Horizontal Synch, and Vertical Synch) for capture by M3 cards using analog to digital conversion video capture methods.
- Analog interface DDC clock and data (DVI_DDC_DATA, DVI_DDC_CLK)
  - These two signals form the Display Data Channel for the analog interface. An M3 mCard can communicate its capabilities to a host system's VGA-based video chip using this

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

communications bus. For example, support of this feature can disallow a video subsystem from driving unsupported video resolutions, etc., to the M3 mCard graphics capture circuit. These signals have different electrical characteristics than the DVI DDC interface signals.

3.2.5 Multi-Bank Fan Control Signals

OPMA partitions the server system fans into three control banks. Each bank can be independently controlled to operate at three levels of speed through PWM control. Variable speed fans allow the mCard BMC to run the fans at the slowest (and therefore quietest) possible speed while maintaining associated subsystems at acceptable temperature levels.

Signal Group: Multi-Bank Fan Control

- System fan PWM control (MCARD_FAN_PWM_SYS)
  - PWM output to control the speed of all system fans
- CPU fan PWM control (MCARD_FAN_PWM_CPU)
  - PWM output to control the speed of all CPU fans
- Power supply fan PWM control (MCARD_FAN_PWM_PS)
  - PWM output to control the speed of all power supply fans

3.2.6 Multiplexed Fan Tach Input Signals

The fan tachometer inputs are provided for the mCard to read in the RPM of each set of fans connected to it. It's difficult to ascertain the optimal number of fan tach inputs to support over a wide set of server system implementations. In addition, interoperability of mCard tiers and their host server systems is of high importance. While some BMCs have enough native tach inputs to handle certain system scenarios, none have been found that handle all possible fan requirements. As a result, OPMA incorporates fan tach signal multiplexing. Platforms with more than four fans are required to use multiplexers on the motherboard and such multiplexers are controlled by the mCard. The mCards whose BMCs have less than four fan tach inputs must add an additional 4:2 multiplexer (if the BMC has two fan tach inputs), or 4:1 multiplexer (if the BMC has only one fan tach input) as required on the mCard. Refer to section 8.3 on page 63 for more details.

Signal Group: Multiplexed Fan Tachometers

- Multiplexed fan tach inputs (MCARD_FAN_TACH0–MCARD_FAN_TACH3)
  - Four tachometer inputs from motherboard-based fan tachometer mux.

3.2.7 Fan Tach Mux Bank Selector Signals

These signals select one of four fan tachometer banks to be switched in for read back by the mCard.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Signal Group: Multiplexed Fan Tachometer Controls

- Fan tach multiplexer control (MCARD_FAN_SEL0_L–MCARD_FAN_SEL3_L)
  - Four tachometer mux control active Low signals
  - Controls which tachometer bank is switched in from the motherboard-mounted digital multiplexer. Firmware may have to enable more than one of these signals simultaneously depending on the number of fans in the system..

3.2.8 Single Wire Analog Voltage Sensor Signals

The analog voltage sensors are primary voltage rails brought in for the mCard A/D converters or analog comparators. The mCard monitors these rails for appropriate voltage levels. While some of the existing BMCs used on mCards provide A/D converters, others provide analog comparators for voltage monitoring. The motherboard vendor must properly condition the voltage rails connected to these inputs. Depending on which voltage rail is being monitored in each of these analog channels, all the voltage rails signals must be divided down or amplified up appropriately. The ACOMP_ADC voltage has to have voltage levels that is no greater than +3.3V (AVDD) ± 2% and no less than +1.8V.

- ACOMP_ADC# Input Voltage Range = +1.8V to AVDD (max), positive
- AVDD = +3.3V ± 2%

Signal Group: Single Wire Analog Sensors

- Analog voltage sensors (ACOMP_ADC0–ACOMP_ADC5)
  - Six analog voltage sensor inputs

3.2.9 MCard Serial Port and ICMB

These signals are provided to support 16550 UART style interfaces. The signals through the connector are digital logic levels only. Any circuitry required for translation to specific levels (i.e., RS-232 and RS-485) is provided on the motherboard. Two channels of this type are defined.

Signal Group: mCard Serial Port and ICMB

- Channel 0 is a full UART that allow connection to either DTE (Data Terminal Equipment – connect to another computer) or to DCE (Data Communications Equipment – connects to a MODEM).
  - Dedicated for use by BMC UART-generated traffic that will be routed to a back panel RS-232 connector
  - Hardware flow control supported
- Channel 1 includes TxD and RxD only along with TxD data enable control signal
  - Dedicated for use by ICMB traffic

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

– XON and XOFF (software flow control) only

3.2.10 Serial Over LAN Feedback Path

Traffic generated by the system UART may need to be redirected over the LAN for consumption by a remote text console. This is referred to as serial over LAN (SoL). OPMA provides the signals in this section to capture the local system's host UART stream for formation into packets by the BMC and later transmission to the remote console. See Chapter 6 beginning on page 51 for more details on expected usage.

Signal Group: SoL Serial Port

- One transmit/receive channel pair—includes TxD and RxD only
  – Dedicated purpose is to feed back system UART data to mCard for SoL support.

3.2.11 Dedicated Management Ethernet Signals

These signals provide the interface for the mCard dedicated management LAN. These signals go across the OPMA connector and connect the mCard NIC to a management LAN RJ-45 connector.

Signal Group: Management Ethernet

- Ten signals that implement the dedicated management Ethernet.

***Note**: Support of 802.11af (Power over Ethernet) is possible. This topic may require further discussion with other potential OPMA adopters, and an update to this specification should be performed before implementing to maintain interoperability. AMD will not be responsible for updating this specification.*

3.2.12 BMC Host Interface ID Signals

These signals are provided for the system level software, BIOS etc., to identify the type of IPMI communications interface supported by the installed mCard solution. The *IPMI Specification* currently describes four possible host interfaces—KCS, BT, SMIC, and SSIF.

Signal Group: Interface Type ID

- mCard IPMI communications interface ID referred in this document as INTERFACE_ID by name and the actual signal names on the connector for these signals are referred as MCARD_ID0 – MCARD_ID2.
  – Three mCard interface ID signals
  – Speeds boot process by allowing the BIOS to quickly identify the IPMI interface type of the mCard for initial handshaking operations with BIOS during POST. See Section 8.15 on page 71 and Section 9.2 on page 73 for more details.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

3.2.13 MCard Presence Detection Signal

This signal allows the BIOS to determine if an mCard is present in the OPMA connector. The BIOS should not attempt to read the INTERFACE_ID signals until after it has detected the presence of a mCard. Refer to Section 8.15 and Section 9.2 for details.

Signal Group: Card Detect

- mCard presence detection (MCARD_DETECT_L)
  - Speeds boot process by allowing the BIOS to quickly detect the presence of an mCard in the OPMA connector.

3.2.14 System Status Signals

These signals provide specific system status to the mCard.

Signal Group: Status

- System Power OK indication (PWROK):
  - This signal indicates that system power sequencing was successful and that all voltage rails have stabilized.
- Critical system thermal trip point detection (MCARD_THERMTRIP_L):
  - Single consolidated ThermTrip signal from all of the CPUs. ThermTrip indicates that the processor has overheated to the point of potential hardware damage and has shut down as a result.
  - Connected directly to a BMC GPIO (not through the I^2C or SMBus-based GPIO expander).
  - I^2C or SMBus-based GPIO expanders may be used to allow mCard to detect which CPU had the ThermTrip and to disable a set of processors based on that condition.
- Chassis intrusion detection (MCARD_INTRUDER_L):
  - Single consolidated intrusion sensor from multiple bays or locations within the system.
- Line AC Power indicator (SYS_LINE_AC_L):
  - Provides an indication of standby power status to the mCard. An active signal indicates that the standby power is good. A rising edge on this signal indicates that system AC power has been cut off. The system power supply typically continues to supply standby power for a small amount of time (system specific) after AC power is removed. The BMC is operational during this small window and should use the assertion of this signal to do any emergency clean up that may be required. For example, if the BMC is in the middle of writing to NV storage, it should abort this process in an orderly manner so that data corruption does not occur.
- System Power button (SYS_PWRBTN_L):
  - The mCard monitors this input to detect when the system power button is pushed. This signal can be used to remotely detect a local Power button assertion.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

- System Reset button (SYS_RSTBTN_L):
  - The mCard monitors this input to detect when the system reset button is pushed. This signal can be used to remotely detect a local reset button assertion.
- System NMI button (SYS_NMIBTN_L):
  - The mCard monitors this input to detect when the system NMI button is pushed. This signal can be used to remotely detect a local NMI button assertion.
- System SMI (SYS_SMI_L):
  - The mCard monitors this input for detecting SMI activity on the motherboard. Usage of this signal is platform specific.

    ***Note**: The mCards do not generate SMI to the system using this signal; it is for monitoring purposes only.*
- System PCI reset (SYS_PCI_RST_L):
  - The mCard monitors this input to detect a PCI reset event. One potential use is to detect system resets that are initiated using Ctrl-Alt-Del. If the system does not contain a PCI bus, this signal should be used to monitor the reset status of the implemented system bus.
- System ACPI state indication (SYS_ACPI_STATE0–SYS_ACPI_STATE2)
  - Three signals
  - Binary encoded ACPI state indicator for the mCard. Motherboard hardware encodes the system ACPI state and presents it to the mCard through these inputs. See Section 8.8 on page 68 for usage information on these signals.
- SMBus I/O Expander Interrupt (SYS_SMBUS_IO_EXP_INTR_L):
  - The primary purpose of this signal is to provide the ability to interrupt the BMC based on signal inputs into I^2C or SMBus-based GPIO expanders. SMBus-based I/O expander devices (i.e., PCA 9555) may also be available that generate interrupts on a state change on any of its inputs.

3.2.15 System Control Signals

These signals are used by the mCard to control some specific, system-level parameters.

Signal Group: Control

- Local access lock out (MCARD_LOCAL_LOCK_L):
  - Assertion of this signal by the mCard enables a local mode access lock out mechanism.
- Clear CMOS control (MCARD_CLR_CMOS_L):
  - The mCard uses this signal to clear the host system CMOS RAM. Clearing the CMOS is often necessary after actions such as updating BIOS. Asserting this signal for at least 250 ms and then deasserting it causes the system CMOS to be cleared.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

- Speaker data (MCARD_SYS_SPKR_L):
  - The BMC may pulse this signal to send tones to the system speaker/transducer that is typically soldered to the motherboard. This allows the BMC a simple audile alarm capability. Special motherboard support is required to integrate the BMC speaker control with that normally provided by the system chipset.
- System Control Interrupt (MCARD_SCI_INT_L):
  - The mCard uses this signal to assert an SCI to the system to indicate to the system that the mCard has identified an event which requires attention from the host system. The system will send some system specific IPMI commands in response to receipt of this signal.

3.2.16 I^2C or SMBus Signals

OPMA supports a total of five I^2C or SMBus buses allowing an mCard to communicate with multiple I^2C and SMBus devices in the system. While the number of physical I^2C or SMBus interfaces on specific BMCs vary, it is the responsibility the mCard designer to ensure that all of the I^2C or SMBus pins on the OPMA connector are in some way connected to the BMC. The options for BMCs that support less than five discrete buses natively are to provide an I^2C or SMBus mux on the mCard to generate the additional buses, or to wire some of the buses together on the mCard itself (not recommended).

The OPMA specification reserves one I^2C or SMBus address for the I^2C or SMBus mux device to support BMCs that do not have five native I^2C or SMBus buses. See Section 3.3.2 on page 35 for more information.

Signal Group: I^2C or SMBus

- mCard IPMB (MCARD_I2C_IPMB_SCL, MCARD_I2C_IPMB_SDA)
  - One bus (two signals) for Intelligent Platform Management Bus
- Host CPU I2C bus or SMBus (MCARD_I2C_CPU_SCL, MCARD_I2C_CPU_SDA, MCARD_I2C_CPU_ALERT_L)
  - One bus (three signals) for communicating with host system CPUs.
  - SMBus Clock and Data from the OPMA mCard to host system CPUs.
  - MCARD_I2C_CPU_ALERT_L signal is the indication from the CPU(s) that service by the mCard is required. In multiprocessor systems, alerts from all CPUs must be combined on the motherboard before being routed to the mCard.
  - Refer to the host system's CPU motherboard design guide for electrical and termination requirements when interfacing to these signals.
  - mCard (BMC) hardware connecting to this bus must be SMBus 2.0 compliant.
- mCard shared host I^2C bus or SMBus (MCARD_I2C_SHARED_SCL, MCARD_I2C_SHARED_SDA)
  - One bus (two signals) for communicating with devices that are on an I^2C bus or SMBus that another master is also connected to. The typical example is sharing an I^2C bus with

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

the host chipset's SMBus controller so that both the system and the BMC have direct access to SMBus-based devices such as DRAM SPD (Serial Presence Detect).

  - If used in this configuration, the SMBus master devices (the host and the mCard) must conform to the multi-mastering protocol.

- mCard private I^2C or SMBus buses (MCARD_I2C_PRIVATE0_SCL, MCARD_I2C_PRIVATE1_SCL, MCARD_I2C_PRIVATE0_SDA, and MCARD_I2C_PRIVATE1_SDA)
  - Two buses (four signals) for mCard private I^2C or SMBus buses. The bulk of the I^2C or SMBus sensor devices that are monitored by the BMC should be placed on these buses.

3.2.17 LPC Bus Signals

The LPC bus is the main system CPU interface to the mCard. BMCs accept IPMI commands from the host system either through the LPC bus (KCS, BT, SMIC) or through the I^2C or SMBus (SSIF).

Signal Group: LPC

- All LPC signals as defined in the LPC specification
- Includes SERIRQ and serial SMI

3.2.18 Miscellaneous Signals

No relationship is implied between miscellaneous signals in this group.

Signal Group: Misc.

- 32.768-KHz clock signal (CLK_32768)
  - 32.768-KHz clock signal driven from the motherboard. Improves cost reduction for M1 solutions as well as other uses. All host systems should have this signal readily available regardless of ACPI state for driving the RTC circuit.

3.2.19 Firmware Debugger Probe Signals

These signals allow the connection of a debug probe for mCard firmware development and debugging. Refer to Section 8.10 on page 69 for more information.

Signal Group: Debug Interface

- Debug interface data and control (DEBUG_IF0–DEBUG_IF7):
  - Eight signals for debugging BMC firmware
  - The meanings (including electrical voltage levels/characteristics) of these signals are determined by the debug interface (for example, but not limited to JTAG) of the mCard's BMC. *This is not strictly a JTAG interface.* Motherboard designers should not make any assumptions about the voltages that are externally applied to these pins by a debug probe. ODMs/OEMs should simply route these signals from the OPMA connector to a

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

conveniently located connector on the motherboard, where they can be physically accessed by firmware developers even after the system is assembled. See Section 8.10 on page 69 for more information on the connector and its relation to the OPMA connector.

Signal Group: Debug Power

- Debug interface power (DEBUG_PWR0, DEBUG_PWR1)
  - Two signals
  - Used to support debug probes that provide power to the debug target through the debug interface. The electrical characteristics of these power pins are determined by the debug probe for the BMC, which is on the mCard board installed in the server.

3.2.20 Network Controller Interface (NC-SI) Sideband Signals

These signals are used in a "Shared NIC" environment, where the OPMA mCard OOB interface connects to a sideband port on the host platform's primary NIC. The NIC filters network traffic intended for the management subsystem and routes it through these sideband signals. Cross references between the MCARD_RMII related signals defined herein and the NC-SI specification are listed in Table 4 on page 34. For electrical details, message protocols and functionality of these signals, refer to the DMTF's NC-SI specification. In Table 4, the "Signal Direction from mCard Point of View" column indicates signal direction from the OPMA mCard perspective. For example, MCARD_RMII_TXD_0 is designated as "output" meaning that this signal is an output from the OPMA mCard. Refer to the NC-SI specification for direction of these signals on the NIC side.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Table 4. Network Controller Sideband Interface (NC-SI) Signals

Signal Name	Signal Direction from mCard Point of View	OPMA Connector Pin Number	Corresponding NC-SI Specification Reference Signal Names	Description
MCARD_RMII_TXD_0	Output	187	TXD[0]	Transmit data 0
MCARD_RMII_TXD_1	Output	189	TXD[1]	Transmit data 1
MCARD_RMII_RXD_0	Input	193	RXD[0]	Receive data 0
MCARD_RMII_RXD_1	Input	195	RXD[1]	Receive data 1
MCARD_RMII_REF_CLK	Input	160	REF_CLK	Reference clock
MCARD_RMII_CRS_DV	Input	162	CRS_DV	Data Valid
MCARD_RMII_TX_EN	Output	164	TX_EN	Transmit enable
MCARD_RMII_RX_ER	Input	166	RX_ER	Receive error

3.3 OPMA Hardware Resources

The following sections describe OPMA reserved hardware resources. The addressing scheme for devices described here is an 8-bit addressing scheme.

3.3.1 SEEPROM Address Reservation

Unless restricted, mCards could locate their SEL (System Event Log) and SDRR (Sensor Data Record Repository) SEEPROMs on any OPMA I^2C or SMBus, and at any of the eight possible SEEPROM device addresses (0–7). In order to simplify BMC firmware, OPMA reserves two SEEPROM I^2C or SMBus addresses for the SEL and the SDRR. *If such devices are to be used,* mCard designers must locate SDR and SEL SEEPROM devices at SEEPROM I^2C or SMBus addresses A0h and A2h. In addition, these devices must be located on the OPMA-defined MCARD_I2C_PRIVATE0 only.

The mCards using this scheme must select SEEPROM devices that can be addressed at A0h and A2h locations. The A0h address is reserved for the SEL device and the A2h address is reserved for the SDRR device. If a single SEEPROM device is selected for both the SEL and SDRR, it can be located at address A0h. In such cases, address A2h is still reserved and not utilized.

OPMA-compliant motherboards may locate SEEPROM devices that are outside of the OPMA specification at any I^2C or SMBus addresses on the MCARD_I2C_PRIVATE0 besides the OPMA-reserved addresses of A0h and A2h. There are no SEEPROM device restrictions for any of the four other OPMA-defined I^2C or SMBus buses.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

3.3.2 I²C or SMBus Multiplexer Address Reservation

OPMA specifies that the mCard must drive five I²C or SMBus buses across the OPMA interface. However, not all BMCs have five native I²C or SMBus buses coming from them. There are two ways to handle fewer than five I²C or SMBus buses. Either the mCard designers can route a single I²C or SMBus interface from its BMC to multiple OPMA I²C or SMBus buses on the OPMA interface, or they can design in an I²C or SMBus mux so that one I²C or SMBus breaks out into multiple I²C or SMBus sub-buses.

Simply routing the signals without a mux (per the first solution given above) is cheaper, but will probably not be compatible with the maximum number of motherboards due to I²C or SMBus device address contention and I²C or SMBus device load partitioning on the motherboard. This solution is not recommended.

The recommended solution is to design in the I²C or SMBus mux if the BMC natively supports less than five I²C or SMBus buses. However, the motherboard designer can also add I²C or SMBus muxes to the motherboard design for purposes outside of OPMA requirements. Since the motherboard and the mCard designers are working independently, it's possible that they might both assign the same I²C or SMBus address to a mux device on a given bus, resulting in I²C or SMBus address contention. As a result, OPMA requires that, if the mCard uses an I²C or SMBus mux, it must reside at mux I²C or SMBus address E0h. To utilize this address the mCard must use a mux device whose base address starts at E0h. The motherboard designer is prohibited from configuring any motherboard-based I²C or SMBus mux device to I²C or SMBus address E0h on *any* OPMA-accessible I²C or SMBus.

The OPMA specification highly recommends using a SMBus 2.0 compliant, dedicated I²C/SMBus for the CPU host interface I²C/SMBus, i.e., MCARD_I2C_CPU bus.

3.3.3 Reserved Interface Signals

Most of the signals that are marked "reserved" in the OPMA specification have already been defined to support emerging technologies that are currently not made public.

Do not use any signals marked "reserved" for any purpose. Such signals should be left as No Connect (NC).

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

3.4 OPMA Feature Card Power Requirements

The mCard subsystem is designed to operate in all ACPI states of a server platform. For this reason dual voltage planes are supplied through the mCard connector. The following data provides power requirements for the mCard subsystem:

- VDD_3.3_DUAL is a +3.3VDC voltage rail that is supplied in ACPI power states S0, S1, S2, S3, S4 and S5
- VDD_5_DUAL is a +5VDC voltage rail that is supplied in ACPI power states S0, S1, S2, S3, S4 and S5
- In ACPI power states S0 and S1 the platform must be capable of supplying to the mCard subsystem:
  - 3.3 volts, 1.5 amps
  - 5 volts, 500 milliamps
- In ACPI power states S2, S3, S4 and S5 the platform must be capable of supplying to the mCard subsystem:
  - 3.3 volts, 1 amps
  - 5 volts, 100 milliamps

3.5 OPMA Feature Card Signal Tolerance Requirements

All mCard input signal buffers must be 5-V tolerant. All mCard output signal drivers must be at least 3.3-V capable.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

3.6 OPMA Signals Grouped by Function

Table 5 lists the signals that comprise the OPMA interface.

Table 5. OPMA Signals

Signal Group	Signal Name	Signal Function	I/O	Signal Termination on MB *
LED Control	FAULT_LED_L	Fault LED control	O	4.7k PU to +3.3V
	CHASSIS_ID_L	Chassis identification LED control	O	4.7k PU to +3.3V
USB	USB_P	USB Differential Pair	I/O	N/A
	USB_N			
Buttons	MCARD_PWRBTN_L	MCard remote system power button	O	4.7k PU to +3.3V
	MCARD_RSTBTN_L	MCard remote system reset button	O	4.7k PU to +3.3V
	MCARD_NMIBTN_L	MCard remote system NMI button	O	4.7k PU to +3.3V
Video Capture DVI-I	DVI_TX0_H	TMDS differential pair; channel 0	I/O	N/A
	DVI_TX0_L			
	DVI_TX1_H	TMDS differential pair; channel 1	I/O	N/A
	DVI_TX1_L			
	DVI_TX2_H	TMDS differential pair; channel 2	I/O	N/A
	DVI_TX2_L			
	DVI_TX_CLK_H	TMDS differential pair; clock	I/O	N/A
	DVI_TX_CLK_L			
	DVI_DDC_DATA	Data signal for DDC interface	I/O	4.7k PU to +3.3V
	DVI_DDC_CLK	Clock signal for DDC interface	I/O	4.7k PU to +3.3V
	ANALOG_DDC_DATA	Data signal for Analog VGA interface	I/O	4.7k PU to +3.3V
	ANALOG_DDC_CLK	Clock signal for Analog VGA interface	I/O	4.7k PU to +3.3V
	ANALOG_RED	Red for Analog VGA interface	I	N/A
	ANALOG_GREEN	Green for Analog VGA interface	I	N/A
	ANALOG_BLUE	Blue for Analog VGA interface	I	N/A
	ANALOG_HSYNC	Horizontal sync for Analog VGA interface	I/O	N/A
	ANALOG_VSYNC	Vertical sync for Analog VGA interface	I/O	N/A
Multi-Bank Fan Control	MCARD_FAN_PWM_CPU	mCard PWM CPU fan control	O	3.3V signal level
	MCARD_FAN_PWM_SYS	mCard PWM system fan control	O	3.3V signal level
	MCARD_FAN_PWM_PS	mCard PWM power supply fan control	O	3.3V signal level
Multiplexed Fan Tachometers	MCARD_FAN_TACH0	Multiplexed fan tach input 0	I	4.7k PU to +3.3V
	MCARD_FAN_TACH1	Multiplexed fan tach input 1	I	4.7k PU to +3.3V
	MCARD_FAN_TACH2	Multiplexed fan tach input 2	I	4.7k PU to +3.3V
	MCARD_FAN_TACH3	Multiplexed fan tach input 3	I	4.7k PU to +3.3V
Fan Tachometer Mux Controls	MCARD_FAN_ SEL0_L	Multiplexed fan tach mux control 0	O	4.7k PU to +3.3V
	MCARD_FAN_SEL1_L	Multiplexed fan tach mux control 1	O	4.7k PU to +3.3V
	MCARD_FAN_ SEL2_L	Multiplexed fan tach mux control 2	O	4.7k PU to +3.3V
	MCARD_FAN_ SEL3_L	Multiplexed fan tach mux control 3	O	4.7k PU to +3.3V

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Signal Group	Signal Name	Signal Function	I/O	Signal Termination on MB *
Single Wire Analog Sensors	ACOMP_ADC0	Analog voltage input 0	I	N/A
	ACOMP_ADC1	Analog voltage input 1	I	N/A
	ACOMP_ADC2	Analog voltage input 2	I	N/A
	ACOMP_ADC3	Analog voltage input 3	I	N/A
	ACOMP_ADC4	Analog voltage input 4	I	N/A
	ACOMP_ADC5	Analog voltage input 5	I	N/A
MCARD Serial Port	MCARD_DTR0_L	Data Terminal Ready for Serial Port 0	O	N/A
	MCARD_DCD0_L	Data Carrier Detect for Serial Port 0	I	N/A
	MCARD_DSR0_L	Data Set Ready for Serial Port 0	I	N/A
	MCARD_RI0_L	Ring Indicator for Serial Port 0	I	N/A
	MCARD_RXD0	Receive Data for Serial Port 0	I	N/A
	MCARD_TXD0	Transmit Data for Serial Port 0	O	N/A
	MCARD_CTS0_L	Clear to Send for Serial Port 0	I	N/A
	MCARD_RTS0_L	Request to Send for Serial Port 0	O	N/A
ICMB	MCARD_RXD1	Receive Data for ICMB	I	N/A
	MCARD_TXD1	Transmit Data for ICMB	O	N/A
	MCARD_TXD1_EN	Transmit Data Enable for ICMB	O	N/A
Serial Port for Serial Over LAN	MCARD_RXD_SOL	Receive Data for Serial Over LAN	I	N/A
	MCARD_TXD_SOL	Transmit Data for serial over LAN	O	N/A
	MCARD_AUX_SOL_CTRL_L	Serial Port handshake signal mux control	O	4.7k PU to +3.3V
Management Ethernet	MCARD_TX_H	Transmit Differential Pair	O	N/A
	MCARD_TX_L			
	MCARD_RX_H	Receive Differential Pair	I	N/A
	MCARD_RX_L			
	POE_PWR	Power over Ethernet positive DC supply	I	N/A
	POE_PWR			
	POE_GND	Power over Ethernet negative return	O	N/A
	POE_GND			
	LAN_BUSY_LED_L	Busy LED	O	N/A
	LAN_LINK_LED_L	Link LED	O	N/A
Interface Type ID	MCARD_ID0	Binary encoded interface type identification number	O	4.7k PU to +3.3V
	MCARD_ID1			
	MCARD_ID2			
Card Detect	MCARD_DETECT_L	Indicates presence/absence of mCard	O	4.7k PU to +3.3V
Status	ALL_PWROK	Indicates all the voltage rails are good	I	4.7k PU to +3.3V
	SYS_INTRUDER_L	Indicates intrusion on the system	I	10k PU to +3.3V
	SYS_THERMTRIP_L	System thermal trip alert signal	I	4.7k PU to +3.3V
	SYS_LINE_AC_L	Line AC present signal	I	4.7k PU to +3.3V
	SYS_PWRBTN_L	Indicates status of system power button	I	3.3V thru switch
	SYS_RSTBTN_L	Indicates status of system reset button	I	3.3V thru switch
	SYS_NMIBTN_L	Indicates status of system NMI button	I	3.3V thru switch

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Signal Group	Signal Name	Signal Function	I/O	Signal Termination on MB *
Status (cont.)	SYS_SMI_L	Motherboard SMI status.	I	4.7k PU to +3.3V
	SYS_PCI_RST_L	Motherboard reset status	I	4.7K PU to +3.3V
	SYS_ACPI_STATE0	Binary encoded indication of the current system ACPI sleep state (S0, S1, S2, S3, S4/S5)	I	4.7k PU to +3.3V
	SYS_ACPI_STATE1		I	4.7k PU to +3.3V
	SYS_ACPI_STATE2		I	4.7k PU to +3.3V
	SYS_SMBUS_IO_EXP_INTR_L	Interrupt from SMBus-based I/O expander	I	4.7k PU to +3.3V
Control	MCARD_CLR_CMOS_L	mCard system CMOS clear control	O	4.7k PU to +3.3V
	MCARD_LOCAL_LOCK_L	mCard local lock out control	O	4.7k PU to +3.3V
	MCARD_SYS_SPKR_DATA	mCard system speaker control	O	N/A
	MCARD_SCI_INT_L	mCard SCI to the system chipset	O	4.7k PU to +3.3V
I^2C or SMBus X 5	MCARD_I2C_IPMB_SCL	I^2C clock for the IPMB bus	I/O	4.7k PU to +3.3V
	MCARD_I2C_IPMB_SDA	I^2C data for the IPMB bus	I/O	4.7k PU to +3.3V
	MCARD_I2C_CPU_SCL**	I^2C clock for the host CPU interface bus	I/O	**
	MCARD_I2C_CPU_SDA**	I^2C data for the host CPU interface bus	I/O	**
	MCARD_I2C_CPU_ALERT_L*	Alert signal from the host CPU I^2C bus	I/O	**
	MCARD_I2C_PRIVATE0_SCL	I^2C clock for mCard private 0 I^2C bus	I/O	4.7k PU to +3.3V
	MCARD_I2C_PRIVATE0_SDA	I^2C data for mCard private 0 I^2C bus	I/O	4.7k PU to +3.3V
	MCARD_I2C_PRIVATE1_SCL	I^2C clock for mCard private 1 I^2C bus	I/O	4.7k PU to +3.3V
	MCARD_I2C_PRIVATE1_SDA	I^2C data for mCard private 1 I^2C bus	I/O	4.7k PU to +3.3V
	MCARD_I2C_SHARED_SCL	mCard shared host device I^2C clock	I/O	4.7k PU to +3.3V
	MCARD_I2C_SHARED_SDA	mCard shared host device I^2C data	I/O	4.7k PU to +3.3V
LPC	LAD0	LPC address-data bus	I/O	10k PU to +3.3V
	LAD1			
	LAD2			
	LAD3			
	LFRAME_L	LPC frame signal	I	N/A
	LRST_L	LPC reset for mCard LPC interface	I	N/A
	LDRQ_L	LPC DMA request signal	O	10k PU to +3.3V
	SERIRQ	Serialized IRQ signal	I/O	N/A
	LCLKRUN_L	Clock run (same as PCI CLKRUN#)	O	10k PU to +3.3V
	LCLK	LPC clock signal	I	N/A
Misc	CLK_32768	32.768-KHz clock signal	I	N/A
Debug Interface	DEBUG_IF0	Firmware debugger probe signals	I/O	N/A
	DEBUG_IF1			
	DEBUG_IF2			
	DEBUG_IF3			
	DEBUG_IF4			
	DEBUG_IF5			
	DEBUG_IF6			
	DEBUG_IF7			

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Signal Group	Signal Name	Signal Function	I/O	Signal Termination on MB *
Debug Power	DEBUG_PWR0	Firmware debugger probe power	I	N/A
	DEBUG_PWR1			
NC-SI	MCARD_RMII_TXD_0	Transmit data 0	O	***
	MCARD_RMII_TXD_1	Transmit data 1	O	***
	MCARD_RMII_RXD_0	Receive data 0	I	***
	MCARD_RMII_RXD_1	Receive data 1	I	***
	MCARD_RMII_REF_CLK	Reference clock	I	***
	MCARD_RMII_CRS_DV	Carrier Sense / Receive data valid	I	***
	MCARD_RMII_TX_EN	Transmit enable	O	***
	MCARD_RMII_RX_ER	Receive error	I	***
mCard Power and Ground	VDD_3.3_DUAL	3.3V supplied in all sleep modes	I	N/A
	VDD_3.3_DUAL			
	VDD_3.3_DUAL			
	VDD_3.3_DUAL			
	VDD_5_DUAL	5V supplied in all sleep modes	I	
	VDD_5_DUAL			

Notes:

** Signal-level terminations on the motherboard. PU – Refers to pull-up. The pull-up resistor values indicated in the signal termination column are suggested values. Motherboard designers must take their circuit and the type of I/O on the motherboard (e.g., push-pull, open-drain, etc.) into consideration while designing the values for the pull-ups.*

*** For termination requirements, refer to the host system processor motherboard design guide.*

**** For termination requirements and electrical characteristics of these signals, refer to the DMTF NC-SI specification.*

Compatibility Note: For consistency going forward, this version of the specification terminates all pull-ups to the +3.3VDC rail whereas earlier versions of the specification terminated some of these signals to the +5VDC rail. This specification is backward-compatible with motherboards that are compliant to the earlier termination specification.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Chapter 4 OPMA Connector Specification and Pin Assignments

OPMA leverages the 200-pin SO-DIMM connector commonly used by DDR2 infrastructure for mobile computers. This connector is reasonably small and contains 200 pins which support currently defined and projected expansion signals. These connectors are available in both vertical and right angle versions and will be readily available for the next two to three years at relatively low cost. The number of OPMA connector insertions/removal cycles in the field is expected to be low and on par with the expected number of insertions/removal cycles performed for DRAM sticks. Thus, no connector reliability or longevity issues are expected when leveraging this DRAM infrastructure for OPMA.

For purposes of brevity, technical documentation for the connectors recommended should be used as extensions to the OPMA specification for the purpose of connector definition. If connector substitutions are made, the substitute device specifications must conform to specifications referenced in those documents. For example, the right angle OPMA connector incorporates a standoff height that is greater than 7 mm. This height is needed to avoid the use of keep-out zones on the motherboard while allowing the mCard developer to populate both sides of the board with components. Table 6 lists the pin assignments for the OPMA connector.

Table 6. Pin Assignments for the OPMA Connector

	GND	1	2	RSVD	RSVD	
RSVD	RSVD	3	4	GND		
	GND	5	6	RSVD	RSVD	
RSVD	RSVD	7	8	GND		
	GND	9	10	DVI_TX0_H	Video Capture DVI-I	
LPC	LDRQ_L	11	12	DVI_TX0_L		
	LFRAME_L	13	14	GND		
	LAD0	15	16	DVI_TX1_H		
	LCLKRUN_L	17	18	DVI_TX1_L		
	LAD1	19	20	GND		
	LAD2	21	22	DVI_TX2_H		
	LAD3	23	24	DVI_TX2_L		
	SERIRQ	25	26	GND		
	LRST_L	27	28	DVI_TX_CLK_H		
	LCLK	29	30	DVI_TX_CLK_L		
	GND	31	32	GND		
RSVD	RSVD	33	34	DVI_DDC_DATA		
	RSVD	35	36	DVI_DDC_CLK		
	GND	37	38	RSVD		
I^2C or SMBus	MCARD_I2C_PRIVATE0_SCL	39	40	ANALOG_DDC_DATA		
	MCARD_I2C_PRIVATE0_SDA	41	42	ANALOG_DDC_CLK		
	GND	43	44	GND		
	MCARD_I2C_IPMB_SCL	45	46	ANALOG_RED		
	MCARD_I2C_IPMB_SDA	47	48	ANALOG_GREEN		
	GND	49	50	ANALOG_BLUE		
	MCARD_I2C_CPU_SCL	51	52	ANALOG_HSYNC		
	MCARD_I2C_CPU_SDA	53	54	ANALOG_VSYNC		

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

	MCARD_I2C_CPU_ALERT_L	55	56	GND	
	GND	57	58	RSVD	RSVD
	MCARD_I2C_PRIVATE1_SCL	59	60	RSVD	
	MCARD_I2C_PRIVATE1_SDA	61	62	GND	
	GND	63	64	MCARD_FAN_PWM_CPU	Multi-Bank Fan Control
	MCARD_I2C_SHARED_SCL	65	66	MCARD_FAN_PWM_SYS	
	MCARD_I2C_SHARED_SDA	67	68	MCARD_FAN_PWM_PS	
	GND	69	70	GND	
DEBUG	DEBUG_IF0	71	72	MCARD_FAN_TACH0	Multiplexed Fan Tachometers
	DEBUG_IF1	73	74	MCARD_FAN_TACH1	
	DEBUG_IF2	75	76	MCARD_FAN_TACH2	
	DEBUG_IF3	77	78	MCARD_FAN_TACH3	
	GND	79	80	GND	
MCard Serial Port	MCARD_DTR0_L	81	82	MCARD_FAN_ SEL0_L	Multiplexed Fan Tachometer Controls
	MCARD_DCD0_L	83	84	MCARD_FAN_SEL1_L	
	MCARD_DSR0_L	85	86	MCARD_FAN_ SEL2_L	
	MCARD_RI0_L	87	88	MCARD_FAN_ SEL3_L	
	MCARD_RXD0	89	90	GND	
	MCARD_TXD0	91	92	MCARD_RXD_SOL	TX/RX for SOL Support
	MCARD_CTS0_L	93	94	MCARD_TXD_SOL	
	MCARD_RTS0_L	95	96	GND	
	GND	97	98	FAULT_LED_L	LED Control
DEBUG	DEBUG_PWR0	99	100	CHASSIS_ID_L	
	DEBUG_PWR1	101	102	GND	
	GND	103	104	USB_P	USB
Interface Type ID	MCARD_ID0	105	106	USB_N	
	MCARD_ID1	107	108	GND	
	MCARD_ID2	109	110	MCARD_PWRBTN_L	Buttons
	GND	111	112	MCARD_RSTBTN_L	
DEBUG	DEBUG_IF4	113	114	MCARD_NMIBTN_L	
	DEBUG_IF5	115	116	GND	
	DEBUG_IF6	117	118	MCARD_AUX_SOL_CTRL_L	TX/RX SOL Ctrl
	DEBUG_IF7	119	120	RSVD	RSVD
	GND	121	122	GND	
Control	MCARD_LOCAL_LOCK_L	123	124	MCARD_RXD1	ICMB
	MCARD_SYS_SPKR_DATA	125	126	MCARD_TXD1	
	MCARD_CLR_CMOS_L	127	128	MCARD_TXD1_ENABLE	
	GND	129	130	GND	
Misc	CLK_32768	131	132	ALL_PWROK	Status
	GND	133	134	SYS_THERMTRIP_L	
Card Detect	MCARD_DETECT_L	135	136	SYS_INTRUDER_L	
	GND	137	138	SYS_LINE_AC_L	
Single Wire Analog Sensors	ACOMP_ADC0	139	140	SYS_PWRBTN_L	
	ACOMP_ADC1	141	142	SYS_RSTBTN_L	
	ACOMP_ADC2	143	144	SYS_NMIBTN_L	
	ACOMP_ADC3	145	146	SYS_SMI_L	
	ACOMP_ADC4	147	148	SYS_PCI_RST_L	
	ACOMP_ADC5	149	150	SYS_ACPI_STATE0	
	GND	151	152	SYS_ACPI_STATE1	
Control	MCARD_SCI_INT_L	153	154	SYS_ACPI_STATE2	
RSVD	RSVD	155	156	SYS_IO_EXP_INT_L	
	GND	157	158	GND	
RSVD	RSVD	159	160	MCARD_RMII_REF_CLK	NC-SI
	RSVD	161	162	MCARD_RMII_CRS_DV	
	RSVD	163	164	MCARD_RMII_TX_EN	
	RSVD	165	166	MCARD_RMII_RX_ER	
	GND	167	168	GND	

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

	GND	169	170	MCARD_TX_H	Mgmt LAN
Power	VDD_3.3_DUAL	171	172	MCARD_TX_L	
	VDD_3.3_DUAL	173	174	GND	
	VDD_3.3_DUAL	175	176	MCARD_RX_H	
	VDD_3.3_DUAL	177	178	MCARD_RX_L	
	GND	179	180	POE_GND	
	GND	181	182	POE_PWR	
	VDD_5_DUAL	183	184	POE_PWR	
	VDD_5_DUAL	185	186	POE_GND	
NC-SI	MCARD_RMII_TXD_0	187	188	LAN_BUSY_LED_L	
	MCARD_RMII_TXD_1	189	190	LAN_LINK_LED_L	
	RSVD	191	192	GND	
	MCARD_RMII_RXD_0	193	194	RSVD	RSVD
	MCARD_RMII_RXD_1	195	196	GND	
	GND	197	198	RSVD	RSVD
RSVD	RSVD	199	200	GND	

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Chapter 5 OPMA Feature Card Mechanicals

This chapter describes all mechanical aspects of the mCard subsystem board.

5.1 Board Mechanical Outline

The mechanical outline in this section represents the maximum X:Y:Z dimensions of an mCard. The mCards may be shortened in the Y dimension only. The X dimension (i.e., the length of the edge that contains the gold fingers) must remain a constant between all mCards. The mCard Y dimension should be kept as short as possible by the card IHV. The only requirement is that such shortening must not interfere with mechanical keepers as provided by SO-DIMM DDR2 connector manufacturers in any mounting configuration. The maximum Y dimension was chosen to accommodate additional components required to implement an M3 solution. It is expected that M1 and M2 implementations will be noticeably shorter. Lower cost and smaller size are expected to be competition points for mCards in the field.

This mechanical specification assumes that components are mounted on both sides of the mCard. The maximum height for components mounted on the bottom side of the mCard is specified at 2.54 mm (0.10 inches) to allow for other devices to be placed on the motherboard while mCard is mounted horizontally. The maximum height for topside components is specified at 7.98 mm (0.314 inches).

The current version of mCard mechanical drawing (refer to the Note on this page) defines specific keying requirements to allow the mCard to fit in SO-DIMM DDR2 socket on the motherboard used for OPMA purposes. Refer to the keying dimensions defined in the mechanical drawing for the mCard in Figure 4 on page 47.

No special mechanical keying that disallows SO-DIMM DDR2 DRAM devices to be plugged into an OPMA connector is required because SO-DIMM DRAM devices are not used in rack mount, blade, or pedestal servers/workstations.

For electrical parameters like plating and metallization on the card edge, refer to the SO-DIMM JEDEC specification, MO-224D.

Note: Change in OPMA mCard keying for specification release version 1.2 and above.

What changed: OPMA mCard polarization key is now DDR2 style only.

The mCard mechanical key specified in 1.0–1.1 versions of the OPMA specification defined a double-wide slotted key which allowed card installation into either SO-DIMM DDR1 or SO-DIMM DDR2 based OPMA sockets on the motherboard. OPMA specification version 1.2 and later versions require the card key specification to mechanically conform to SO-DIMM DDR2 only.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Implications:

As a result of this tightened definition, future OPMA mCards will fit only in motherboards which use the SO-DIMM DDR2 connector as the OPMA socket. OPMA cards manufactured to the 1.0–1.1 versions of the specification will continue to work in all motherboards due to the extra mCard key width but will have the potential for mechanical pin misalignment in some corner cases of dimensional tolerance stack up between the mCard, and the socket.

Reason for Change:

It's been determined that the original double-wide keying scheme could cause misalignment between socket pins and mCard gold fingers. This has been seen only under worst case manufacturing tolerances between the SO-DIMM socket and the OPMA mCard, namely where the mCard was manufactured to the low end of the allowed tolerance, and the socket used provided the maximum allowable width. Since the SO-DIMM DDR2 socket has now achieved broad availability and commodity pricing (which was not the case when the OPMA spec was originally introduced), this specification modification is appropriate at this time.

Requirements / Recommendations:

- The above change requires that all future OPMA compliant motherboards be equipped with SO-DIMM DDR2 sockets.
- The above change requires all OPMA mCard vendors to design new mCards with SO-DIMM DDR2 keying only.
- mCards already manufactured to the 1.0–1.1 versions of the OPMA specification will still fit into the newer SO-DIMM DDR2 sockets. Since mechanical alignment of the pins is not guaranteed in these cases, alignment should be visually inspected before applying power to the system.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

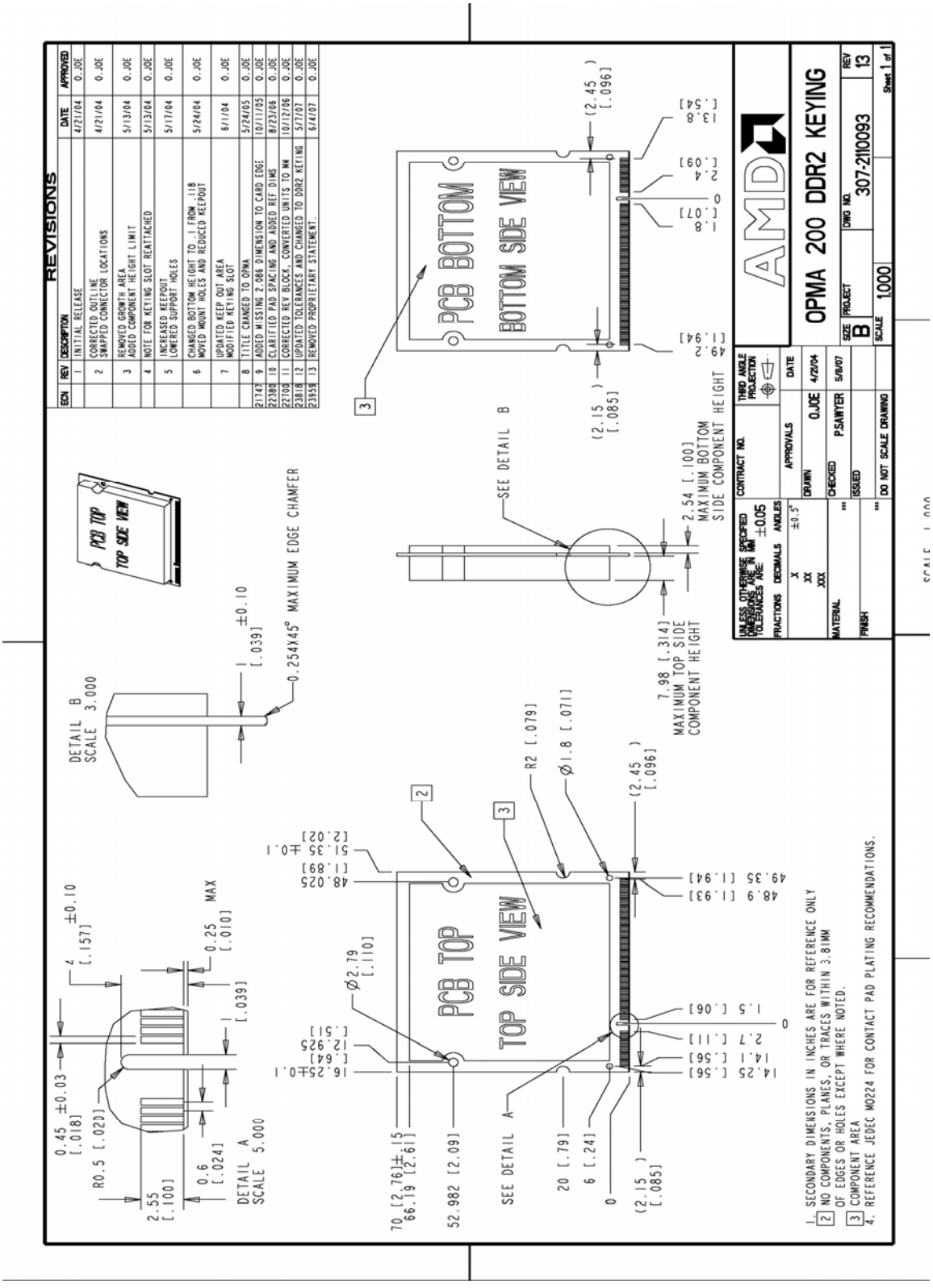

Figure 4. MCard Mechanical Form Factor

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

5.2 Mechanical Strategy and Configuration

Most management subsystems have been implemented as down solutions. In this case, there is no question regarding circuit device allocation since all components are located on the motherboard. However, device location becomes a factor when creating a modular, feature card-based solution, especially when the modularization supports solutions with varying capabilities. It must be ascertained whether the circuit, device, connector, etc., in question is to be located on the feature card or on the motherboard. Such decisions should be driven by the goals and expected usage models.

The basic goals of the mCard mechanical and connector strategy are:

- Low cost
- Small size
- Flexibility in placement on the motherboard

Cost may be a key element. Costs should be partitioned as fairly as possible given the goals and usage models. For example, motherboard ODMs do not want to be saddled with paying for high cost devices that will only be used with high-end management solutions. Conversely, mCard IHVs simply can't make a business if their cards cost too much.

The partitioning of these physical devices was done with the basic goals in mind along with the expected usage model that the majority of server systems that are shipped to the end customer with OPMA connectors designed in will also come with some type of manageability subsystem. The top two scenarios are expected to be motherboards that add an OPMA connector to an existing down solution as an upgrade path, and motherboards that use OPMA as the sole manageability interface and that ship with M1 style mCards installed.

As a result of the above, OPMA positions level translation components, etc., on the motherboard. The main components left on the mCard are those such as BMC, SDRR/FRU/SEL SEEPROMS, and ASICs for M3 implementations. Also, not all motherboards will natively provide operating voltage (such as 2.5 volts) for the BMC device. In order to ensure long term cross compatibility of OPMA feature cards and motherboards, BMC-specific operating voltages other than 3.3 VDC and 5.0 VDC must be generated on the mCard itself.

5.3 OPMA LAN and Serial Port Connector Scheme

The mCard mounts anywhere on the motherboard, limited only by motherboard-specific factors. It can be mounted horizontally (i.e., blades and 1U–2U chasses) or vertically (4U chassis). The mCard UART (management serial port) signals are routed through the OPMA connector to the manufacturer's choice of management serial port connectors on the chassis (typically either an RJ-11/45 or 9-pin D shell connector). Connector type and chassis location of the manageability serial port is beyond the scope of OPMA. However, OPMA assumes that the management serial port is shared with the system serial port. This arrangement is required so that both the CLI and

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

the system text consoles (BIOS setup, etc.) can share the same serial legacy concentrator infrastructure that is commonly found in the field today. Refer to Chapter 6 beginning on page 51 for more information on sharing the serial-port connector on the back panel.

The dedicated management LAN Ethernet signals from the OPMA connector are routed through the motherboard to a chassis-mounted RJ-45 connector. Except in the upgrade kit mode, if an M3 or M2 mCard with dedicated NIC is plugged into the OPMA connector, this RJ-45 connector carries all of the management traffic. In these cases, the transformer matrix is located on the mCard itself, while the EMI ferrite and the RJ-45 is part of the motherboard. *RJ-45 connectors with integrated magnetics should not be used on the motherboard for mCard out-of-band management traffic.*

This cost effective solution also provides the benefit that the magnetics are adjusted based on the characteristics of the dedicated M2/M3 NIC. Each mCard might use a different NIC device on it, so a common magnetics solution put near the RJ-45 header may or may not be suitable for all the NIC devices. If an mCard with a NC-SI LAN interface is plugged into the OPMA connector, the management RJ-45 connector should contain a blank plug or be covered by a cover plate to indicate that this port is not being used. Figure 5 shows the configuration of the management LAN and UART connectors for an mCard.

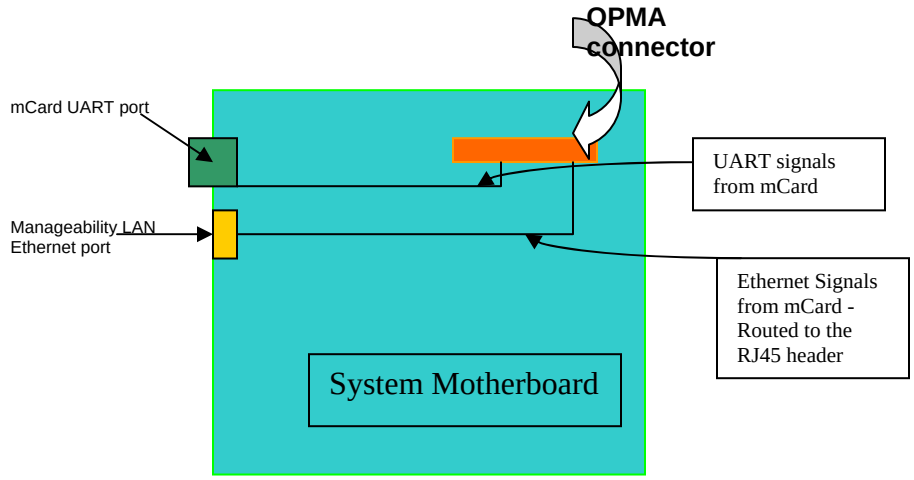

Figure 5. Management LAN and UART Connectors for an mCard

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Chapter 6 OPMA SoL Support

In this chapter, the phrase “UART handshake signals” refers to all mCard UART signals except TxD and RxD. OPMA must abstract the UART handshake signals of common BMC chips at the level of the OPMA connector. Consideration must be made for how various BMC chips and operational modes for those chips can be abstracted at the connector and hardware resource level to allow mCard interoperability. OPMA must also define a scheme whereby the system board UART is automatically connected to the serial port connector when no mCard is installed. This chapter provides information on how to implement this scheme on the motherboard in a manner designed to allow the motherboard to accept mCards that use different types of BMCs.

Serial over LAN (SoL) is the ability to redirect a local text console over the management LAN to a remote viewing/control station. Text-based BIOS setup screens and OS control consoles can be redirected using this method. SoL is a replacement technology for the legacy serial concentrators and can lead to lower installation and maintenance cost due to reduced cabling and the elimination of serial concentrators from the management infrastructure.

OPMA must abstract the SoL capabilities of common BMC chips at the level of the OPMA connector. This requires some motherboard support because BMC chips have varying numbers and types of onboard peripheral resources. For the purposes of explanation, the focus in this chapter is on two exemplar BMC devices that handle SoL in different ways. These devices are referred to as “Device A” and “Device B.” It is assumed that all BMC-device UART support operates sufficiently like one of these implementation classes. Device A contains an integrated 16550 UART emulator. It presents standard 16550 registers to the LPC bus at legacy ISA addresses. Device B contains integrated UARTs, but none of their register interfaces are exposed to the host system. The main differentiating factor is the system accessibility of the UART registers through the LPC bus on Device A. On Device B, only the BMC has access to these registers. It is possible to have an mCard whose BMC does not have an integrated, system-accessible 16550-compatible UART, but which includes 16550 UART functionality (i.e., by means of a separate device located on the mCard). In such a case, this mCard can be connected to the system and treated for all intents and purposes like a Device A-based mCard implementation as long as the registers of the external UART are accessible by the system on the LPC bus.

If a Device A-based OPMA card is installed, the host system UART should be disabled by the system BIOS, and the Device A UART should take over all functions of both the host system UART and of the BMC UART (typically used for implementing the command line interface (CLI)). In this usage scenario, SoL is achieved without feeding back RS-232 encoded data to the mCard from the host system UART. In this case, all UART handshake signals are driven from Device A.

Alternatively, if a Device B-based mCard is installed in the system, the host system UART generates the text-based BIOS and OS control consoles. To support SoL on Device B class devices, the Device B BMC must capture this console stream. OPMA supports this capture by

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled “OPMA SPECIFICATION LICENSE AGREEMENT.”

feeding back the system UART signals to the mCard where local circuitry under the control of the BMC subsequently routes the host UART data stream as needed.

The host system UART signals must be automatically routed to the back-panel serial port connector in the event that no mCard is installed. This routing drives the requirement for a motherboard-mounted UART TxD and RxD signal mux as well as a mux for UART handshake signals such as RTS, CTS, DSR, RI, etc. When an mCard is installed, its card detect signal (MCARD_DETECT_L) may, for example, be used to directly control the A:B selection of the motherboard mounted UART TxD/RxD signal mux (designated "X4" in Figure 6 through Figure 11 on pages 54–59). The green line represents the installed mCard in Figure 6 through Figure 11. When any type of mCard is installed, the "A" path should be automatically selected. When no card is installed, the "B" path is selected, and the host system UART TxD and RxD signals are connected directly to the back panel RS-232 connector.

The TxD and RxD signals from the host system UART go to the MCARD_TXD_SOL and MCARD_RXD_SOL signals on the OPMA connector. The TxD and RxD signals from the management UART go to the MCARD_TXD0 and MCARD_RXD0 signals on the OPMA connector.

To reduce complexity and connector pin count while handling all scenarios, UART handshake signals are treated differently than UART TxD and RxD signals. Handshake signals are muxed to the connector through another signal mux, which is designated "X5" in the following figures.

Figure 6 through Figure 11 show how OPMA supports both the Device A and Device B class mCards. Everything within the blue dashed line in the figures is physically located on the mCard, while everything else is on the host motherboard. The thick, solid lines indicate that a signal path is active or that a signal is asserted. Thin lines indicate a signal path that exists but which is inactive or may also indicate a signal that is deasserted. Black solid or dotted lines indicate TxD/RxD data paths. Blue lines indicate UART handshake signal paths. Red lines are mux control signals.

The figures showing Device A and Device B-representative mCard implementations (i.e., items inside the blue dashed lines) are for illustration only and are not meant to imply specification requirements. They are provided to show how these different classes of mCard might be made to interoperate in a single socket infrastructure despite their UART resource differences. For example, the technologies used for muxes X2-X4 are undefined by this specification. The motherboard-mounted muxes may be any technology desired as long as the signal routing through these muxes is performed in compliance with this specification. Figure 6 on page 54 and Figure 7 on page 55 depict two very similar scenarios for Device B class mCards. Both figures show that the system UART is routed to the serial port connector. The TxD and RxD signals actually go onto the mCard and then through a series of muxes before being routed to the TxD and RxD outputs of the mCard. This routing allows the BMC to monitor the keystrokes entered from a terminal plugged into the serial port connector. Monitoring (a/k/a snooping) is required so that the BMC, which controls the UART muxes, can switch UART signal routing (both TxD and RxD signals) to the serial port connector and back when certain IPMI-defined escape sequences are detected. Also, the system UART handshake signals do not cross the OPMA connector. Instead,

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

they are fed into a handshake signal mux that is mounted on the motherboard (depicted by X5 in the figures).

The system UART is routed in the previously described way when the user wants access to a console that is generated either by the BIOS (i.e., setup screen) or the operating system (i.e., SAC or Special Administrative console). The only difference between the figures is that Figure 6 on page 54 depicts an mCard where the BMC UART implements UART handshake signals, while Figure 7 on page 55 shows a BMC UART that does not. Neither of these cases route the BMC UART handshake signals to the serial port connector because the system UART has been routed to the connector. The mCards that do not drive the UART handshake signals from the BMC must, on the card, terminate all of the UART handshake signals to their active states. Figure 8 on page 56 and Figure 9 on page 57 are related in the same way as Figure 6 on page 54 and Figure 7 on page 55 are related. However Figure 8 on page 56 and Figure 9 on page 57 show the UARTs set up for SoL and CLI operation. In this mode, any console that is generated by the system UART is piped over the Ethernet to a remote console so that the operator can remotely access the BIOS setup screen. remotely. In addition, this mode presents the BMC CLI to the local serial port connector. Figure 8 on page 56 depicts a BMC that implements handshake signals, and so the heavy blue line from the BMC UART0 indicates the route to the serial port connector. Figure 9 on page 57 shows a BMC that does not implement UART handshake lines. The mCard must terminate these handshakes signals to the actives states for all signals, and these terminated signals are then routed to the serial port connector.

Figure 10 on page 58 depicts a much simpler conceptual arrangement because the 16550 UART registers of the Device A device are available to the system on the LPC bus. In this case, the system UART is always disabled by system BIOS at boot time and the system UART is grayed out in the figure to reflect this.

Finally, Figure 11 on page 59 depicts a system operating without an mCard installed. In this case, the system UART must be automatically connected to the back panel. Mux X4 is controlled by the MCARD_DETECT_L. Mux X5 is controlled by the MCARD_AUX_SOL_CTRL_L signal.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

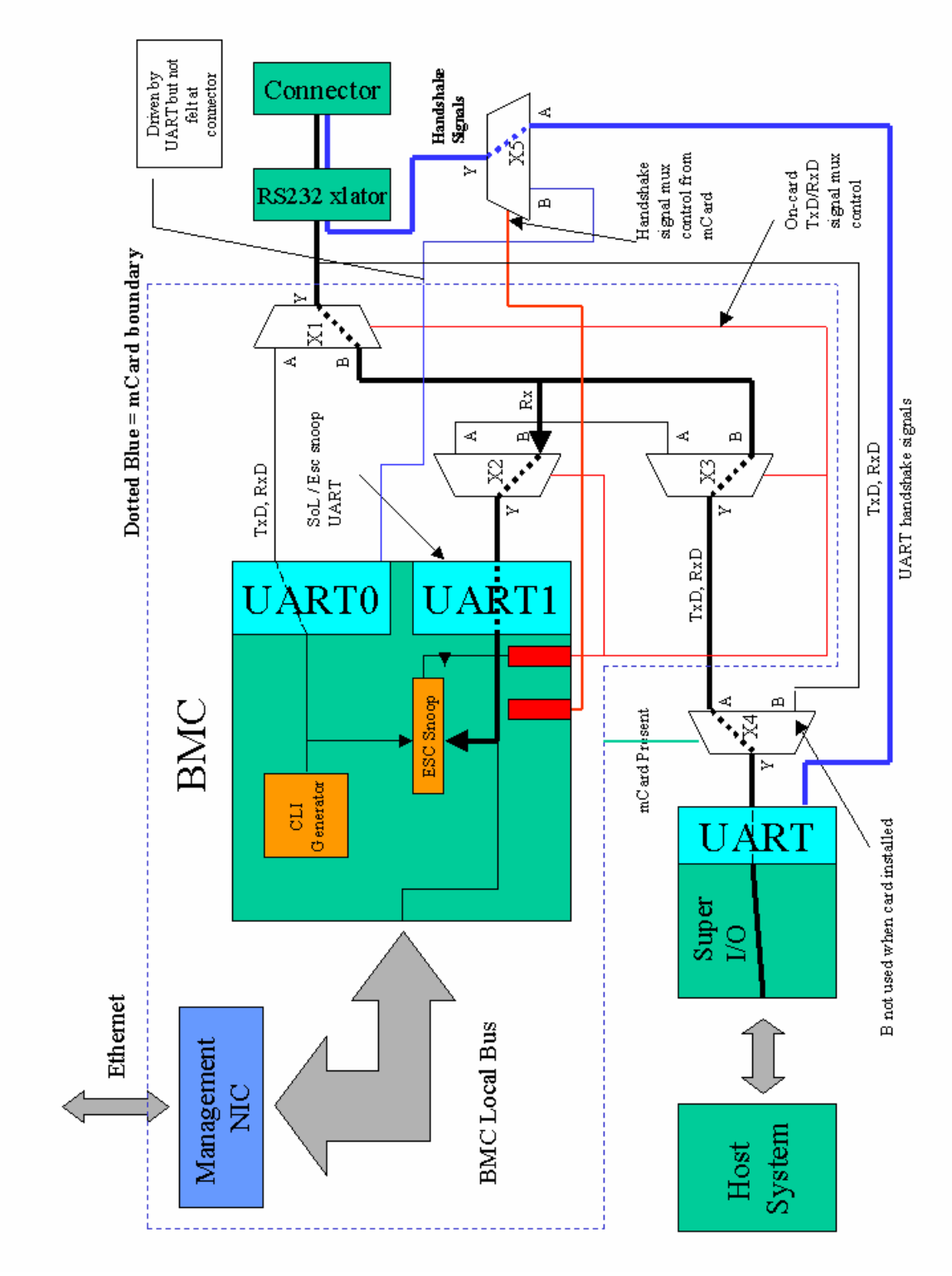

Figure 6. Device B mCard with Host UART Connected to Serial Port Connector (A)

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

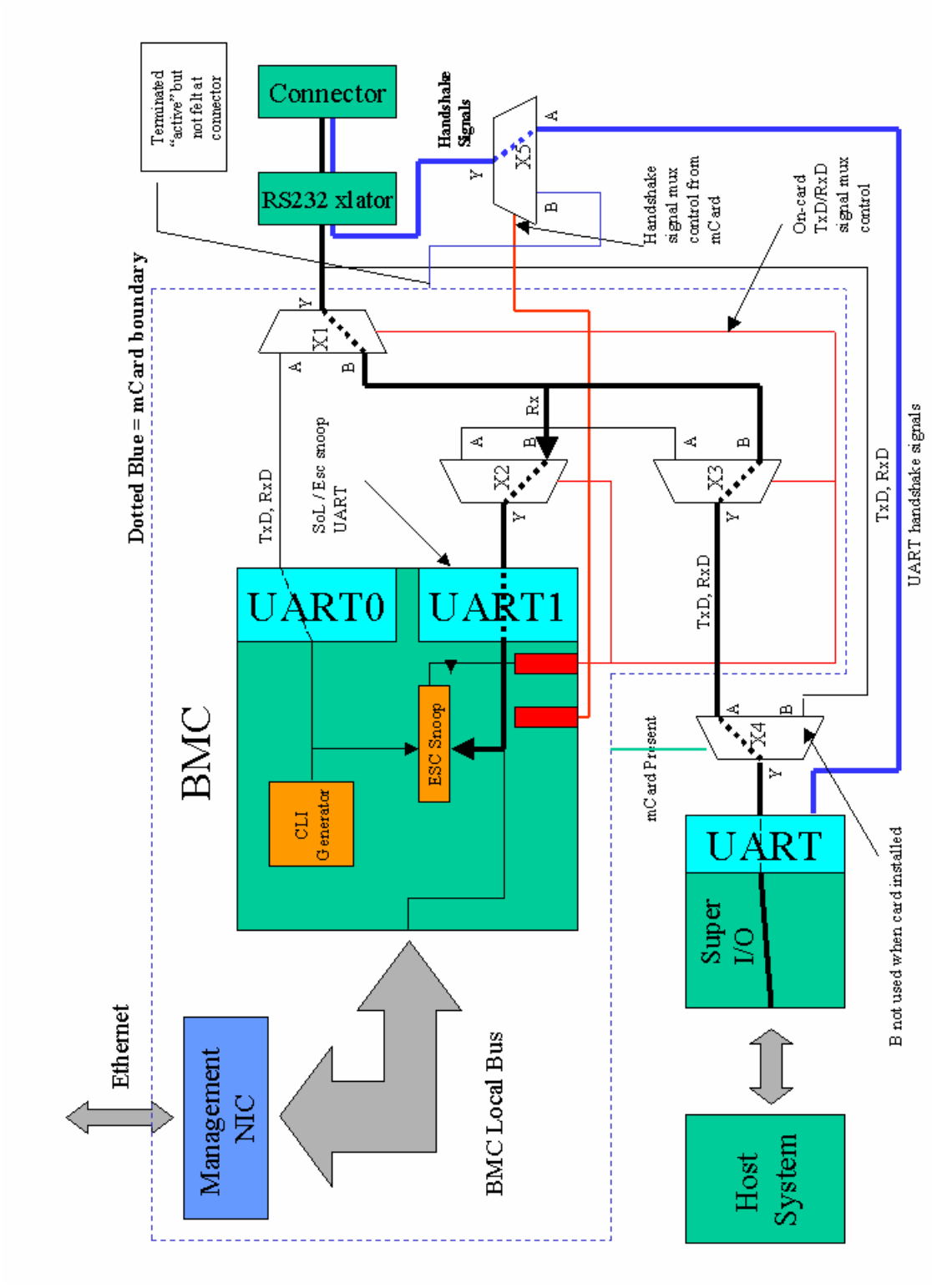

Figure 7. Device B mCard with Host UART Connected to Serial Port Connector (B)

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

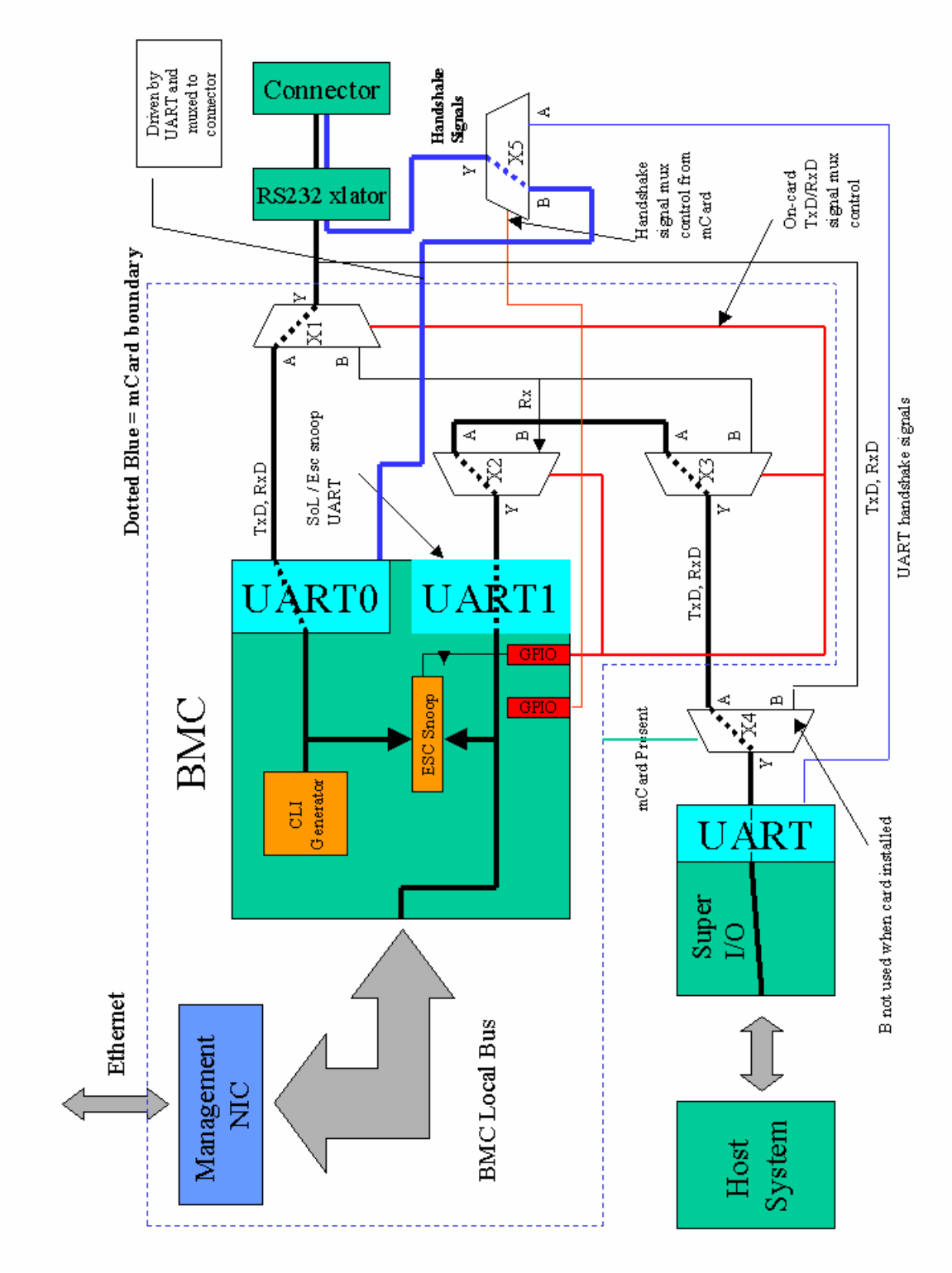

Figure 8. Device B Representative mCard in SoL Mode (A)

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

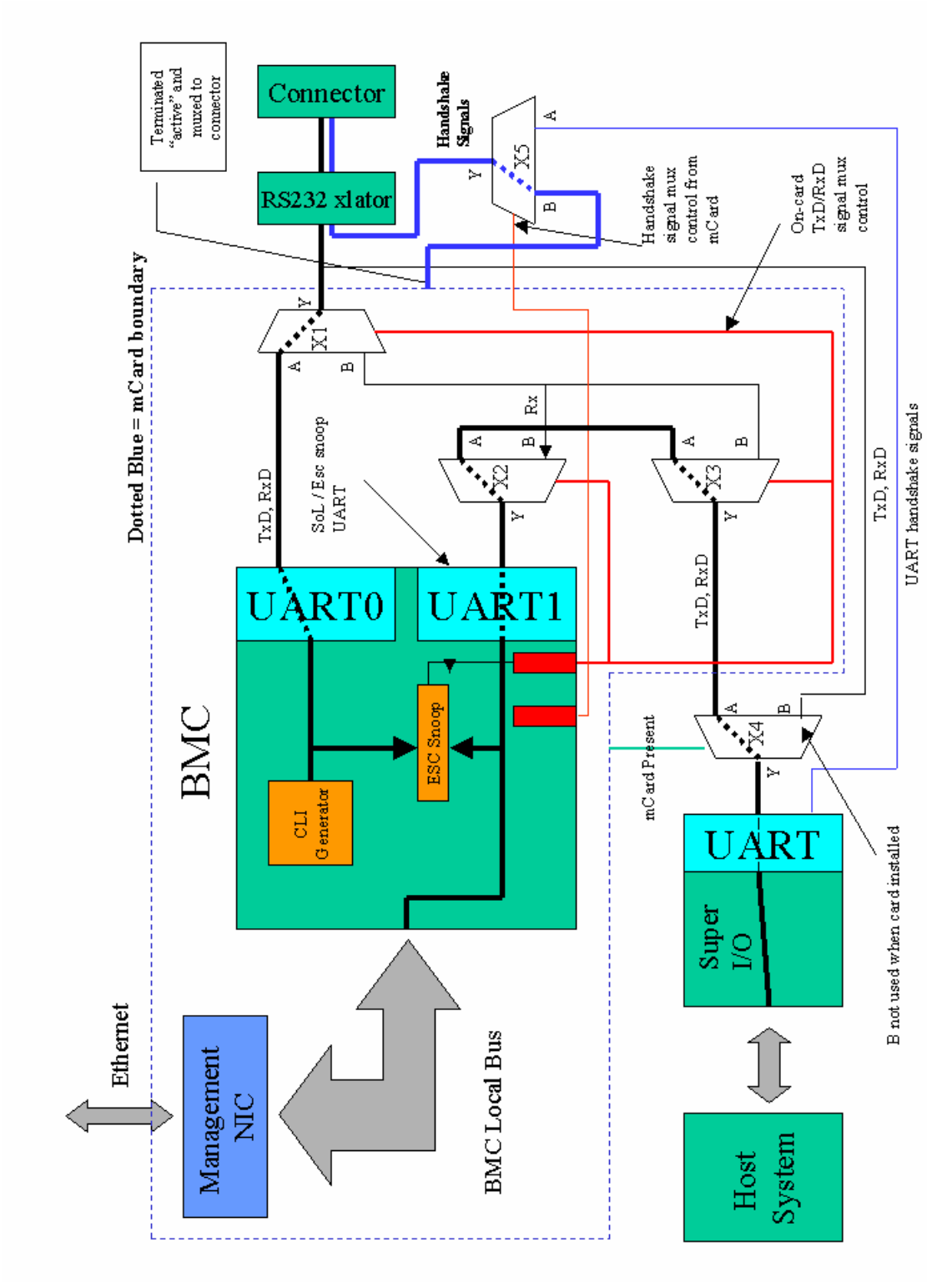

Figure 9. Device B Representative mCard in SoL Mode (B)

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

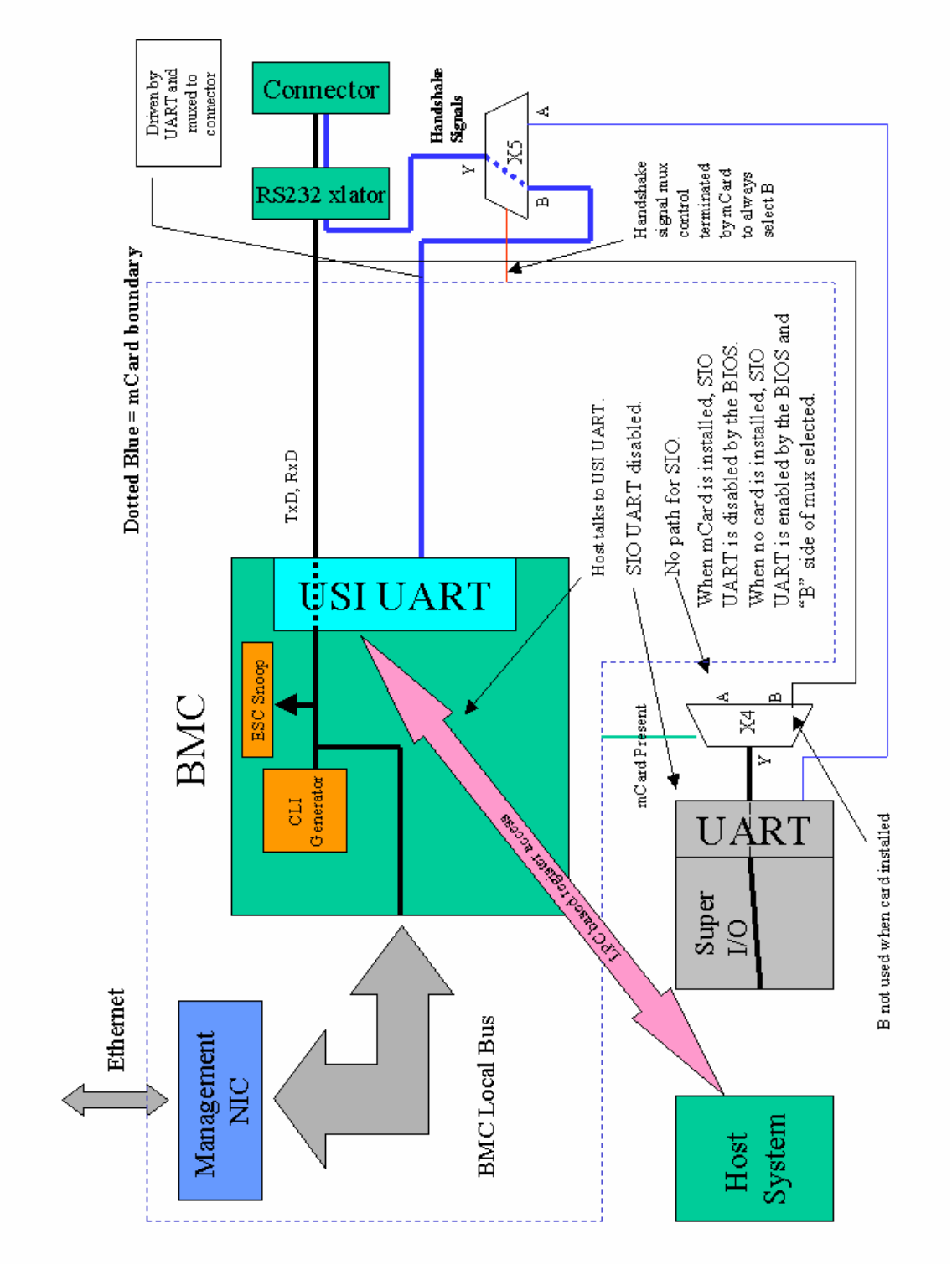

Figure 10. Device A Representative mCard in Both CLI and SoL Modes

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

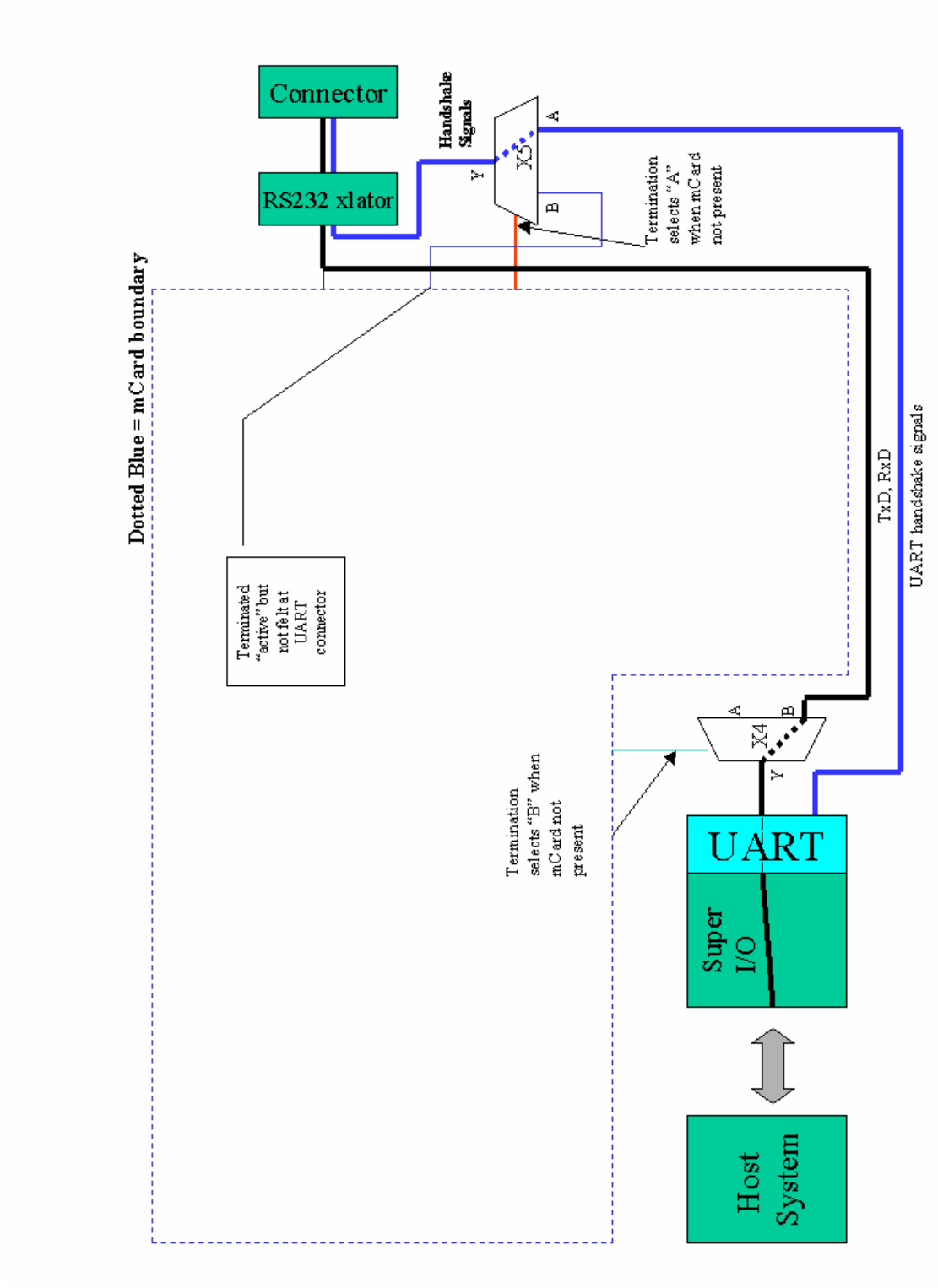

Figure 11. Host Serial Port Operation with No mCard Installed

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Chapter 7 Specification Version Compatibility Rules

COTS infrastructures achieve cost competitiveness by building volume quantities. However, stock on hand may be exposed to obsolescence if care is not taken toward preserving backward compatibility when specifications change.

When the system BIOS boots, it sends the SetSystemTypeIdentifier command to the mCard. One of the parameters for this command is the motherboard specification level compliance. The mCard accepts this command, processes the associated parameters, and makes a decision as to whether the mCard is compatible with the motherboard. This decision can be made on the strict recognition of the motherboard's OEM ID and management sensor Implementation ID (typical) or it can be based on a number of other factors including the OPMA specification compliance level. For the purposes of OPMA, specification version compatibility means that newer specification revisions must ensure that the following conditions are observed.

Note that "compatible" refers mainly to OPMA electrical and hardware feature compatibility. For an mCard to work with a given motherboard it is required to have both hardware compatibility between the OPMA card and the motherboard along with BMC firmware awareness of the sensor set on the motherboard.

An mCard that is compatible with an OPMA motherboard may work by ensuring that the BMC firmware understands the motherboard's sensor list and sensor topology. An mCard that is not compatible with an OPMA motherboard (because the mCard is compliant to a significantly different and newer version of the specification) can not be made to work with that motherboard.

7.1 Electrical Compatibility

When mCards and motherboards that are compliant to varying degrees of the OPMA specification are connected, no electrical damage may occur. Any electrical changes that occur in later versions of the specification that alter pin voltages or semantics must be isolated from the connector by the motherboard and the mCard until both agree that they are of compatible levels. All connector signals that are listed as reserved should be No Connects on both the motherboard and the OPMA card side to assure electrical compatibility with later specification versions.

7.2 Backward Compatibility

OPMA backward compatibility is viewed from the standpoint of the motherboard. A motherboard that is OPMA 1.0 compliant is obviously compatible with an mCard that is also OPMA 1.0-compliant. An OPMA motherboard that is compliant with a version of the OPMA specification that is greater than 1.0 must still be 100% compatible and support all features of an existing OPMA 1.0 mCard. Of course, new features that might be part of the later version of the

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

specification would not be known by the 1.0 mCard and those features would not be supported by the combined system.

7.3 Forward Compatibility

If an mCard is compliant with a later version of the specification than the motherboard, the mCard determines whether it will work with that motherboard and which (if any) of the new features will be supported. For example, minor specification revisions may clarify feature usage on the mCard so that mCards compliant with spec revision levels 1.1, 1.2, etc., will be compatible.

For purposes of forward compatibility, the BIOS-BMC interactions are kept simple and only occur during system boot. While system BIOS can be changed and fields updated, a more test-intensive process is to release a new BIOS into the field rather than to release a new version of mCard firmware. Any small change to the BIOS could have system wide impacts while changes to the management subsystem firmware are more focused and easily validated.

7.4 Compatibility for Transition from MCARD_I^{2}C_SIDEBAND_NIC to MCARD_I^{2}C_CPU Bus

With the introduction of NC-SI in OPMA specification revision 1.3 and above, support for the previously defined MCARD_I2C_SIDEBAND_NIC bus is being re-assigned as an SMBus 2.0 interface to host system CPU(s).

Compatibility between motherboards and mCards must be handled using the OPMA SetSystemTypeIdentifier command compliance field during BIOS-BMC boot interaction (refer to Section 12.2.2 on page 87). BIOS on motherboards supporting the older I^{2}C/SMBus based sideband interface must program this specification compliance field to 0x0100 or 0x0120 (indicating 1.0 or 1.2 specification compliance). BIOS on motherboards supporting NC-SI NICs must use the value of 0x0130 and above for the compliance field.

OPMA mCards supporting revisions 1.3 and above must not communicate any NIC sideband protocol on pins 51 (CLK), 53 (Data) and 55 (alert), and must use them only for the SMBus 2.0 connectivity and interaction protocol with host system CPU(s). Refer to sections 3.2.16 on page 31, and 3.2.20 on page 33 for details.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Chapter 8 Motherboard Hardware Support for OPMA Compliance

The following sections provide more details on motherboard support required for OPMA compliance along with sample implementation circuitry for OPMA-required signals. Refer to the appropriate sections of this specification for detailed descriptions of the OPMA signals associated with these support requirements and functions. Note that the sample circuits and devices on the circuits shown in this section are examples only, the system designers may design other circuit models with devices on the motherboard to achieve the same functionality. The OPMA specification requires that these signals be utilized to only achieve the intended functionality supported by the signals.

8.1 General Signal Termination

OPMA signal termination requirements for the motherboard are given in Table 5 on page 37.

8.2 MCard SMI Generation Support

No separate SMI generation output signal is available through the OPMA connector interface. If an mCard is required to generate an SMI to the host system then it uses the SERIRQ-based method. OPMA-compliant motherboards must provide the LPC SERIRQ support.

8.3 Fan Tachometer Read Back

The OPMA connector specifies four tachometer input signals for monitoring fans. Multiplexers are required on the motherboard for systems with more than four fans and OPMA connector defines fan select mux control signals for controlling the multiplexers. Systems with four or less fans could directly hook up to the OPMA fan tachometer inputs with proper signal conditioning, in such cases the mux control signals have to be properly terminated on the motherboard. Figure 12 and Figure 13 on page 64 show the block diagram and an example of the fan tach monitoring circuit respectively, for a system with sixteen fans. In this example circuit, the 16:4 multiplexer is controlled by four mux select signals provided across the OPMA connector and the mCard controls these signals to select one bank of up to four fans to be monitored at a time. The motherboard must provide a 16:4 multiplexer for the mCard to perform this function in this case. The motherboard must also provide appropriate pull-ups for the fan select signals. Motherboard must provide proper terminations for any unused tach inputs or mux select control signals.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

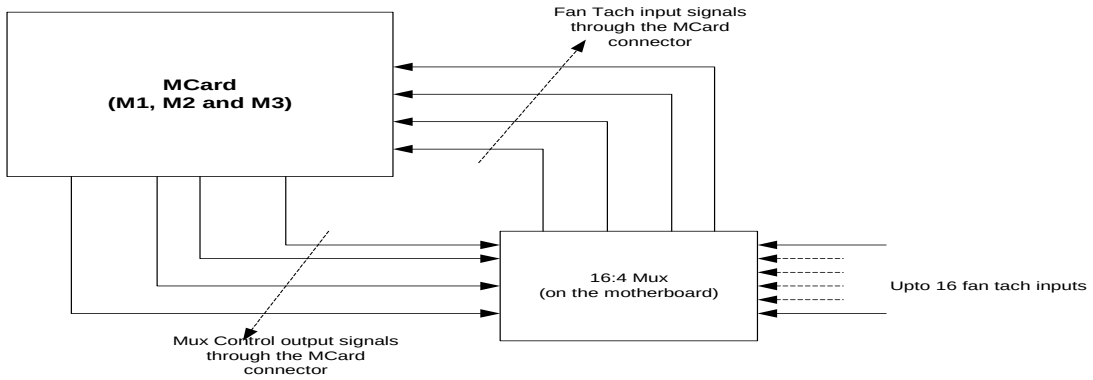

Figure 12. Fan Tach Monitoring Circuit Logic Block Diagram

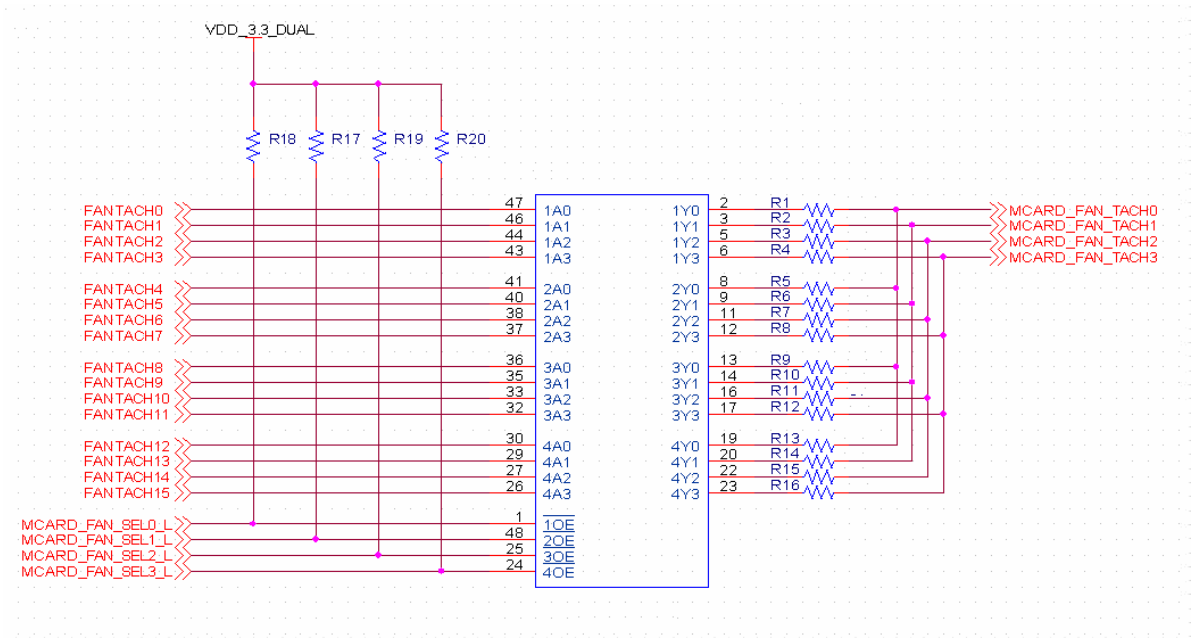

Figure 13. Fan Tach Monitoring Circuit Example

8.4 Fan Speed Control

The motherboard must provide the necessary circuit that converts PWM pulses from the mCard output to an analog DC level that is then amplified to actually control the fans. It is part of the fan speed control circuit. Figure 14 shows the block diagram of the fan speed control circuit. Figure 15 on page 65 shows an example of the fan speed control PWM-to-DC converter circuit. If the platform is using PWM controlled fans this circuitry is not required, but proper hardware buffering might be required for directly routing the PWM signals to the fan circuitry.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

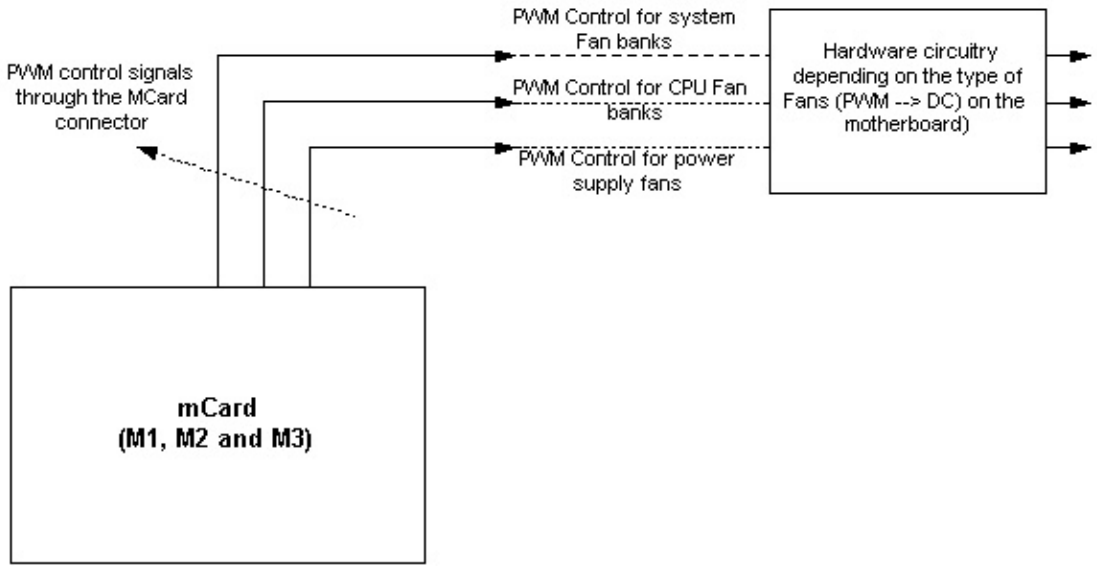

Figure 14. Fan Speed Control Circuit Logic Block Diagram

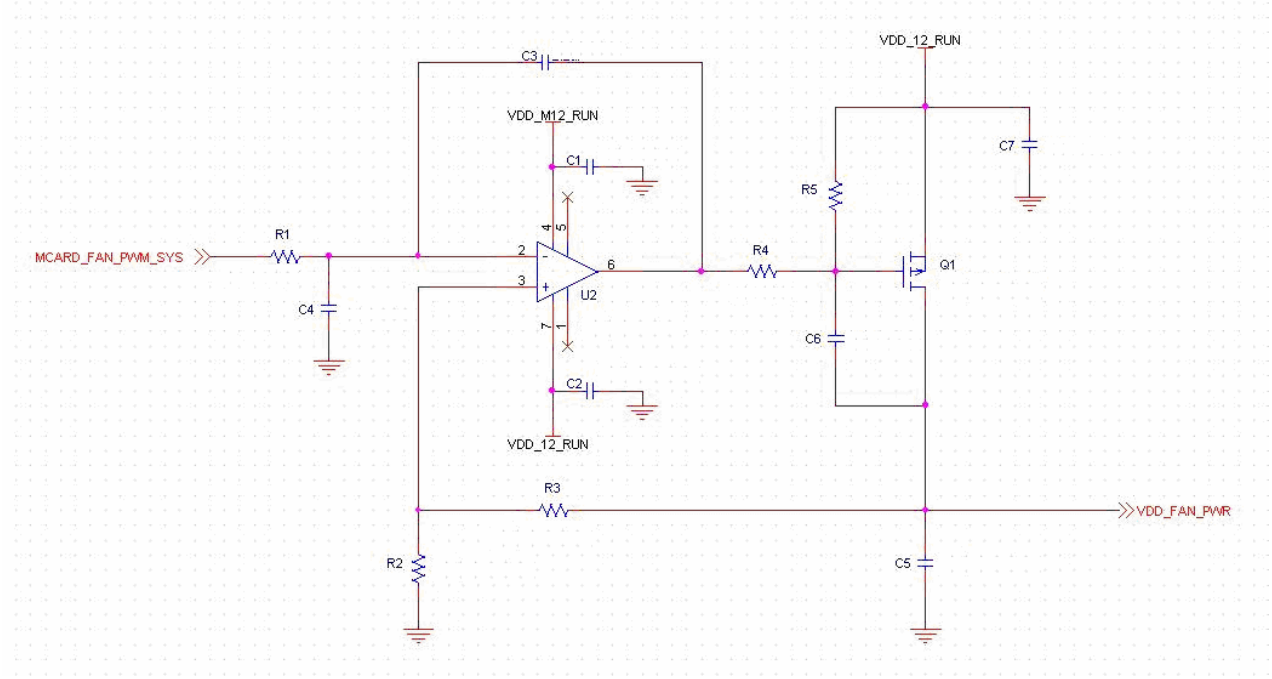

Figure 15. Fan Speed Control PWM-to-DC Converter Circuit Example

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

8.5 Clear CMOS Circuit

The motherboard must provide hardware circuitry for clearing the BIOS CMOS contents when the mCard BMC activates the output to clear the CMOS. This provides a remote, software controlled way of clearing the system CMOS without having to open the system chassis. This circuitry must co-exist with the traditional jumper-based method provided by the motherboard to clear the CMOS contents. The BMC asserts MCARD_CLR_CMOS_L for a minimum of 250 ms to clear the CMOS. The motherboard hardware ensures that the CMOS is cleared if this signal is held active by the BMC for >= 250 ms. If the CMOS RAM chip requires a longer CMOS clear signal activation period in order to clear the CMOS, motherboard hardware must be provided to stretch the 250 ms pulse (not shown in Figure 16) provided by the BMC. The circuit shown below is a reference example for one way to achieve CMOS clearing using MCARD_CLR_CMOS_L signal. Systems using other methods for clearing CMOS must utilize the MCARD_CLR_CMOS_L signal from the OPMA interface and provide necessary circuit to support the functionality of clearing CMOS through the OPMA mCard.

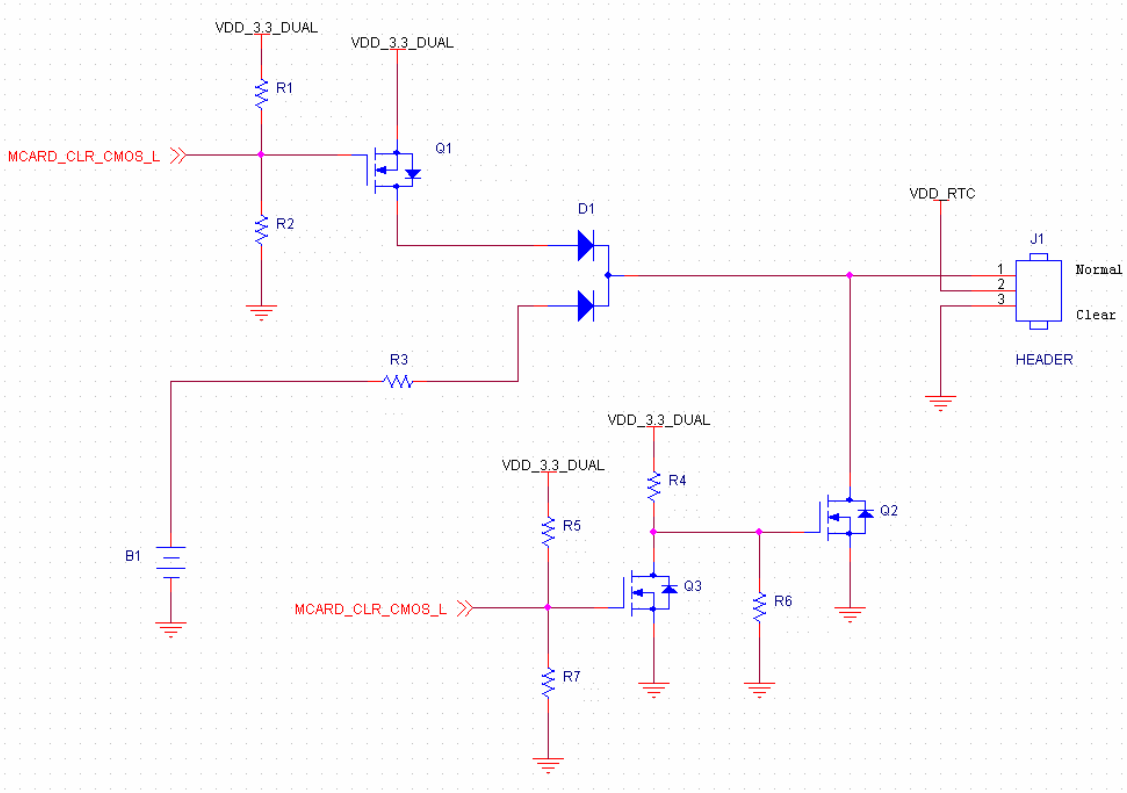

Figure 16. CMOS Clearing Circuit Example with MCARD_CLR_CMOS_L Signal

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

8.6 System Speaker Control Circuit

The motherboard must provide the hardware circuitry for the mCard to share control of the system speaker (transducer) with the Southbridge or I/O hub. This signal allows the mCard to drive the system speaker with the MCARD_SYS_SPKR_DATA signal to generate audible alerts. It may be used as part of the server identification capability in a rack environment. Figure 17 shows an example of the speaker control circuit.

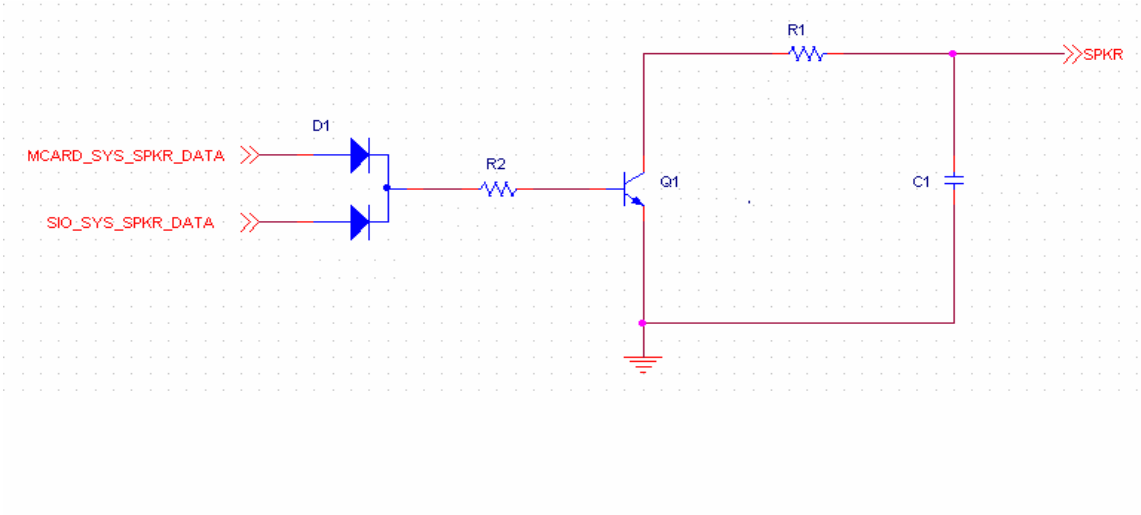

Figure 17. Speaker Control Circuit Example

8.7 Local Access Lock Out

The motherboard may provide hardware circuitry for locking out local access to the system for the purpose of providing security in a server farm environment. Using this recommended but optional feature, the mCard may be commanded remotely to lock out some or all local console access to its host. Lock out, if implemented, should disable local access to front panel reset, power, and NMI switches as a minimum. It is recommended that this capability also be used to lock out local access to PS2/USB keyboard and PS2/USB mouse.

Local lock out should *not* restrict remote access to these interfaces. In other words, when locally locked out, the remote system operator should still be able to assert system RESET, POWER, and NMI signals through the mCard. Also, remote lock out has no affect on the ability to use remote human interface devices such as remote serial console, and KVMoIP interfaces. It also has no effect on virtual, USB-based mass storage devices. Only local access is disallowed.

OPMA provides the MCARD_LOCAL_LOCK_L signal for this purpose. When this signal is activated by the mCard, local access as previously described is disallowed.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Figure 18 shows an example of the partial local access lock-out circuit. Other circuitries may be included in the motherboard for remote lockout purposes. OPMA requires the motherboard to use the MCARD_LOCAL_LOCK_L signal in conjunction with the proper circuitry when providing the lockout functionality.

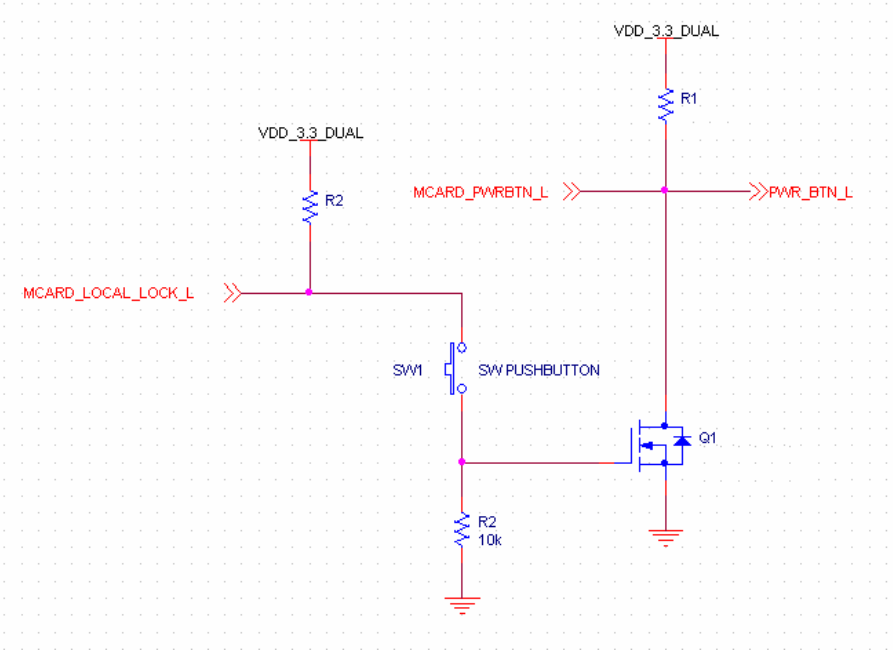

Figure 18. Partial Circuit Example for Local Access Lock Out

8.8 ACPI State Reporting

The motherboard must provide hardware circuitry to drive an encoded version of the ACPI state of the system to the OPMA connector. The mCard uses this information to provide accurate ACPI status to a remote location. This information is also useful in determining when certain sensors are and are not valid. Some sensors may not be valid in certain ACPI states. The assumption here is that the encoding function could easily be put into an existing PLD on the server motherboard. PLDs are typically used for purposes like power rail sequencing, etc. Table 7 lists the signal encodings for the ACPI states which are the status of SYS_ACPI_STATE2, SYS_ACPI_STATE1 and SYS_ACPI_STATE0 signals defined in the signal interface – SYS_ACPI_STATE2 being the most significant bit.

Table 7. Signal Encoding for ACPI State Reporting

State	ACPI State	Encoded Value
1	ACPI S0	000b
2	ACPI S1	001b

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

State	ACPI State	Encoded Value
3	ACPI S2	010b
4	ACPI S3	011b
5	ACPI S4, S5	100b
6	Reserved	101b
7	Reserved	110b
8	Reserved	111b

8.9 DVI–Digital Visual Interface

The motherboard must route the DVI-I signals from the video chip to the OPMA connector. The video chip selected for the system must be capable of providing DVI-I signals. The DVI-I signals are used by the M3 tier of the mCard for the purpose of video redirection in the KVMoIP (Keyboard, Video and Mouse redirection through internet protocol) feature. The DVI signals should be properly terminated on the motherboard to handle the case where either no mCard is installed or where an mCard that does not support KVMoIP is installed.

8.10 Firmware Debug Header

There may be a need to connect a debug probe to the BMC that is on the mCard being debugged in order to support firmware development. While it is not an OPMA requirement, the mCard IHVs may decide to put the associated debug header on the mCard itself.

However, the IHV does not know in advance where the ODM or OEM will place the mCard on the motherboard or what other physical obstacles might be present after the system is assembled. In this case, it may be difficult or impossible to connect the debug probe to an mCard based debug header after the system is assembled. In addition, the IHV may decide not to consume mCard board space with a debug header. As a result, OPMA supports the use of a motherboard-based Berg header (Figure 19) and defines which Berg pin must be routed to which OPMA connector pin as listed in Table 8.

2	4	6	8	10
1	3	5	7	9

Figure 19. Motherboard Berg Header Numbering Scheme (Top View)

Table 8. Debug Header-to-OPMA Connector Mapping

OPMA Connector Signal	Motherboard Debug Berg Pin No.
DEBUG_IF0	1
DEBUG_IF1	2

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

OPMA Connector Signal	Motherboard Debug Berg Pin No.
DEBUG_IF2	3
DEBUG_IF3	4
DEBUG_IF4	5
DEBUG_IF5	6
DEBUG_IF6	7
DEBUG_IF7	8
DEBUG_PWR0	9
DEBUG_PWR1	10

Beyond the routing of debug header pins to OPMA connector pins, OPMA neither defines nor assumes anything about the signals on these pins. The actual meaning and electrical characteristics of these pins is defined by each of the mCard OEMs. It is expected that there may be a different pin meaning for each BMC chip used in mCard implementations. It is the end user's responsibility to provide an adapter to connect a BMC-specific debug probe to a motherboard mounted generic debug header.

8.11 ICMB RS-485 Level Translation Circuit

The motherboard must provide level translation hardware along with the connector for the ICMB (Intelligent Chassis Management Bus). Physically, the ICMB is a multi-drop, multi-master, two-wire, half-duplex, differential drive bus, utilizing RS-485 transceivers. For more details on the ICMB connector specification and circuitry, refer to the following document:

IPMI-Intelligent Chassis Management Bus Bridge Specification v1.0, Document Revision 1.1.

8.12 Management UART Signal Level Translation

The motherboard must provide the necessary level translation circuitry to convert the management UART signals coming from the OPMA connector to RS-232 compliance. From the level translator, these signals are routed to the pack panel RS-232 connector that is shared between the management UART and the system UART.

8.13 Management UART Signal Multiplexing

The system serial port connector may contain either BMC console traffic (i.e., a Command Line Interface) or host system console traffic (i.e., BIOS setup screen). Separate serial port connectors are not allowed for management traffic and the local system UART traffic. When no mCard is installed, the system UART must be automatically routed to the system serial port connector.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

In order for this to work with the various types of BMCs found on mCards, serial port multiplexers are required on the motherboard. See Chapter 6 beginning on page 51 for more details. The motherboard's system UART TxD/RxD mux must route these signals to the mCard socket when MCARD_DETECT_L is active. When MCARD_DETECT_L is inactive, the TxD/RxD signals are routed to the system serial port connector. The motherboard's UART handshake signal mux must route the UART handshake signals from the mCard connector to the system serial port connector when MCARD_AUX_SOL_CTRL_L is active. When MCARD_AUX_SOL_CTRL_L is inactive, the system UART handshake signals are routed to the system serial port connector.

8.14 Support for Dedicated Management LAN

The mCards may use either a shared NIC configuration or contain a dedicated management NIC. The motherboard must route the mCard dedicated management NIC signals to a dedicated management network RJ-45 connector that is accessible by the end user. The designer must also provide means to block off this RJ-45 connector in the event that an mCard is installed that uses a shared NIC configuration.

8.15 MCard Presence Detect and Interface ID Support

The motherboard must support inputs with proper terminations through which the system BIOS can detect the mCard presence and decode IPMI interface type supported by the mCard. OPMA presents four signals: one for card presence detect (MCARD_DETECT_L) and three INTERFACE_ID signals (MCARD_ID0 – MCARD_ID2) for the interface type identifier. Refer to the appropriate sections of this specification for descriptions on these signals. See Table 5 on page 37 for appropriate termination of these signals on the motherboard.

8.16 Motherboard Support for I/O Functions

The OPMA interface dedicates a fixed number of I/O signals to discrete sensors monitored by the mCard. These I/O signals are defined in the OPMA interface signal definition section. If the motherboard requires additional discrete I/O sensors to be monitored for additional manageability functions then those signals should be routed through I^2C or SMBus-based GPIO expanders. If any of these additional signals require interrupt attention from the mCard (as opposed to routine sensor polling performed by the mCard) then an I^2C or SMBus-based GPIO expander with interrupting capability may be used to drive the OPMA SYS_SMBUS_IO_EXP_INTR_L signal.

Example:

If the motherboard requires mCard to monitor a power supply presence/absence condition, the OPMA connector does not have a dedicated pin for this sensor. Accordingly, the motherboard must provide this signal through an I^2C or SMBus-based GPIO expander (such as a PCF 8575 device). Since this sensor does not require an interrupt attention from the mCard, an 8575 device would suffice in this case. When defining manageability subsystem I/O requirements, the motherboard designer should carefully consider which I^2C or SMBus expander-based GPIO

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

signals may or may not require interrupting capabilities for manageability functions, and then select the proper type of GPIO expander accordingly.

8.17 Motherboard Support for mCard SCI Interrupt Signal

The motherboard must route the MCARD_SCI_INT_L signal from the OPMA connector to an input on the Southbridge that is capable of generating an SCI (System Control Interrupt) to the ACPI subsystem. The mCard firmware will activate this output based on specific conditions and policies that are defined for a particular system, and the system level response to this interrupt will be based on those preset policies. This interrupt is intended to allow the BMC to initiate communications with the ACPI software running on the host system.

A typical usage would be as a replacement for certain types of temperature polling by server software. Instead of requiring the system to poll the BMC for system temperatures, the BMC would interrupt the ACPI subsystem when certain threshold temperatures were reached. OSPM at the system level could then reduce the heat generation of the processor(s) and/or other BMC-monitored devices.

8.18 Motherboard I/O Terminations

All motherboard hardware I/Os that interface with the OPMA connector must have 3.3-V input buffers. Also, motherboard hardware requirements must be implemented with 3.3-V signal logic.

8.19 Motherboard Support for NC-SI

For motherboards using network controllers that support the NC-SI interface, refer to that controller's data sheet for proper terminations. If the NC-SI interface is used, the proper OPMA mCards supporting each such interface must be chosen. Refer to DMTF NC-SI specification for electrical characteristics and requirements for clock generation circuits and other data signals in this interface.

8.20 Motherboard Support for MCARD_I2C_CPU Bus

For motherboards supporting the I^2C/SMBus interface to the OPMA mCard from the host system's CPU, refer to the motherboard design guide for that host system CPU's termination and electrical characteristics requirements. The OPMA specification requires that the MCARD_I2C_CPU_SCL, MCARD_I2C_SDA and MCARD_I2C_CPU_ALERT_L from the OPMA connector must be utilized only to interface with the host system CPU's I^2C/SMBus.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Chapter 9 OPMA Host System BIOS Support

While OPMA attempts to minimize impacts on existing BIOS, any form of mCard cross-platform compatibility requires some degree of system BIOS support. The main reason for OPMA specific BIOS support is to enable simple plug and play capability. Under normal circumstances, the BIOS should not halt the boot process or require keystroke input on any sort of BMC related error; the absence of an mCard should never stop a server from booting up. Any reference in the following paragraphs to BIOS printing messages to the boot screen should be taken to mean that BIOS must write the indicated message to the text-based BIOS POST screen that is displayed during POST.

9.1 MCard Presence Detect

BIOS must be capable of identifying the presence/absence of an mCard. This is done by monitoring the MCARD_PRESENCE_L signal using a BIOS-controlled GPIO that is typically implemented using the system Southbridge chip. If the GPIO is not available directly from the chipset, the system must use an alternative solution to read this signal from the mCard (i.e., an SMBus based GPIO expander).

If an mCard is found to be present, the BIOS then determines which IPMI interface to use for sending IPMI commands. See Section 9.2 for details on command interface type detection. BIOS and other system software reads the card detect signal and then determine the mCard presence as defined in Table 9.

Table 9. Card Detect Signal States

Card Detect State	Meaning
Asserted (Low)	An mCard is installed in the OPMA connector.
Deasserted (High)	No mCard is installed in the OPMA connector.

9.2 MCard IPMI Command Interface Type Detection

The BIOS must be capable of determining which IPMI interface the mCard supports. This is done through the INTERFACE_ID signals also referred as MCARD_ID0 – MCARD_ID2 initialized by the mCard and monitored by the BIOS-controlled GPIOs that are typically implemented using the system Southbridge chip. If the GPIOs are not available directly from the chipset, the system must use an alternative solution to read the signals from the mCard (i.e., an SMBus-based GPIO expander).

The value read back from the INTERFACE_ID signals indicates the default IPMI communications interface supported by the management controller. In cases where the mCard supports more than one IPMI interface, these additional interfaces are conveyed through an IPMI

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

OEM command, which is detailed in Section 12.3 on page 91. BIOS or system software can decide at runtime which interface to use for IPMI communications. These signals are decoded in Table 10.

Table 10. Management Subsystem Host Interface Type Encoding

Interface Definition	Encoded Value
KCS Interface	000b
Block Transfer Interface	001b
SMIC Interface	010b
SMBus Interface (SSIF)	011b
Reserved	100b
Reserved	101b
mCard in upgrade kit mode	110b
Interface type not initialized	111b

To avoid race conditions associated with the configuration of the MCARD_ID bits, the BIOS should sample these bits for up to 25 seconds after having determined that an mCard is present by using the MCARD_DETECT_L signal. Most likely the BIOS will not have to wait this long during system boot up for the BMC to be ready, but some BMCs take a longer time to initialize. Any delays would only be noticed when AC power is applied to a system and the system is immediately configured to boot up without further intervention. No delay would be noticed by the BIOS during POST caused by system reset or by boot up from any ACPI state other than mechanical off. This is because the BMC is powered at all times except during the Mechanical Off state.

If the interface ID bits remain 111b for the full 25 seconds, BIOS should assume that a problem exists with the mCard and print this fact to the POST screen. BIOS must wait an additional 50 ms beyond the initial detection of any pattern other than 111b before taking a final MCARD_ID reading. This allows slower BMC microcontrollers enough time to set up these bits if the controls for them reside in multiple BMC registers.

9.3 IPMI Command Hardware Interface Support

The host system BIOS must contain basic IPMI command support for some or all supported IPMI 2.0 command interfaces: KCS, BT, SMIC, and SSIF. If the BIOS does not have IPMI command firmware support for the IPMI interface that the BMC indicates is available, the BIOS must print to it's POST screen that an mCard was detected and that it exposed the indicated IPMI command interface, but that BIOS does not support that IPMI command interface.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

9.4 IPMI Command BIOS Interface Support

The system BIOS must support all of the BIOS-Firmware interaction specific commands defined in Chapter 12 beginning on page 83.

9.5 MCard Presence, Health and BMC Firmware Revision Reporting

The system BIOS must print a POST screen message that indicates whether or not an mCard was detected as being present. If an mCard is present, the BIOS must determine if an IPMI command interface is present on the mCard that the BIOS know how to communicate with. If not, the BIOS must state this fact. If so, the BIOS must state which IPMI interface initial BMC communications will take place over. After BIOS-BMC communications have been initiated, the BIOS will execute several IPMI commands in order to synch up with the BMC. One of those commands will be GetDeviceID which will return the BMC firmware revision. BIOS must print this to the POST screen. BIOS will also issue the GetSelfTestResults command. BIOS will print the results of this command to the POST screen. The following are example BIOS POST screen displays:

For the case of no card installed:

No mCard detected in OPMA connector

A card is detected, uses the KCS interface, and is functioning properly:

MCard detected in OPMA connector

Initial IPMI communications to occur via KCS interface

BMC firmware revision: (put revision strong here)

BMC self test PASSED

A card is detected, uses the SMIC interface which BIOS does not support:

MCard detected in OPMA connector

Initial IPMI communications to occur via KCS interface (Interface detected is not supported by BIOS)

A card is detected, but the INTERFACE_ID bits always read back 111b:

MCard detected in OPMA connector

Timeout waiting for interface ID reporting

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

A card is detected, INTERFACE_ID indicates that it uses the KCS interface, but does not respond to IPMI commands on that interface:

MCard detected in OPMA connector

Initial IPMI communications to occur via KCS interface

BMC command timeout on KCS interface

A card is detected, and the BIOS sends the SetSystemTypeIdentifier command to the BMC. The BMC returns a completion code of C9h which indicates the BMC was built to a later version of the OPMA specification than the motherboard was. Since the mCard may not be sure that it will function with the older motherboard, it will therefore not scan the sensors:

MCard detected in OPMA connector

Initial IPMI communications to occur via KCS interface

The installed mCard has determined that it is not spec level compatible with the host system.

A card is detected, and the BIOS sends the SetSystemTypeIdentifier command to the BMC. The BMC returns a completion code of CCh which indicates the BMC does not know the ID of the motherboard, and will therefore not scan the sensors:

MCard detected in OPMA connector

Initial IPMI communications to occur via KCS interface

The installed mCard has not been ported to this system

9.6 System Identification

The system BIOS must contain a system identification OEM ID and management sensor implementation ID. The BIOS communicates this information to the BMC as part of the POST BIOS-BMC synchronization process. See Section 12.2 on page 86 for more information.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Chapter 10 OPMA Feature Card Considerations

The following sections provide details on items that OPMA feature card hardware and firmware designers should consider for achieving OPMA compatibility. This is not an exhaustive list of OPMA compatibility requirements, but rather an emphasis on a few key points that are targeted toward mCard designers.

10.1 Interface Identification

The INTERFACE_ID signals are typically implemented by GPIOs coming off of the BMC. After a reset of the manageability subsystem itself (typically accomplished after cold power up from the ACPI Mechanical Off state, by pressing a Reset button on the mCard, or by sending an IPMI command to reset the BMC itself), the INTERFACE_ID signals should be reset to the Not Initialized state of 111b. Since the motherboard terminates these signals with pullups, the mCard designer may connect these signals to BMC GPIOs that default to inputs or to tristated outputs after reset. Once the BMC is ready to start communicating with the host, it should configure the INTERFACE_ID signals according to Table 10 on page 74 so that communication can begin. These signals should only be configured after the BMC is actually ready to communicate since the read back of a value other than 111b means the interface is ready for the host to send commands to. The mCard must ensure that the transition of the INTERFACE_ID signals from 111b to any other pattern must take less than 50 ms to complete.

The use of BMC controlled GPIOs may be replaced by other logic on the mCard subsystem as long as all of the valid bit patterns of Table 10 on page 74 are automatically generated for the system to read and all timing constraints as listed above are met. Most importantly, the INTERFACE_ID bits may not change from the 111b pattern until the BMC is actually ready to accept commands on the indicated channel. There is no requirement for system BIOS or other software to perform command retries, etc., in order to synch up with firmware. The hardware facilities previously listed were defined to eliminate the difficulties and uncertainty associated with command retries.

10.2 CMOS Reset

The mCard controls the MCARD_CLR_CMOS_L signal which, when activated, clears the system CMOS. When commanded to clear the managed platform's CMOS by OPMA defined IPMI commands, the mCard must assert MCARD_CLR_CMOS_L for no less than 250 ms before deasserting it.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

10.3 Respect the Reserved Signals

OPMA feature cards must not use any of the connector signals that are marked for future OPMA specification expansion. These signals should all be No Connects on the card itself. Failure to do this may mean that the OPMA card is incompatible with motherboards that are built to later versions of this specification. The possibility of hardware damage to the mCard and to the motherboard cannot be ruled out if the mCard makes any sort of signal connection to OPMA signals marked Reserved.

10.4 Sideband Signals

A primary goal for the OPMA specification is to assist in the build up of an infrastructure ecosystem of management subsystem card designers and builders. Key to this goal is the cross compatibility of OPMA cards across motherboards. Sideband signals are not allowed in OPMA compliant systems since they are by definition proprietary.

10.5 Local Voltages

OPMA cards must generate required voltages locally if they are different from the 3.3-V and 5-V rails that are available over the OPMA connector.

10.6 Fan Tachometer Multiplexing

OPMA cards must support the OPMA 16:4 fan tach muxing scheme. If the mCard BMC does not have four native fan tach inputs, additional muxing must be added on the mCard itself to compensate.

10.7 Single Back Panel Serial Connector Support

RS-232-based text console management infrastructure is equipment and cabling intensive. New servers coming to market must integrate cleanly into the installed base of this type of management communications backbone. To do this, all text console traffic must exit the server at a single connector. The mCard must therefore support the multiplexing of CLI traffic and OS console traffic to a single back panel serial port connector. This is a firmware and potentially a hardware statement for mCard designers. A portion of this support must also be handled on the motherboard. See Chapter 6 beginning on page 51 for more information.

10.8 I^2C or SMBus Mux Addressing

MCards must respect OPMA rules for on-card SEEPROM and I^2C or SMBus mux addressing. See sections 3.3.1 and 3.3.2 on page 35 for more details.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Chapter 11 BMC Firmware OPMA Compatibility Requirements

Apart from the standard firmware configurations and feature sets supported by the BMC firmware infrastructure, the mCard firmware must implement certain capabilities in order to be OPMA compliant. The following sections highlight the specific features. For detailed description of each feature refer to the appropriate sections in the main portion of the OPMA specification.

11.1 KCS Interface

The mCard BMC devices supporting two or more KCS interfaces must enable at least two of the KCS ports for IPMI communications. The primary KCS must be enabled at the primary address as stated in the *IPMI Specification*.

11.2 MCard IPMI Interface ID GPIO Initialization

The mCard firmware must initialize the Interface ID GPIOs defined on the interface as quickly as it can after the BMC is powered up so that the system BIOS can identify the IPMI interface type supported by the mCard. If the mCard supports more than one type of IPMI system interface, the interface ID GPIOs should be initialized with the default bit pattern as defined in the OPMA specification.

11.3 MCard-Specific IPMI-OEM Command Support

The mCard firmware should support all the IPMI-OEM command extensions defined in Chapter 12 beginning on page 83. The IPMI-OEM extension commands defined for the mCard fall under the 3Eh, 3Fh NetFns. The mCard firmware should implement those commands using the specified NetFns.

11.4 System Identification and mCard Capabilities

The initial boot sequence IPMI interactions between the system BIOS and the mCard communicate the motherboard identification information to the mCard during system POST. This is accomplished using the SetSystemTypeIdentifier command. When the BMC obtains the host system OEM and subsystem implementation ID values, it must store this information in a nonvolatile location. If the BMC is reset independently of the host system, the BMC must detect this and reload the last known good host ID values from the nonvolatile storage after the BMC reset is completed. These parameters should be rewritten into the BMC nonvolatile store during the BIOS boot handshaking each time the motherboard is rebooted. The system ID feature requires that the system go through one successful boot sequence before actual deployment.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Each mCard make/model must maintain a list of the motherboard identification information so that it can detect when it's installed in a compatible motherboard. Achieving 100% compatibility

is unlikely with any given motherboard without some level of firmware and/or sensor porting to a given motherboard. This also implies compatibility testing by at least the mCard provider, and in many cases, will include the motherboard manufacturer.

The mCard firmware disables its sensor scanning functions if any invalid systems or un-supported systems are identified during these interactions. In addition, it should reject any commands that request sensor data. Returning a completion code of D5h to these commands rejects them as defined in the *IPMI Specification*.

If the mCard is installed as an upgrade kit, the SetSystemTypeIdentifier command is not sent to the mCard by the host system BIOS. This is because the upgrade kit's IPMI command interface is disabled by rule and communication with the BIOS is neither possible nor desired. System BIOS knows that the mCard is in upgrade kit mode by examining the INTERFACE_ID signals on the OPMA connector interface.

The mCard firmware provides some of the key capabilities that it supports through the Get mCard Capabilities IPMI-OEM command.

The mCard firmware provides a set of host IDs that it can support by default through the GetSupportedHostIDs IPMI-OEM command. By default the firmware should support at least one host ID.

11.5 Host System Compatibility Detection and Handling

The mCard is responsible for determining its compatibility with the motherboard in which it is installed. It's important to note that the motherboard system BIOS does not have any code that is specific to a particular make and model of mCard, while mCards may have code that is unique to a certain motherboard. As stated in Section 11.4, mCards must contain a list of motherboard IDs of compatible motherboards. In the event that an mCard is plugged into a system with an unlisted ID, the mCard may still decide that the motherboard is compatible. This compatibility decision model may be used by firmware implementations that are very generic and that mainly need sensor information to be compatible with a given motherboard. Such sensor information could be supplied in a variety of ways, including being downloaded out of band, so the specification leaves this point open for innovation on the part of the mCard designer.

If the mCard determines that it is not compatible with the motherboard, it must respond back to the SetSystemTypeIdentifier command with a completion code of CCh to indicate that the host system is unsupported.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

11.6 Upgrade Kit Support

The mCard firmware must provide IPMB channel support. The IPMB is the primary communication path between a down solution (motherboard mounted BMC) and an mCard in an upgrade mode. The mCard firmware in such configuration must disable its host IPMI interface to avoid any interference with the host interface and the down solution. The mCard firmware does a broadcast through the IPMB channel to determine any BMC device present on the IPMB if it detects another BMC device then the mCard changes its device ID to an address such as 28h (or 48h if 28h is already used) and perform the above mentioned functions.

In an upgrade mode configuration, the mCard obtains all the on-board sensor configurations, sensor data, etc., from the down solution BMC through standard IPMI commands as defined in the *IPMI Specification*.

11.7 Multiple Sensor Mapping Support

The goal behind supporting multiple sensor mappings is to utilize a single binary image of the mCard firmware to support more than one platform with very similar sensor mappings. The mCard firmware by default should provide a list of system identifiers that it can support and can queried through the Get Supported Host IDs IPMI-OEM command. By identifying what system the mCard is plugged into (through Set System Identifier command), the mCard firmware should be capable of switching the sensor configurations and SDRRs for a particular host. By default, the firmware supports at least one system configuration.

11.8 ACPI State Detection

The mCard hardware interface connector provides three inputs for the system to encode the ACPI states. The mCard firmware uses the three inputs to identify the current ACPI state of the system. The encoding scheme for these three inputs is defined in the OPMA specification.

11.9 Fan Monitoring

The mCard hardware interface currently supports up to 16 fans through a motherboard mounted 16:4 mux device. The mCard firmware must switch the mux device using the four mux control outputs. The mCard firmware must use a proper strategy so that the pulse accumulator or tachometer inputs on the BMC are provided with enough time to accumulate the pulses for accurate tachometer readings.

11.10 Fan Control

OPMA is designed to support PWM outputs for controlling three different groups of fans. The groups are identified as System, CPU and Power Supply fans. Each group of fans is controlled at the same speed level. OPMA proposes up to three speed levels of speed for each group. PWM-DC converter circuitry must be provided on the motherboard to control DC fans. If the

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

platform/motherboard is designed to use PWM controlled fans, the PWM-DC converter circuitry may not be required, and the PWM signals from the mCard must be properly buffered before routing to the fan circuitry.

11.11 Multi-Master I^2C or SMBus Support

The OPMA interface is designed to support multiple I^2C or SMBus buses. One of the I^2C or SMBus is designated as a shared I^2C or SMBus where the resources could be shared between multiple I^2C or SMBus controllers. The devices in that bus could be accessed by other I^2C or SMBus master controllers and since I^2C or SMBus inherently supports multi-master protocol the mCard firmware must be capable of handling multi-master scenarios.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Chapter 12 OPMA-Defined IPMI Command Extensions

While OPMA attempts to standardize the hardware interface of feature card-based management subsystems, maximum benefit from the adoption of such a standard is only gained if it sufficiently enables interoperability of mCards and motherboards. Achieving 100% compatibility between mCards and all mCard-aware systems is a good goal, but it is not a goal of this version of the specification. As a result, the detection and handling of incompatible scenarios becomes a requirement.

For example, if a M1 tier card with SMBus based sideband as described in OPMA specification rev 1.2 and below is plugged into a server that either has no SMBus based IPMI pass-through capabilities on its main system NIC or if it uses a different pass-through communications channel or protocol than that implemented on the system NIC, then this must be detected and handled. In this particular case there is no easy recovery from this type of scenario, so the best solution is simply to warn the user of the issue.

This and other issues such as a variable number of IPMI KCS system interfaces, support of different levels of the OPMA specification by either the system or by the mCard, etc. drive the need for some additional BIOS-BMC communications at boot time. This type of automated interoperability checking and handling requires the specification and adoption of some additional software interfaces.

OPMA acknowledges the widespread adoption of the IPMI on management controllers, and leverages the presence of this basic firmware infrastructure's in-built commands and command extension capabilities.

The mCard firmware must implement all features that are listed as mandatory in the *IPMI Specification* version 1.5. In addition, mCards must support the following set of OEM-specific IPMI commands to ensure cross platform compatibility, to benefit from additional capabilities provided by standard mCard hardware features, and to handle incompatibilities as cleanly as possible.

The OPMA-defined OEM IPMI commands fall under the OEM NetFn that is defined for the controller-specific OEM/Group definition in the *IPMI Specification*. Eight pairs of vendor-specific NetFns are defined under this group. The mCard uses the last pair with the values of 3Eh, 3Fh. All the OEM commands which are defined specifically by mCard fall under the 3Eh, 3Fh NetFns.

12.1 Set/Get Sensor Reading Offset Command

This is a Qlogic-defined IPMI OEM command that was defined primarily to provide a signed offset value to be used while monitoring certain sensors. Command definition is reprinted in Table 11 on page 84 by written permission of Qlogic.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Table 11. Set/GetSensorReadingOffset NetFn Codes

Command	NetFn (Request, Response)	CMD
SetSensorReadingOffset	30h, 31h	04h
GetSensorReadingOffset	30h, 31h	05h

12.1.1 Command—SetSensorReadingOffset

Table 12 provides the format for the SetSensorReadingOffset command.

Table 12. SetSensorReadingOffset Command Format

Data Type	Byte	Data Field
Request Data	1	Sensor LUN [7:2]—Reserved [1:0]—Sensor Owner LUN. LUN in the sensor owner used to send/receive IPMB messages to access the sensor. Use 00b if system software is the sensor owner.
	2	Sensor Number Unique number identifying the sensor behind a given slave address and LUN. Code FFh reserved.
	3	Offset Twos-complement, signed, 8-bit value
Response Data	1	Completion Code

Request Data

Sensor LUN—This is the LUN for the sensor being set up for a reading offset. The combination of sensor number and LUN uniquely identifies a sensor for a given sensor owner.

Sensor Number—This is the sensor number for the sensor that is being set up for a reading offset. The combination of sensor number and LUN uniquely identifies a sensor for a given sensor owner.

Offset—This twos-complement, 8-bit value is the offset that is added before the sensor reading is utilized for event generation and for IPMI commands such as GetSensorReading. Value = reading + offset. The offset value must be stored in nonvolatile memory. The *IPMI Specification* allows 1 a byte value for the value of the data read back from the sensor. The combined value data (reading + offset) cannot exceed one byte in size.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Response Data

Completion Code—This is the return code. Table 13 provides the valid return codes for this command.

Table 13. SetSensorReadingOffset Completion Codes

Completion Code	Description
00h	Successful
C2h	Command invalid for given LUN
CCh	Returned if any data field contains unsupported values such as sensor/LUN not being present.

12.1.2 Command—GetSensorReadingOffset

Table 14 provides the format for the GetSensorReadingOffset command.

Table 14. GetSensorReadingOffset Command Format

Data Type	Byte	Data Field
Request Data	1	Sensor LUN [7:2]—Reserved [1:0]—Sensor Owner LUN. LUN in the sensor owner used to send/receive IPMB messages to access the sensor. Use 00b if system software is the sensor owner.
	2	Sensor Number Unique number identifying the sensor behind a given slave address and LUN. Code FFh reserved.
Response Data	1	Completion Code
	2	Offset Twos-compliment, signed, 8-bit value.

Request Data

Sensor LUN—This is the LUN for the sensor that is being set up for a reading offset. The combination of sensor number and LUN uniquely identifies a sensor for a given sensor owner.

Sensor Number—This is the sensor number for the sensor that is being set up for a reading offset. The combination of sensor number and LUN uniquely identifies a sensor for a given sensor owner.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Response Data

Completion Code—This is the return code. Table 15 provides the valid return codes for this command:

Table 15. GetSensorReadingOffset Completion Codes

Completion Code	Description
00h	Successful
C2h	Command invalid for given LUN
CCh	Returned if any data field contains unsupported values such as sensor/LUN not being present.

Offset—This twos-complement, 8-bit value is the offset that is added before the sensor reading is utilized for event generation and for IPMI commands such as Get Sensor Reading. Value = reading + offset.

12.2 Set/Get System Type Identifier

The following paragraphs describe why the mCard must provide system ID information to the BMC during POST and describe the SetSystemTypeIdentifier and GetSystemTypeIdentifier commands.

12.2.1 System ID

The mCard firmware can be very system specific. Currently, even minor variations in the sensor arrays between very similar managed platforms can require new firmware components to be installed. OPMA seeks to reduce the number of individual firmware builds required for a given mCard design. Of course, various methods could be used to automate this process, but the goal is to limit the scope of BMC firmware changes required to implement this capability.

The mCards need the ability to support cross platform operation. A mCard must be able to identify what sensor configuration its host system has implemented before it starts scanning the sensors. Scanning an unknown sensor set can result in invalid readings and false error reports or other issues. For example, certain BMC implementations may enforce policies to reboot or shutdown systems that are reporting readings in the warning or critical ranges.

In order to enable the BMC to make the above determinations automatically, the system BIOS must provide system identification information to the BMC during POST. This is done using the SetSystemTypeIdentifier command as described in the following paragraphs.

The system identifier information is composed of two 16-bit values. The first is the OPMA system OEM ID. If the OEM/ODM is a member of the PCI SIG, use the PCI SIG-assigned

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

vendor ID for this field. If the OEM/ODM is not a member of the PCI SIG, contact OPMA specification maintenance personnel to be assigned a unique OPMA system OEM ID. The second 16-bit value is the host system manageability subsystem sensor array Implementation ID. Each system motherboard that uses a manageability sensor array which is unique from a BMC firmware support standpoint must be assigned a unique Implementation ID by the motherboard designer.

This number can be different for every motherboard design from a given OEM/ODM, or it can cover several server motherboards as long as, for a given mCard hardware design, the same exact BMC firmware binary will run unmodified on all motherboards that share both an OEM ID and an Implementation ID. Only server motherboard designers need to obtain mCard system OEM IDs and to generate Implementation IDs. The mCard subsystem providers must be made aware of these assignments by the host system providers. OPMA does not specify where the BIOS must store the aforementioned ID information. Table 16 lists the NetFn codes for the Set/GetSystemTypeIdentifier commands.

Table 16. Set/GetSystemTypeIdentifier NetFn Codes

Command	NetFn (Request, Response)	CMD
SetSystemTypeIdentifier	3Eh, 3Fh	A0h
GetSystemTypeIdentifier	3Eh, 3Fh	A1h

12.2.2 Command—SetSystemTypeIdentifier (cmd A0h)

The BIOS uses this command to communicate the host system ID information to the mCard at boot time.

Once the host has communicated its host system ID information to the BMC using this command, a remote operator may access this data using the GetSystemTypeIdentifier command. In this way, the remote operator can determine what system models his IT infrastructure contains and what OPMA compatibility level his server motherboards support. Table 17 on page 88 provides the format for the SetSystemTypeIdentifier.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Table 17. SetSystemTypeIdentifier Command Format

Data Type	Byte	Data Field
Request Data	1–2	OEM ID (treat as a word) PCI SIG vendor ID for motherboard manufacturer. Byte 1 – Least significant byte of the PCI SIG vendor ID Byte 2 – Most significant byte of the PCI SIG vendor ID
	3–4	Implementation ID (treat as a word) Binary encoded sensor subsystem identifier. The system manufacturer supplies a unique value for this field for each manageability subsystem environment in his product suite. Byte 3 – Least significant byte of the manufacturer supplied unique value Byte 4 – Most significant byte of the manufacturer supplied unique value
	5–6	OPMA Specification Compliance (treat as a word) BCD encoded. Byte 5 [7:0] – Most significant two digits of the revision Byte 6 [16:8] – Least significant two digits of the revision 0000h = reserved.
Response Data	1	Completion Code

Request Data

OEM ID—The mCard system vendor ID information. This is the vendor ID as assigned by the PCI SIG to particular vendor (refer to the PCI SIG for details). If the system builder does not have an assigned PCI vendor ID, then OPMA specification maintenance personnel will assign an mCard OEM ID to be used in this field. The mCards use this in conjunction with the Implementation ID field to determine OPMA compatibility with the host motherboard.

Implementation ID—A value that is assigned by its designer to each motherboard. If no existing BMC firmware binary will run unaltered on a new motherboard design, the designer must create a new Implementation ID number and assign it to that motherboard. The firmware can then be altered to support the new configuration. The new board's ID information should be added to the firmware's motherboard recognition database at that time.

Motherboard OPMA Specification Compliance—This field specifies which version of the OPMA specification the motherboard complies with.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Response Data

Completion Code—The valid completion codes returned for this command are taken from the *IPMI Specification*. Table 18 provides the completion codes for this command.

Table 18. SetSystemTypeIdentifier Completion Codes

Completion Code	mCard Meaning
00h	Successful
CCh	Invalid data field in the request (unsupported host system)
C9h	Parameter out of range (system OPMA compliance is down level with respect to mCard)

12.2.3 Command—GetSystemTypeIdentifier (cmd A1h)

This command is used to get the host system ID information and host's OPMA specification support version that was communicated to the mCard by the SetSystemTypeIdentifier command. Table 19 lists the format for the GetSystemTypeIdentifier command.

Table 19. GetSystemTypeIdentifier Command Format

Data Type	Byte	Data Field
Request Data		None
Response Data	1	Completion Code
	2-3	OEM ID (treat as a word) PCI SIG vendor ID for motherboard manufacturer. Byte 2 – Least significant byte of the PCI SIG vendor ID Byte 3 – Most significant byte of the PCI SIG vendor ID
	4-5	Implementation ID (treat as a word) Binary encoded sensor subsystem identifier. The system manufacturer supplies a unique value for this field for each manageability subsystem environment in his product suite. Byte 4 – Least significant byte of the manufacturer supplied unique value Byte 5 – Most significant byte of the manufacturer supplied unique value

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Data Type	Byte	Data Field
Response Data (cont.)	6-7	Host System OPMA Compliance (treat as a word) BCD encoded. Byte 6 [7:0] – Most significant two digits of the revision Byte 7 [16:8] – Least significant two digits of the revision 0000h = reserved.

Response Data

Completion Code—This is the return code. The valid completion codes returned for this command are within the definition of the *IPMI Specification*. Table 20 lists the completion code for the GetSystemTypeIdentifier command.

Table 20. GetSystemTypeIdentifier Completion Codes

Completion Code	Description
00h	Successful
Note: *Other IPMI-defined result codes are also valid for this command. See the* IPMI Specification *version 1.5 for meanings of any non-zero result codes returned from this command. The OPMA specification does not re-list this information for every command in the interest of brevity.*	

OEM ID—This is the value programmed by the SetSystemTypeIdentifier command. If the GetSystemTypeIdentifier command is executed before SetSystemTypeIdentifier has been run, then the value of 0000h is returned in this field.

Implementation ID—This is the value programmed by the SetSystemTypeIdentifier command. If the GetSystemTypeIdentifier command is executed before the SetSystemTypeIdentifier has been run, then the value of 0000h is returned in this field.

Host System OPMA Compliance—This is the value programmed by the SetSystemTypeIdentifier command. If the GetSystemTypeIdentifier command is executed before SetSystemTypeIdentifier has been run, then the value of 0000h is returned in this field. If the previous "Set System Type Identifier" command specified a value of 0140h to indicate that, for example, compliance with OPMA specification version 1.4 is the minimum allowable, then a value of 0140h will be returned in this field.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

12.3 MCard Capabilities Identifier

This is an OEM IPMI command defined for the system software to identify the type of mCard currently present in the system for any interaction specific decisions it has to make. Table 21 lists the NetFn codes for the GetmCardCapabilities command.

Table 21. GetmCardCapabilities NetFn Code

Command	NetFn (Request, Response)	CMD
GetmCardCapabilities	3Eh, 3Fh	A2h

12.3.1 Command—GetmCardCapabilities

This command provides information about the mCard in the system. Table 22 provides the format for the GetmCardCapabilities command.

Table 22. GetmCardCapabilities Command Format

Data Type	Byte	Data Field
Request Data		None
Response Data	1	Completion Code
	2–3	mCard Subsystem Specification Compliance BCD encoded. 0000h = reserved. [15:8]—Least significant digits of the revision [7:0]—Most significant digits of the revision Compliance with OPMA specification version 1.0 would return a value of 0100h in this field.
	4	mCard Out-of-Band LAN Support Type Binary encoded 01h = M1 class: with SMBus IPMI pass through type support 02h = M2 class: with dedicated NIC 03h = M3 class: with dedicated NIC plus KVMoIP etc., support 04h = M1 class: with NC-SI 05h = M2 class: with NC-SI 06h = M3 class: with NC-SI plus KVMoIP etc., support
	5	mCard Out-of-Band LAN Support Status 01h = Out-of-band access enabled 02h = Out-of-band access disabled

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Data Type	Byte	Data Field
	6	mCard Terminal Mode (CLI) Support Status 01h = Terminal mode support enabled 02h = Terminal mode support disabled
Response Data (cont.)	7	mCard IPMI Interface Supported Bit mapped If bit is set then interface is supported. Otherwise, the interface is not supported: [0]—KCS 0 [1]—KCS 1 [2]—BT [3]—SSIF [4]—SMIC [5]—Reserved [6]—Reserved [7]—Reserved
	8	Sensor Scanning Status Binary encoded 01h = Sensor scanning active 02h = Sensor scanning inactive
	9	Virtual Comm 1 Port Binary encoded 00h = Not supported 01h = Supported
	10	Virtual Floppy Interface Binary encoded 00h = Not supported 01h = Supported

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Response Data

Completion Code—This is the return code. The valid completion codes returned for this command are within the definition of the *IPMI Specification.* Table 23 lists the completion codes for this command.

Table 23. GetmCardCapabilities Completion Codes

Completion Code	Description
00h	Successful

mCard Subsystem Specification Compliance—This field identifies the OPMA specification version number to which this mCard is compliant.

mCard OOB LAN Support Type—This field contains a code which indicates the type of out-of-band LAN support the currently installed mCard supports. In cases where a mCard has the hardware capability to support multiple OOB LAN interfaces, at a given time only one of the interfaces have to be enabled. The mCard firmware after determining the interface to use for OOB LAN support must set this field to indicate the active OOB LAN interface. For example, if a M3 card has hardware capabilities to support both dedicated NIC and NC-SI, the firmware after identifying which one of the interface to use, must set this field to the appropriate value (see Table 22 on page 91) indicating either dedicated NIC or NC-SI.

mCard Out of Band Access Support Status—This field provides the enable status of the OOB LAN.

mCard Terminal Mode Support Status—This field provides information on the capabilities of the mCard supported UART (provides information whether CLI type interface is supported or not).

mCard IPMI Interface Support—This field specifies the IPMI interfaces supported by the mCard. For example, an mCard might support more than one IPMI-defined communications interface like KCS and BT. If a bit is set in this field, the corresponding IPMI interface is supported. The default interface is indicated through the INTERFACE_ID bits defined in Section 3.2.12 on page 28.

Sensor Scanning Status—This field indicates the sensor scanning status of the mCard. Under most circumstances, the mCard scans the sensors, and this field returns 01h. In some cases (such as when the mCard is used as an upgrade kit, or when the mCard detects that it's installed in an unsupported host system, or when mCard detects that it is down level with respect to the host motherboard), sensor scanning is disabled, so this field returns 02h.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Virtual Comm 1 Port—This field indicates whether or not the installed mCard contains a BMC that presents a legacy PC compatible (8250, 16450, 16550) UART on the host interface bus (i.e., LPC). If such communications port registers are present, they must be mapped to the legacy "comm. 1" address range of 3F8h–3FFh. In this case, the system BIOS should disable or relocate any RS-232 compatible UART on the motherboard that defaults to these addresses. If the mCard does not support a virtual UART, the host system UART is used in conjunction with BMC UART(s) to support all required text console (BMC CLI, BIOS setup, etc.) that are required. See Chapter 6 beginning on page 51 for more details on the serial port support scheme.

Virtual Floppy Interface—This field indicates whether or not the installed mCard contains a BMC that presents a PC-compatible (765, 82077, or 8477 FDC device) floppy disk controller interface (FDC) on the host interface bus (i.e., LPC). If such an interface is present, the system BIOS setup options must provide the capabilities to enable this interface as a boot device.

12.4 MCard Clear CMOS

This OEM command is used by an external application for clearing the CMOS. The mCard hardware provides support through its interface for clearing the CMOS. Table 24 lists the NetFn codes for the ClearCMOS command.

Table 24. ClearCMOS NetFn Codes

Command	NetFn (Request, Response)	CMD
ClearCMOS	3Eh, 3Fh	A3h

12.4.1 Command—ClearCMOS (cmd A3h)

Commands the mCard to clear the host system CMOS RAM. Table 25 provides the format for the ClearCMOS command.

Table 25. ClearCMOS Control Command Format

Data Type	Byte	Data Field
Request Data		None
Response Data	1	Completion Code

Request Data

No request data is associated with this command. When the command is received, the BMC should assert the MCARD_CLR_CMOS_L signal for a minimum of 250 ms and then deassert it.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Response Data

Completion Code—This is the return code. The valid completion codes returned for this command are within the definition of the *IPMI Specification*. Table 26 lists the completion code for this command.

Table 26. ClearCMOS Completion Code

Completion Code	Description
00h	Successful

12.5 MCard Local Lock Out

This OEM command is used by application software communicating with the mCard to either enable or to lock out local access to the system. The system motherboard must provide sufficient hardware circuitry to support this feature. Table 27 lists the NetFn codes for the Set/GetLocalAccessLockOutState commands.

Table 27. Set/GetLocalAccessLockOutState NetFn Code

Command	NetFn (Request, Response)	CMD
SetLocalAccessLockOutState	3Eh, 3Fh	A4h
GetLocalAccessLockOutState	3Eh, 3Fh	A5h

12.5.1 Command—SetLocalLockOutState (cmd A4h)

This command is used for activating the local access lock out function. While the support for this command at the motherboard hardware level is optional, BMC firmware must implement this command. The mCard knows the capabilities of the motherboard by examining the OPMA vendor and system ID information available from the motherboard. If the motherboard does not support this command, this command should return a completion code of C1h to indicate this to the command initiator. Table 28 provides the format for the SetLocalLockOutState command.

Table 28. SetLocalLockOutState Command Format

Data Type	Byte	Data Field
Request Data	1	Control 00h = Lock the local access 01h = Unlock the local access
Response Data	1	Completion Code

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Request Data

Control—The system will use this field to either activate (lock) or de-activate (unlock) the mCard local access lock out control signal.

Response Data

Completion Code—This is the return code. The valid completion codes returned for this command are within the definition of the *IPMI Specification*. Table 29 provides the completion codes for this command.

Table 29. SetLocalLockOutState Completion Codes

Completion Code	Description
00h	Successful
C1h	Motherboard does not support lock out

12.5.2 Command—GetLocalLockOutState (cmd A5h)

The mCard returns the status of local access based on the SetLocalLockOutState command that was issued before. The default is unlocked. Table 30 provides the format for the GetLocalLockOutState command.

Table 30. GetLocalLockOutState Command Format

Data Type	Byte	Data Field
Request Data		None
Response Data	1	Completion Code
	2	Lockout Control Output State 00h = Locked 01h = Unlocked Default = 1 (Unlocked), and exists if SetLocalLockOutState command was not issued since the last time the mCard came out of reset or was power up.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Response Data

Completion Code—This is the return code. The valid completion codes returned for this command are within the definition of the *IPMI Specification.* Table 31 lists the completion codes for this command.

Table 31. GetLocalLockOutState Completion Codes

Completion Code	Description
00h	Successful
C1h	Motherboard does not support lock out.

Lock Out Control Output State—Provides the state of the local lock out signal. This status is provided by keeping track of the control action specified in the previous SetLocalLockOutState command that was received. The default power up status is unlocked.

12.6 Get Supported Host System IDs

This OEM command is used to ascertain which host motherboards the firmware on a particular mCard supports. It returns a fixed field size. Any unused fields are returned with 00h. If an mCard indicates that it does not recognize a given motherboard that it's plugged into, a remote operator can use this command to determine which motherboards the mCard does support. In addition, it can use the GetSystemTypeIdentifier command to determine the ID of the motherboard that the mCard is plugged into. Table 32 lists the NetFn codes for the GetSupportedHostIDs command.

Table 32. GetSupportedHostIDs NetFn Codes

Command	NetFn (Request, Response)	CMD
GetSupportedHostIDs	3Eh, 3Fh	A6h

12.6.1 Command—GetSupportedHostIDs (cmd A6h)

This command returns the complete list of OEM systems that is supported by the firmware running on the mCard BMC. This command must return at least one valid OEMID/Implementation ID pair.

Table 33 on page 98 provides the format for the GetSupportedHostIDs command.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Table 33. GetSupportedHostIDs Command Format

Data Type	Byte	Data Field
Request Data	1	None
Response Data	1	Completion Code
	2	Supported Host System ID Count The number of unique host system configurations that this firmware implementation supports. Each host ID consists of 4 bytes. The maximum number of supported systems by a single firmware implementation is 7.
	3–4	OEM ID 0 (treat as a word) PCI SIG vendor ID for motherboard manufacturer. The mCard must return a valid vendor ID in this field.
	5–6	Implementation ID 0 (treat as a word) Binary encoded sensor subsystem identifier. The system manufacturer supplies a unique value for this field for each manageability subsystem environment in his product suite. The mCards must return a valid Implementation ID in this field.
	7–10	ID Pair 1 This and all following ID pairs use the same byte format and ordering as for bytes 3-4 and 5-6. This and all subsequent supported host ID pair fields are optional The total number of ID pairs provided must match the value given in the Supported Host System ID Count field.
	11–14	ID Pair 2
	15–18	ID Pair 3
	19–22	ID Pair 4
	23–26	ID Pair 5
	27–30	ID Pair 6

Request Data

None**.**

Response Data

Completion Code—This is the return code. The valid completion codes returned for this command are within the definition of the IPMI Specification. Table 34 on page 99 lists the completion codes for this command.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Table 34. GetSupportedHostIDs Completion Codes

Completion Code	Description
00h	Successful

Supported Host System ID Count—The number of host systems supported by this combination of mCard firmware and mCard hardware. The maximum number of unique systems that any firmware binary can support is seven. This is mainly driven by an IPMI response length limitation of 32 bytes.

OEM ID 0—This is the first half of the first pair of host system OEM identifiers. A valid OEM ID (i.e., PCI SIG OEM ID) must always be returned in this field.

Implementation ID 0—This is the second half of the first pair of host system OEM identifiers. A valid system specific implementation ID must always be returned in this field. Implementation IDs may be obtained from OEMs and ODMs that develop OPMA compatible motherboards.

ID Pair 1–6—These are the remaining, optional six system identification pairs. Any of these fields that do not contain valid OEM identification information is to be returned containing the value 0000_0000h. Any provided IDs must be returned contiguously. In other words, the first motherboard ID info must go in ID pair 0, the second in ID pair 1, etc. The number of IDs returned must match the number returned in the Supported Host System ID Count Field.

12.7 Required Support of Normally Optional IPMI Commands

The following commands are fully defined by the *IPMI Specification*, but are listed as optional in that specification. OPMA requires them to be implemented in order to better enable cross platform compatibility. For request and response field information, refer to the *IPMI Specification*, version 1.5.

12.7.1 Set/GetSensorThreshold Commands

OPMA requires the threshold values for certain temperature sensors to be set up inside the BMC during the BIOS boot process for certain system implementations. The BMC firmware must support this IPMI command.

12.7.2 GetSensorReadingFactors Command

The mCard must support this IPMI command to fully support sensors using special scaling factors. The BMC firmware must support this IPMI command.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Appendix A Boot Sequence Theory of Operation

This section covers the sequence of events that are designed to occur with respect to the management subsystem for the various mCard scenarios.

No down solution; mCard not installed:

In this scenario, the only management capability is via the OPMA connector (no BMC soldered to motherboard).

- No mCard is installed (no management solution is in the system).
- Power up system.
- The mCard-aware BIOS gets control, samples MCARD_DETECT_L high (inactive), and determines mCard is not present. As a result, BIOS does not read or interpret INTERFACE_ID signals MCARD_ID0 – MCARD_ID2.
- Policy for handling no mCard present is OEM specific. OPMA does not cover this scenario in any greater detail than to specify that lack of an mCard installed in the connector must never stop the system from powering up or from BIOS getting control and starting the execution of instructions.

No down solution; mCard is installed:

In this scenario, the only management capability is via the OPMA connector (no BMC soldered to motherboard).

- An mCard is installed.
- Power up system.
- The mCard-aware BIOS gets control, samples MCARD_DETECT_L low (active), and determines mCard is present.
- BIOS samples INTERFACE_ID signals and determines which IPMI host interface to use for mCard BMC communications.
- BIOS performs OPMA-defined BMC start up handshaking.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Down solution installed; mCard upgrade kit not installed:

In this scenario, a down solution exists. In addition, an OPMA connector has also been added to the system for upgrade purposes. Note that in upgrade-capable boards, there is no requirement for the system BIOS to be mCard aware. This is because the mCard will never enable its host interface in an upgrade scenario. This Zero Impact design feature of mCard means that an OPMA connector can be easily added to existing designs without making BIOS or firmware changes. If Zero Impact is used, then BIOS would simply ignore mCard completely and would never know it's there. However, nothing stops the BIOS vendor from being mCard aware in this situation, and the following steps list the sequence for an mCard aware BIOS:

- No mCard board is installed in the OPMA connector.
- Power up system.
- Down solution boots normally, starts sensor scan.
- System BIOS starts to boot.
- The mCard-aware BIOS gets control, samples MCARD_DETECT_L high (inactive), and determines that no mCard is present. BIOS can make note of this fact for display on the setup screen.
- BIOS then performs normal IPMI communications with the down solution (if any such communications are indeed required).

Down solution installed; mCard upgrade kit is installed:

In this scenario, a down solution exists. In addition, an OPMA connector has also been added to the system for upgrade purposes. Note that since a down solution is present, the Zero Impact feature of mCard does not require any BIOS awareness of mCard features. However, an mCard aware BIOS is still possible in this scenario. The following steps assume mCard awareness by the BIOS. In this case, the only value that the BIOS gets from mCard awareness is to know that an mCard is installed and that it has configured itself to upgrade mode. If BIOS knows that the motherboard has a down solution, but then subsequently detects an installed mCard which is not indicating that it's in upgrade kit mode, BIOS can flag an error on the boot screen.

- An mCard is installed.
- Power up system.
- The mCard and Down solution boot up.
- Down solution begins sensor scan. Local system software talks only to the down solution same as it did before upgrade kit was installed.
- The mCard detects down solution present, and thus does not enable its IPMI-defined host interface. In addition, mCard sets INTERFACE_ID bits to indicate that it's acting as an upgrade kit.
- The mCard BMC sets itself up to use an IPMI address of 28h (or 48h if 28h is already used).

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

- The mCard is able to process any out of band commands for sensor readings that are sent to IPMI device 28h (or 48h), but it gets all sensor data from down solution via IPMB.
- The mCard-aware BIOS gets control, samples MCARD_DETECT_L low (active), and then looks at INTERFACE_ID signals and determines mCard is in upgrade kit mode. Thus BIOS knows that mCard BMC has its host interface disabled and will not accept IPMI commands. BIOS and system software (i.e., IPMI driver) talk to down solution as they did before upgrade kit was installed.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Appendix B IPMI OEM Commands Summary

Table 35 lists the IPMI OEM commands defined in this specification.

Table 35. IPMI OEM Commands

Commands	NetFn	Command
SetSensorReadingOffset[1]	OEM (30h, 31h)	04h
GetSensorReadingOffset[2]	OEM (30h, 31h)	05h
SetSystemTypeIdentifier	OEM (3Eh, 3Fh)	A0h
GetSystemTypeIdentifier	OEM (3Eh, 3Fh)	A1h
GetmCardCapabilities	OEM (3Eh, 3Fh)	A2h
ClearCMOS	OEM (3Eh, 3Fh)	A3h
SetLocalAccessLockOutState	OEM (3Eh, 3Fh)	A4h
GetLocalAccessLockOutState	OEM (3Eh, 3Fh)	A5h
GetSupportedHostIDs	OEM (3Eh, 3Fh)	A6h

Notes:

1. *The NetFn for this OEM command was defined outside of the OPMA specification.*
2. *The NetFn for this OEM command was defined outside of the OPMA specification.*

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."

Appendix C Specification Modification Roadmap

The 1.0 version of the OPMA specification was designed to be a starting point for the creation of a COTS infrastructure for manageability subsystems. However, it was also understood that the initial specification had to be fairly lightweight in terms of the amount and scope of changes in hardware, BIOS firmware, and BMC firmware required to support it. As a result, several features were held back. A few of the more important features that should be considered for the next specification revision are listed in this appendix. No order of importance is implied.

1. Motherboard – based sensor array descriptors. The way OPMA is currently defined, the BMC must be ported to every system that it is to be run on. While this is in line with the way things are done in state of the art servers today, it is a serious impediment to the creation of a COTS subsystem. Optimally, the BMC would not have to have direct knowledge of system specific sensor configurations. Instead, it would import this information from the system motherboard using some standardized, OPMA-defined method. Whatever method is chosen, it should not involve the system BIOS since BIOS is already complex enough and is difficult to change, and finding available space on server system BIOS ROMs is always an issue. One potential solution is to create a sensor definition descriptor language and to use it to build sensor descriptors that would go in a SEEPROM mounted on the motherboard. The BMC would read that SEEPROM and use it to build a local Sensor Data Record (SDR). An IPMI compliant SDR SEEPROM could be placed on the motherboard, but does not contain enough information to completely describe the sensor set. The language-based approach also opens the door for saving other motherboard specific information on the motherboard itself, thus enhancing the potential for true plug and play operation. Of course, the language would compile into a binary that would go onto the SEEPROM.

© 2004–2008 Advanced Micro Devices, Inc. All rights reserved.
By using or providing feedback on this Specification, you agree to the attached license agreement entitled "OPMA SPECIFICATION LICENSE AGREEMENT."