

OCP



International Partnership

Open Core Protocol Specification

Release 3.0 Errata
OCP-IP Confidential



Open Core Protocol Specification

Errata for Release 3.0

Document Revision - 0.6

© 2000-2010 OCP-IP Association, All Rights Reserved.

Open Core Protocol Specification 3.0 Errata
Document Revision - 0.6

This document, including all software described in it, is furnished under the terms of the Open Core Protocol Specification License Agreement (the "License") and may only be used or copied in accordance with the terms of the License. The information in this document is a work in progress, jointly developed by the members of OCP-IP Association ("OCP-IP") and is furnished for informational use only.

The technology disclosed herein may be protected by one or more patents, copyrights, trademarks and/or trade secrets owned by or licensed to OCP-IP. OCP-IP reserves all rights with respect to such technology and related materials. Any use of the protected technology and related material beyond the terms of the License without the prior written consent of OCP-IP is prohibited.

This document contains material that is confidential to OCP-IP and its members and licensors. The user should assume that all materials contained and/or referenced in this document are confidential and proprietary unless otherwise indicated or apparent from the nature of such materials (for example, references to publicly available forms or documents). Disclosure or use of this document or any material contained herein, other than as expressly permitted, is prohibited without the prior written consent of OCP-IP or such other party that may grant permission to use its proprietary material.

The trademarks, logos, and service marks displayed in this document are the registered and unregistered trademarks of OCP-IP, its members and its licensors. The following trademarks of Sonics, Inc. have been licensed to OCP-IP: FastForward, CoreCreator, SiliconBackplane, SiliconBackplane Agent, InitiatorAgent Module, TargetAgent Module, ServiceAgent Module, SOCCreator, and Open Core Protocol.

The copyright and trademarks owned by OCP-IP, whether registered or unregistered, may not be used in connection with any product or service that is not owned, approved or distributed by OCP-IP, and may not be used in any manner that is likely to cause customer confusion or that disparages OCP-IP. Nothing contained in this document should be construed as granting by implication, estoppel, or otherwise, any license or right to use any copyright without the express written consent of OCP-IP, its licensors or a third party owner of any such trademark.

Printed in the United States of America.

EXCEPT AS OTHERWISE EXPRESSLY PROVIDED, THE OPEN CORE PROTOCOL (OCP) SPECIFICATION IS PROVIDED BY OCP-IP TO MEMBERS "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.

OCP-IP SHALL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES OF ANY KIND OR NATURE WHATSOEVER (INCLUDING, WITHOUT LIMITATION, ANY DAMAGES ARISING FROM LOSS OF USE OR LOST BUSINESS, REVENUE, PROFITS, DATA OR GOODWILL) ARISING IN CONNECTION WITH ANY INFRINGEMENT CLAIMS BY THIRD PARTIES OR THE SPECIFICATION, WHETHER IN AN ACTION IN CONTRACT, TORT, STRICT LIABILITY, NEGLIGENCE, OR ANY OTHER THEORY, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Errata

Type	Spec. Page #	Title	Errata Revision	Errata Page #
Specification Clarifications	57	Tag IDs	0.5	4
	52	Burst Definition	0.5	5
	55	Single Request, Multiple Data Bursts	0.5	6
Clarifications To Developers Guidelines	169-170	Incorrect Signal Names in Response Accept Sequence	0.5	7
Corrections to Compliance Rules	389	Modified Compliance Rule 1.2.29	0.5	8
	390	Deprecated Compliance Rule 1.2.30	0.5	9
	411	Duplicate Compliance Rule 1.7.2	0.6	10
Clarifications to Compliance Checks	172-173	Incrementing Imprecise Read Requires Deassertion of MBurstPrecise	0.5	11
	216	Incorrect Signal Names in Response Phase Guidelines	0.5	12
	232-233	Tag Interleaving for Packing Bursts	0.5	13
	459	STagInOrder Definition	0.5	14
	459	Connection Signals Definition	0.5	15

Introduction

Purpose and Scope

This document lists errata for the *OCP Protocol Specification, Release 3.0* (OCP3). It is intended for users of the OCP3 specification.

Errata Format

Each errata is presented in two sections:

- *Location* indicates where the original material was located in the OCP3 specification.
- *Modified Text* presents the new material. The original material is shown in strikethrough; the new material is shown underlined.

The original document formatting has been preserved, as far as possible. However, the reader should be aware that the table and figure numbers shown in the errata are not guaranteed to match those in the *OCP Protocol Specification, Release 3.0*, particularly where new material has been introduced. Instead, the table numbers should be used as a guide—references to tables and figures are consistent within the body of each specific errata.

Typographic Conventions

The following typographic conventions are used in this document:

strikethrough	Deleted text, i.e., the original text from OCP2 that has been superseded by material from OCP3.
<u>underlined</u>	New material from OCP3 (or OCP3 errata) which replaces material originally in OCP2 (or OCP3, as applicable).
monospaced	Code examples, or an OCP parameter name, e.g., datahandshake.

Definitions

The following terms are used throughout this document:

OCP2	<i>Open Core Protocol Specification, Release 2.2.1</i>
OCP3	<i>Open Core Protocol Specification, Release 3.0</i>

Errata Contributors

Our thanks to the following people who contributed errata:

Contributor	Errata Title
M.P. Rashid, Cadence Design Systems, Inc.	“Duplicate Compliance Rule 1.7.2,” on page 10
Neale Foulds, Duolog Technologies Ltd	“StagInOrder Definition,” on page 14
John Ivie, OCP-IP	Parts of “Deprecated Compliance Rule 1.2.30,” on page 9, “Incrementing Imprecise Read Requires Deassertion of MBurstPrecise,” on page 11, and others
Anita Vandanapu & Luc Ton, Sonics, Inc.	“Modified Compliance Rule 1.2.29,” on page 8

Contributor	Errata Title
Drew Wingard, Sonics, Inc.	"Deprecated Compliance Rule 1.2.30," on page 9
Ashot Abajyan, Sonics, Inc.	"Connection Signals Definition," on page 15
Harutyun Aslanyan & D. N. (Jay) Jayasimha, Sonics, Inc.	"Tag IDs," on page 4
James Aldis, Texas Instruments Incorporated	"Burst Definition," on page 5

Revision History

Revision	Notes
0.1-0.4	OCP-IP internal errata revisions.
0.5	Simplified errata structure; removed TOC and added cross-reference table. First publication to OCP-IP members.
0.6	Added Errata Revision to main errata cross-reference table. Added "Duplicate Compliance Rule 1.7.2" errata.

Specification Clarifications

Tag IDs

Location in OCP3 Specification

In the introduction of Section 4.7, on page 57.

Modified Text

Tags allow out-of-order return of responses and out-of-order commit of write data.

A master drives a tag on MTagID during the request phase. The value of the tag is determined by the master and may or may not convey meaning beyond ordering to the slave. For write transactions with data handshake enabled, the master repeats the same tag on MDataTagID during the datahandshake phase. For read transactions and writes with responses the slave returns the tag of the corresponding request on STagID while supplying the response. The same tag must be used for an entire transaction.

Note that a RdEx command and the associated WR or WRNP commands need not have the same tag IDs since they are separate transactions. Similarly, an RDL command and the associated WRC command need not have the same tag IDs since they are separate transactions.

Burst Definition

Location in OCP3 Specification

Introduction to Section 4.6, last paragraph, page 52.

Modified Text

~~A single (non-burst) request on an OCP interface with burst support is encoded as a request with any legal burst address sequence and a burst length of 1.~~

~~The ReadEx, ReadLinked, and WriteConditional commands can not be used as part of a burst. The unlocking Write or WriteNonPost command associated with a ReadEx command also can not be used as part of a burst.~~

A single word request on an OCP interface with burst support is encoded either a) as any non-BLCK burst address burst sequence with burstlength of 1 or b) as a BLCK burst request with burstlength of 1 and blockheight of 1.

The ReadEx, ReadLinked, and WriteConditional commands can only be used as part of a single-word request. The unlocking Write or WriteNonPost command associated with a ReadEx command can only be used as part of a single word request.

Single Request, Multiple Data Bursts

Location in OCP3 Specification

Footnote to Section 4.6.5, “Single Request / Multiple Data Bursts (Packets),” p55.

Modified Text

Note that the deleted text in the footnote applies to both the OCP2 and OCP3 specification.

When MBurstSingleReq is set to 1, write type transfers have MBurstLength * height datahandshake phases ~~and a single response phase (if writeresp_enable=1)~~ per request¹; while read-type transfers have MBurstLength * height response phases per request as shown in Table 21 on page 42. The height is MBlockHeight for BLCK address sequences, and 1 for all others.

¹ Additionally, there is a single response phase for WRNP and WRC write types while the WR and BCST types have this phase only if writeresp_enable is set to 1. ~~Note that WRC write type is not allowed in a burst.~~

Clarifications To Developers Guidelines

Incorrect Signal Names in Response Accept Sequence

Location in OCP3 Specification

Step E of the sequence described in Section 10.9, "Response Accept," pp169–170.

Modified Text

Signal MRespAccept incorrectly written as RespAccept. See correction in Step E, below.

Sequence

- A. The master starts a read request by driving RD on MCmd and a valid address on MAddr. The slave asserts SCmdAccept immediately, and it drives DVA on SResp and the read data on SData as soon as it sees the read request. The master is not ready to receive the response for the request it just issued, so it deasserts MRespAccept.
- B. Since SCmdAccept is asserted, the request phase ends. The master continues to deassert MRespAccept, however. The slave holds SResp and SData steady.
- C. The master starts a second read request and is ready for the response from its first request, so it asserts MRespAccept. This corresponds to a response accept latency of 2.
- D. Since SCmdAccept is asserted, the request phase ends. The master captures the data for the first read from the slave. Since MRespAccept is asserted, the response phase ends. The slave is not ready to respond to the second read, so it drives NULL on SResp.
- E. The slave responds to the second read by driving DVA on SResp and the read data on SData. The master is not ready for the response, however, so it deasserts MRespAccept.
- F. The master asserts MRespAccept, for a response accept latency of 1.
- G. The master captures the data for the second read from the slave. Since MRespAccept is asserted, the response phase ends.

Corrections to Compliance Rules

Modified Compliance Rule 1.2.29

Location in OCP3 Specification

Compliance rule 1.2.29, p389.

Modified Text

Rule 1.2.29 response reorder STagID tag interleave size

When `tags > 1` and `tag_interleave_size > 0` the slave must ensure that responses associated with packing burst sequences stay together up to the `tag_interleave_size`. When `tags > 1` and `tag_interleave_size == 0` no interleaving of responses between any packing burst sequences with different tags is allowed.

For packing bursts, when `tags > 1` and `tag_interleave_size > 0` the number of bursts that can stay together depend on the block alignment. A suggested implementation of this rule can be found in Section 12.4, "Tags," on page 232.

Protocol hierarchy	Response
Signal group	Dataflow - tag extensions
Critical signals	STagID
Assertion type	Reorder
References	OCP2: "Ordering Restrictions" on page 53 "Burst Interleaving with Tags" on page 58 OCP3: Section 4.7.1 on page 57 Section 4.9.1.7 on page 62

Deprecated Compliance Rule 1.2.30

Location in OCP3 Specification

Compliance rule 1.2.30, p390.

Modified Text

Compliance rule 1.2.30 is deprecated.

Duplicate Compliance Rule 1.7.2

Location in OCP3 Specification

Compliance rule 1.7.2, p390.

Modified Text

Compliance rule number 1.7.2 was applied to two rules. The duplicate has been renumbered as 1.7.10.

Rule 1.7.~~2~~10 `signal_value_MCmd_MConnect_not_connected`

The MCmd signal must be IDLE if MConnect is not in the M_CON state.

Protocol hierarchy	Control
Signal group	Sideband
Critical signals	MCmd, MConnect
Assertion type	Value
Reference	Section 4.3.3.2 on page 46

Clarifications to Compliance Checks

Incrementing Imprecise Read Requires Deassertion of MBurstPrecise

Location in OCP3 Specification

Step A of the sequence described in Section 10.11, “Incrementing Imprecise Read,” pp172–173.

Modified Text

Signal MBurstPrecise should be de-asserted at the start of the read request. See correction to Step A, below.

Sequence

- A. The master starts a read request by driving RD on MCmd, a valid address on MAddr, three on MBurstLength, INCR on MBurstSeq, and deasserts MBurstPrecise. The burst length is the best guess of the master at this point. MBurstSeq and MBurstPrecise are kept constant during the burst. MReqLast must be deasserted until the last request in the burst. The slave is ready to accept any request, so it asserts SCmdAccept.

Incorrect Signal Names in Response Phase Guidelines

Location in OCP3 Specification

Section 12.1.2.2, "Response Phase," p216.

Modified Text

Signal SResp originally incorrectly written as MResp. See correction, below.

12.1.2.2 Response Phase

The response phase begins when the slave drives SResp to a value other than NULL. When SResp != NULL, SResp is referred to as asserted. All of the other response phase outputs of the slave must become valid during the same OCP clock cycle as SResp asserted, and be held steady until the response phase ends. The response phase ends when MRespAccept is sampled asserted (true) by the rising edge of the OCP clock; if MRespAccept is not configured into a particular OCP, MRespAccept is assumed to be always asserted (that is, the response phase always ends in the same cycle it begins). If present, the master can assert MRespAccept in the same cycle that ~~M~~SResp is asserted, or it may stay negated for several OCP clock cycles. The latter choice allows the master to force the slave to hold its response phase outputs so the master can finish the transfer without latching the data signals.

Tag Interleaving for Packing Bursts

Location in OCP3 Specification

Immediately before the last paragraph of Section 12.4 “Tags,” pages 232–233 of Chapter 12, “Developers Guidelines.”

Modified Text

When MTagInOrder is asserted any MTagID and MDataTagID values are “don’t care”. Similarly, the STagID value is “don’t care” when STagInOrder is asserted. Nonetheless, it is suggested that the slave return whatever tag value the master provided.

The tag_interleave_size parameter specifies that responses of the same packing burst transaction must stay together up to this specified value, i.e., the parameter value is an upper bound. Note, however, that the number of responses that must stay together will be less than the tag_interleave_size when the packing burst begins on an unaligned data block boundary. The last set of responses that must stay together for this packing burst can also be less than the tag_interleave_size. The following pseudo-code makes this notion precise.

```
// Given two MAddr values MAddr_previous and MAddr_current,
// use the following pseudo-code to determine if MAddr_current
// belongs to the same power-of-two, aligned data block as
// MAddr_previous.
//
// 'alignMask' identifies the least significant bits of the
// power-of-two, aligned data block. E.g.,

| <u>block size (bits)</u> | <u>alignMask</u> |
|--------------------------|------------------|
| <u>32</u>                | <u>0x3</u>       |
| <u>64</u>                | <u>0x7</u>       |

alignMask = [ tag_interleave_size * ( data_wdth / 8 ) ] - 1

if
    (MAddr_current & ~alignMask)
        == (MAddr_previous & ~alignMask)
then
    MAddr_current belongs to the same power-of-two,
    aligned data block as MAddr_previous.
else
    MAddr_current belongs to a different power-of-two,
    aligned data block than MAddr_previous.
```

Multi-threaded OCP interfaces can also have tags. Each thread’s tags are independent of the other threads’ tags and apply only to the ordering of transfers within that thread. There are no ordering restrictions for transfers on different threads. The number of tags supported by all threads must be uniform, but a master need not make use of all tags on all threads.

STagInOrder Definition

Location in OCP3 Specification

Table 93 on page 459.

Modified Text

A new row for STagInOrder is added to Table 93, as shown below:

Table 93 OCP Trace File, Line Field Decoding

Field	Parameter Condition	Field Width in Bits	Format
<i>... Table rows eliminated for clarity</i>			
STagID	tags > 1 and resp is 1	tagid_width ²	hexadecimal
<u>STagInOrder</u>	<u>tagorder is 1</u>	<u>Always 1</u>	<u>hexadecimal</u>
SData	sdata is 1	data_width	hexadecimal
<i>... Table rows eliminated for clarity</i>			

Connection Signals Definition

Location in OCP3 Specification

Table 93 on page 459.

Modified Text

New rows should be added to Table 93 for connection signals, as shown below:

Table 93 OCP Trace File, Line Field Decoding

Field	Parameter Condition	Field Width in Bits	Format
<i>... Table rows eliminated for clarity</i>			
MFlag	mflag is 1	mflag_wdth	binary
MError	merror is 1	Always 1	binary
<u>MConnect</u>	<u>connection is 1</u>	<u>Always 2</u>	<u>hexadecimal</u>
SFlag	sflag is 1	Always 1	binary
SError	serror is 1	Always 1	binary
<u>SConnect</u>	<u>connection is 1</u>	<u>Always 1</u>	<u>hexadecimal</u>
<u>SWait</u>	<u>connection is 1</u>	<u>Always 1</u>	<u>hexadecimal</u>
<u>ConnectCap</u>	<u>connection is 1</u>	<u>Always 1</u>	<u>binary</u>
<i>... Table rows eliminated for clarity</i>			

OCP-IP Administration

3855 SW 153rd Drive
Beaverton, OR 97006
Ph: +1 (503) 619-0560
Fax: +1 (503) 644-6708
admin@ocpip.org
www.ocpip.org

