OpenCAPI™

www.opencapi.org

OpenCAPI 4.0
32 Gbps PHY Mechanical Specification

Version 1.0
17 March 2021

Approved

Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

_____

OpenCAPI 4.0 32 Gbps PHY Mechanical Specification

OpenCAPI PHY Mechanical Work Group
OpenCAPI Consortium

Version 1.0 (17 March 2021)

Copyright © OpenCAPI Consortium 2020, 2021

Abstract

This document describes technical details for the design of carrier cards and 32 Gbps cable implementations. It is the work product of the OpenCAPI Consortium PHY Signaling Work Group.

This document is handled in compliance with the requirements outlined in the OpenCAPI Consortium Work Group (WG) process document. Comments, questions, etc. can be submitted to membership@opencapi.org.

_____

Version 1.0 Participants
17 March 2021 Page 2 of 23
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

_____

Participants

Brian Allison, IBM, *Chair*

Greg McSorley, Amphenol Corporation,
Technical Editor

_____

Version 1.0   Participants
17 March 2021   Page 3 of 23
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

Contents

Version 1.0
17 March 2021
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

Contents
Page 4 of 23

_____

List of figures

_____

Version 1.0
17 March 2021
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

List of figures
Page 5 of 23

List of tables

Version 1.0 List of tables
17 March 2021 Page 6 of 23
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

_____

Revision log

Each release of this document supersedes all previously released versions. The revision log lists all significant changes made to the document since its initial release.

Revision date	Summary of changes
17 March 2021	Version 1.0. Initial release to OpenCAPI Consortium.

_____

Version 1.0   Revision log
17 March 2021   Page 7 of 23
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

_____

About this document

This specification describes technical details for the design of carrier cards and 32 Gbps cable implementations.

This section contains some general format information and document conventions.

Who should read this document

This specification is intended for designers who plan to develop products that use OpenCAPI. This document provides a technical overview of the mechanical component guidance for 32 Gbps OpenCAPI system design.

Architecture compliance terminology

In architecture descriptions, certain terms carry meaning in addition to their normal use in English. The following terms are used within this architecture specification to describe the requirements an implementation must meet to be considered compliant.

Table 1. Architecture terms

Term	Description
invalid	Used for multi-bit fields where the contents are not reliable. The field or bus shall not be examined for any functional or error checking actions.
may	An architectural option indicating that an implementation is allowed to have this behavior or characteristic.
reserved	With respect to a field of a register or bus: • A reserved field shall be set to 0 by an implementation. • A reserved field shall not be examined by an implementation. With respect to a code point: • A reserved code point shall not be issued by a compliant implementation • A reserved code point shall cause a bounded undefined response (that is, it won't hang the system). • A reserved code point may be used in future revisions of the architecture. The architecture may specify that the use of a reserved code point is an error condition.
shall	An architectural requirement indicating a required behavior or characteristic.
uncertain	Used for single-bit fields where the contents are not reliable. The field or bus shall not be examined for any functional or error checking actions.
undefined	When the value of a field or a bus is undefined, the value may vary between implementations and may vary for a particular implementation for different actions. An implementation shall not examine a field when its value is undefined for functional purposes. However, the field may be checked for errors in those cases where an implementation includes error checking (that is, parity, ECC and so on).

_____

Version 1.0
17 March 2021
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

About this document
Page 8 of 23

_____

Conventions

The OpenCAPI Consortium documentation uses several typesetting conventions.

Bit and byte numbering

Throughout this document, little-endian notation is used, which means that bits and bytes are numbered in descending order from left to right.

Thus, in the description of a 4-byte field, bit 31 is the most significant bit (MSb) and bit 0 is the least significant bit (LSb). The corresponding byte numbering is also shown.

MSb   LSb

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
byte 3								byte 2								byte 1								byte 0							

Big-endian and little-endian byte ordering are shown in _Table 1-2_ and described in the _Power ISA, version 3.0, Book I_.

Table 2. Big- and little-endian comparisons

LE	7	6	5	4	3	2	1	0
	Bit numbering within a byte							
BE	0	1	2	3	4	5	6	7

4-byte field with character data shown

LE	3	2	1	0
Content:	M	I	K	E
BE	0	1	2	3

Illustrating the difference between little endian and big endian storing to memory of the 4-byte field shown to the left.

Memory offset	LE stored	BE stored	
0	E	M	
1	K	I	
2	I	K	
3	M	E	

Version 1.0
17 March 2021
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

About this document
Page 9 of 23

_____

Representation of numbers

The notation for bit encoding is as follows:

- Hexadecimal values are preceded by x and enclosed in single quotation marks. For example x'0A00'. Bit numbering is little endian and, in this example, is 15 to 0.
- Binary values in sentences are shown in single quotation marks; for example, '1010'. Bit numbering in is little endian and, in this example, is 3 to 0.
- ^{n}x means the replication of x, n times. That is, x is concatenated to itself n-1 times. $^{n}0$ and $^{n}1$ are special cases:
  - $^{n}0$ means a field of n bits with each bit equal to 0. For example, $^{5}0$ is equivalent to '00000'.
  - $^{n}1$ means a field of n bits with each bit equal to 1. For example, $^{5}1$ is equivalent to '11111'.

RTL notation

RTL notations are used to specify the architectural transformation performed by execution of a command.

Notation	Meaning
←	Assignment.
‖	Concatenation.
=, ≠	Equal, not equal relations.
≥, ≤	Greater than or equal to, less than or equal to relations.
+	Two's complement addition.
-	Two's complement subtraction, unary minus
∧	Bitwise logical OR
∨	Bitwise logical AND
⊕	Bitwise logical exclusive OR
Max(x,y)	Returns x when x ³ y; otherwise, returns y.
Min(x,y)	Returns x when x £ y; otherwise, returns y.
{x...y}	All integer values from x through y.
A = {x...y}	Returns true when A is a member of the set of integer values in the range of x through y.

_____

Version 1.0
17 March 2021
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

About this document
Page 10 of 23

_____

Notes

This section describes Engineering, Developer, and Editor notes.

Engineering notes

Engineering notes provide additional implementation details and recommendations not found elsewhere. The notes might include architectural compliance requirements. That is, the text might include Architecture compliance terminology. These notes should be read by all implementation and verification teams to ensure architectural compliance.

Engineering note:

This is an example of an Engineering note. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin cursus hendrerit enim, vel tempus nibh ornare ut. Quisque ac augue eu augue convallis hendrerit. Mauris iaculis viverra ipsum nec dapibus. Nunc at porta libero. Curabitur luctus ultrices augue non pulvinar. Vestibulum mattis non ipsum at venenatis. Suspendisse euismod, neque et suscipit luctus, odio metus semper lectus, quis volutpat est libero quis nunc. Vivamus rutrum mauris sed tristique malesuada.

Developer notes

Developer notes are used to document the reasoning and discussions that led to the current version of the architecture. These notes might also include recommended changes for future versions of the architecture, or warnings of approaches that have failed in the past. These notes should be read by verification teams and contributors to the architecture.

Developer note:

This is an example of a Developer note. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin cursus hendrerit enim, vel tempus nibh ornare ut. Quisque ac augue eu augue convallis hendrerit. Mauris iaculis viverra ipsum nec dapibus. Nunc at porta libero. Curabitur luctus ultrices augue non pulvinar. Vestibulum mattis non ipsum at venenatis. Suspendisse euismod, neque et suscipit luctus, odio metus semper lectus, quis volutpat est libero quis nunc. Vivamus rutrum mauris sed tristique malesuada.

Editor notes

Editor notes are reminders to the editors and other contributors of additional work that is required. Editor notes may appear as red text in the body of any section or may include an entire section. This is to indicate text that is either out of date or is speculative (unapproved) in nature. The red text might also include directions to the editor for changes to be applied to a future revision of the document. Approved versions of a document are not expected to be released with red text unless approved by the owners of the document.

_____

Version 1.0   About this document
17 March 2021   Page 11 of 23
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

Terms

The following terms are used in this document.

AAC	Advance accelerator cable.
AC	Alternating current.
ASIC	Application-specific integrated circuit.
BRD	Board.
CPU	Central processing unit.
FPGA	Field-programmable gate array.
Gbps	Gigabits per second.
GND	Ground.
I^2C	Inter-integrated circuit.
INT	Interrupt
ISA	Instruction set architecture.
LGA	Land grid array.
OIF	Optical Internetworking Forum
PCB	Printed circuit board.
PCIe	Peripheral Component Interconnect Express.
RST	Reset line.
RTL	Register transfer language.
SCL	I^2C bus clock
SDA	I^2C bus data
V	Volt.

Version 1.0   Terms
17 March 2021   Page 12 of 23
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

1. Overview

This specification describes the technical details for the design of carrier cards and 32 Gbps cable implementations. This specification is constrained to the use of the tested and verified connector technology and I/O assignments. The scope of this document is to provide a technical overview of the mechanical requirements for 32 Gbps OpenCAPI system design.

Version 1.0
17 March 2021
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

Overview
Page 13 of 23

_____

2. Carrier card

Figure 2-1. illustrates an example of an OpenCAPI accelerator card configuration for a 2-socket system.

Figure 2-1. Internal configuration in a 2-socket system

Example A. Internal Configuration in 2-Socket System
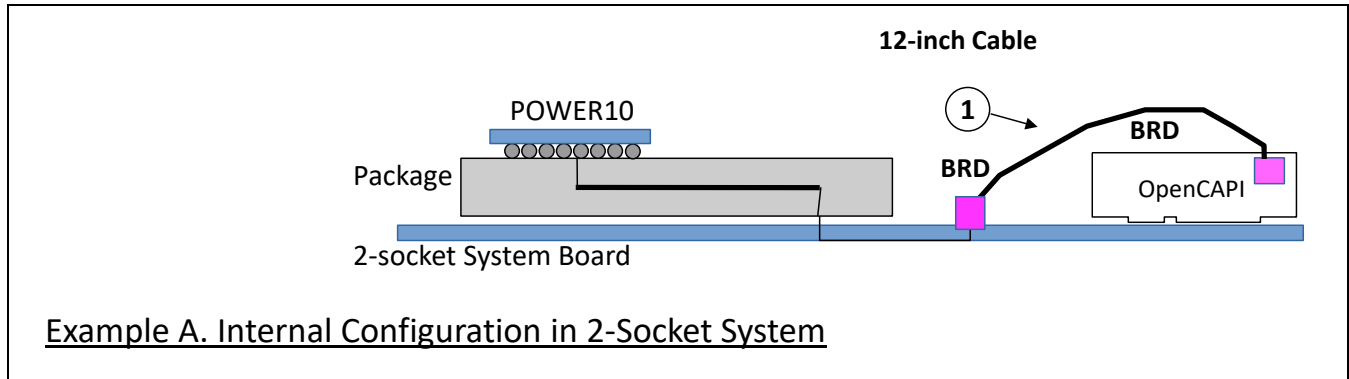
2.1 AAC Backward compatibility

Previous versions of the SlimSAS OpenCAPI card(s) can be plugged to a system using a SlimSAS cable. The system can be configured to operate at the reduced speed of 25 Gbps in backward-compatibility mode.

In *Figure 2-1*, a 12-inch board SlimSAS to SlimSAS (74-pin connector) cable can be used for backward compatibility.

2.2 Carrier card size

The carrier card design should conform to the size requirements for PCIe add-in cards defined by the *PCI Express® Card Electromechanical Specification Revision 3.0* or later. It is recommended that the carrier card be kept to the half or three-quarter lengths per the specification to better accommodate the advanced accelerator connector routing in the chassis.

2.3 Carrier card connector

Only power is being removed from the PCIe slot for the carrier card. Therefore, the size of the PCIe edge connector can be determined by the amount of card power required and the mounting stability and flexibility required. The X1 connector size provides full power and is the most universal size but might not provide the best retention and mechanical stability compared to a x8 connector.

2.4 Carrier card power

The power available from the PCIe edge connector is limited to 75 W. Additional power requirements can be addressed by the auxiliary power connector options identified in the *PCI Express® Card Electromechanical Specification*.

_____

Version 1.0
17 March 2021
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

Carrier card
Page 14 of 23

_____

2.5   Carrier card cooling

See the *PCI Express® Card Electromechanical Specification* for information on cooling requirements for the carrier card.

_____

Version 1.0   Carrier card
17 March 2021   Page 15 of 23
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

_____

3. Advance accelerator cable

The OpenCAPI platform supports the optional 32 Gbps interface to the advance accelerator processor unit in a different drawer of the rack or the riser card plug-in to the PCIe slot in the same system. This section only contains information on topologies, connectivity, and routing guidelines for the advance accelerator cable (AAC) interface.

3.1   Accelerator cable circuit schematic

Figure 3-1 illustrates a typical accelerator cable circuit schematic.

Figure 3-1. Accelerator cable circuit schematic
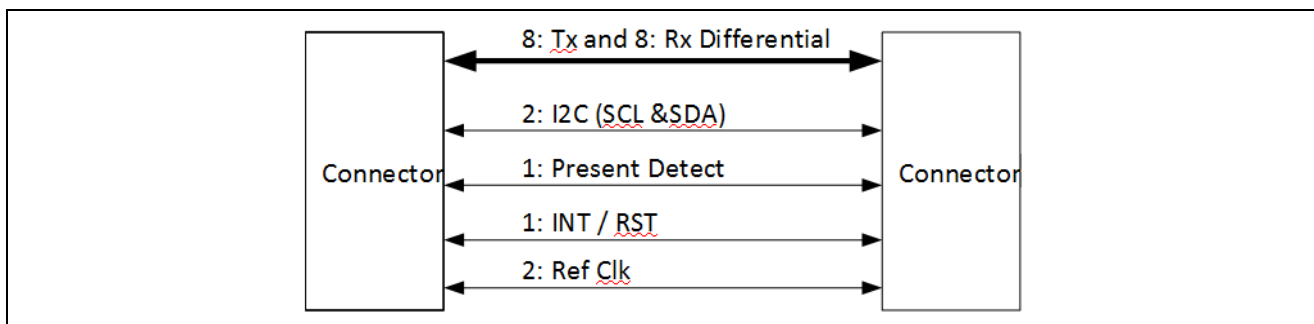
3.1.1   Accelerator cable signal definition

The interconnect allows lane and polarity reversal. Pin swapping is not allowed. Signal direction is defined as follows:

- Tx – is from the Host (CPU) perspective for both the host board as well as the carrier card
- Rx – is from the Host (CPU) perspective for both the host board as well as the carrier card
- AC coupling must be done on both Rx and Tx differential data signals on the carrier card

The following sideband signals are present:

- Ref Clk – Differential reference clock. Driven by the CPU. 156.25 MHz for OIF applications or 133.33 MHz for JEDEC applications. The external reference clock electrical specification can be found in the _OpenCAPI 32 Gbps Physical Signaling Specification_ (_Table 5-1. External reference clock specification_).

- I2C (SCL/SDA) – I²C bus clock and data. I²C on the carrier card is only a slave implementation. The CPU drives these signals to a 3.3 V level from the main 3.3 V power supply. Recommended to have level translators on the carrier card to translate to 1.8 V. Pulled high (1.8 V) on the carrier card. These pins must be at a steady state voltage before the OpenCAPI device is taken out of reset.

- INT/RST – Reset to OpenCAPI FPGA controlled by the processor I²C bus. The CPU drives these signals to a 3.3 V level from the main 3.3 V power supply. This signal must have a pull-up resistor on the host side. Level translators are recommended on the carrier card to translate to 1.8 V. This signal must have a pull-up resistor on the carrier side as well, after the level translation.

_____

Version 1.0   Advance accelerator cable
17 March 2021   Page 16 of 23
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

_____

- Cable Pre-Detect – Presence detect. Cable Pre-Detect must have a 49.9 Ω pull-down to GND resistor located on the adapter card.
- SPARE_1/2 – Spare pins. May be left floating.

3.2   Accelerator cable pin assignments

Figure 3-2 illustrates the accelerator cable pin definition.

Figure 3-2. Accelerator cable pin definition
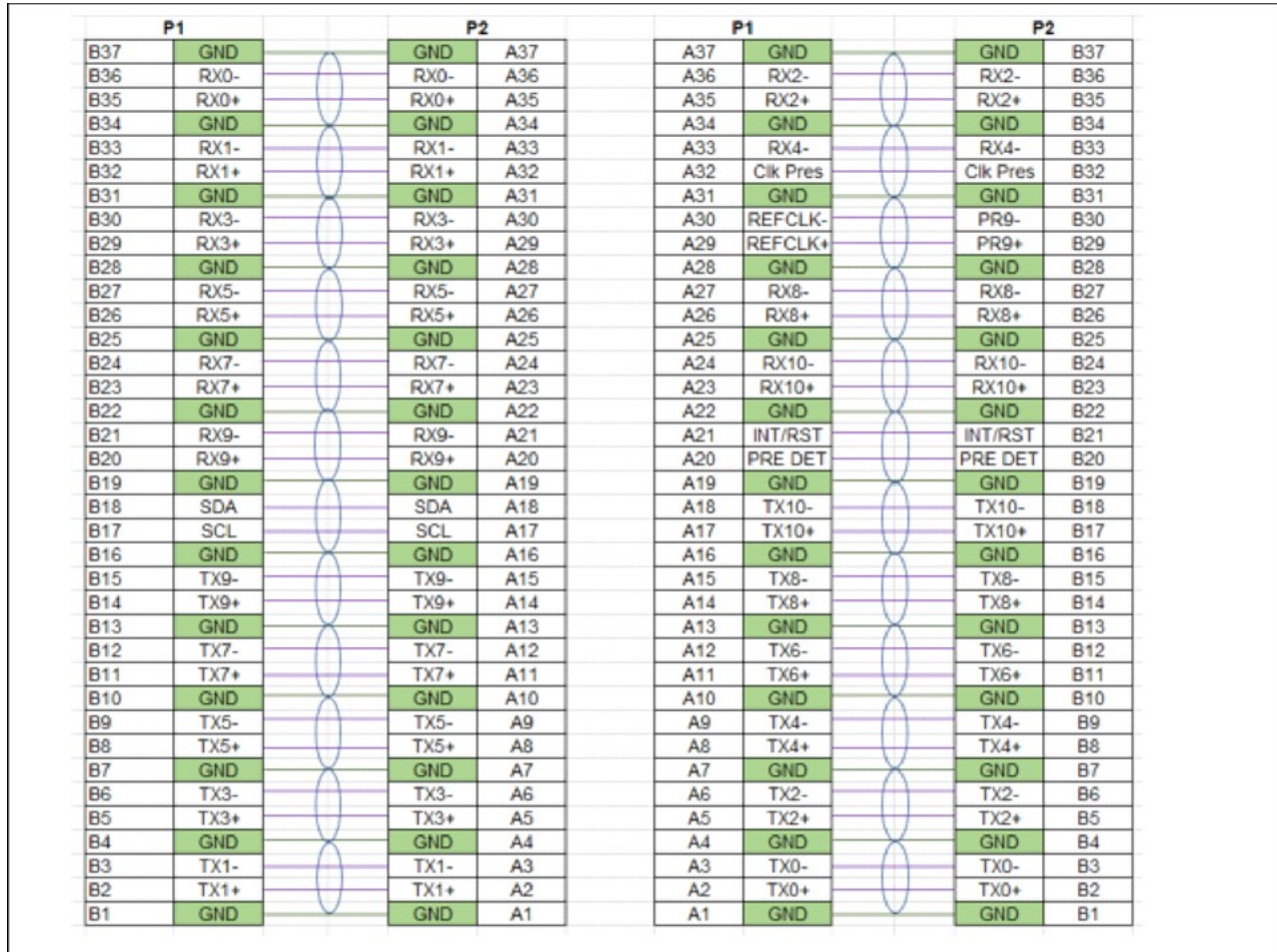
3.2.1   Internal cable pin ordering and mapping

The interconnect allows lane and polarity reversal. Pin swapping is not allowed. The GND board pin ordering and list are shown in *Figure 3-3*. Also shown is a generic pin list for the cable connection.

_____

Version 1.0
17 March 2021
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

Advance accelerator cable
Page 17 of 23

_____

Figure 3-3. GND board pin ordering
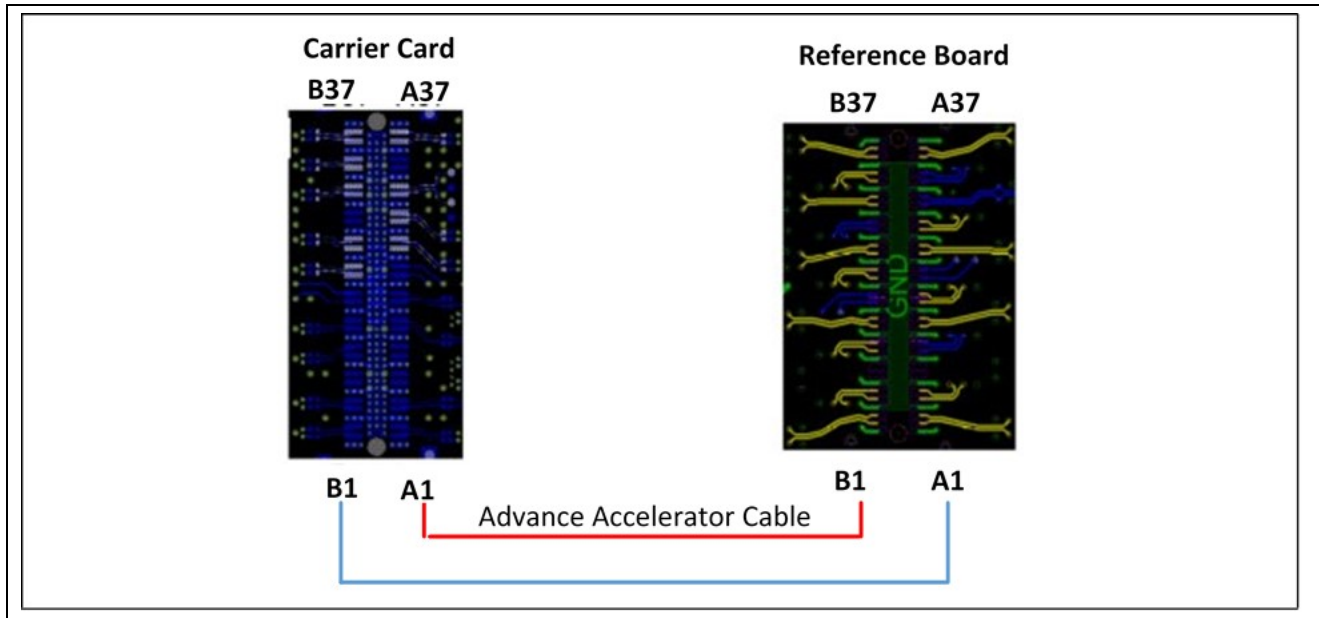
Note: The raw wire within the cable is 12 inches, 30 AWG (6 dB/meter). The I/O assignments are referenced from the CPU board side.

3.2.2   AAC connector escape routing example

Figure 3-4 shows some AAC connector escape routing examples. The connector antipad goes through the top two GND plane layers. The signal must not cross the antipad void.

Connector break-out P and N: The connector break out P and N should be matched immediately to the first via.

Figure 3-4. AAC connector escape routing example

_____

Version 1.0   Advance accelerator cable
17 March 2021   Page 18 of 23
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

_____

4. Advance accelerator card design guidelines

4.1   Advanced accelerator component loss budget

Table 4-1 lists the loss budgets at 25 Gbps and 32 Gbps for an internal configuration in a 2-socket system.

*Table 4-1. Internal configuration in a 2-socket system*

Component	Loss Budget (dB) at 25 Gbps	Loss Budget (dB) at 32 Gbps	Notes
CPU Complex	4.1	4.25	Silicon, substrate trace, and LGA socket
Mother Board Trace	5.1	7.5	Under CPU LGA socket via, 6.5 in PCB Meg6, connector via
12 in x Gen 5 Cable 1	7.4	4	Mated connector pads to mated connector pads
OpenCAPI Card	2.5	4.7	FPGA via to connector pads (3.5 in Meg6)
Advance Accelerator Socket/Package	1.8	2.75	FPGA silicon and laminate package
Total	20.9	23.2	Receiver tolerance (-30 dB). (See Note 1.)
1. The worst-case loss budget requirement at 25 Gbps is -20 dB. At 32 Gbps, the worst-case loss budget requirement is -30 dB. This is due to improved receiver tolerance in the 32 Gbps design.			

_____

Version 1.0   Advance accelerator card design guidelines
17 March 2021   Page 19 of 23
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

_____

4.2 Advanced accelerator card cable connector escape routing

Figure 4-1 shows the SlimSAS connector layout.

Figure 4-1. SlimSAS connector break-out
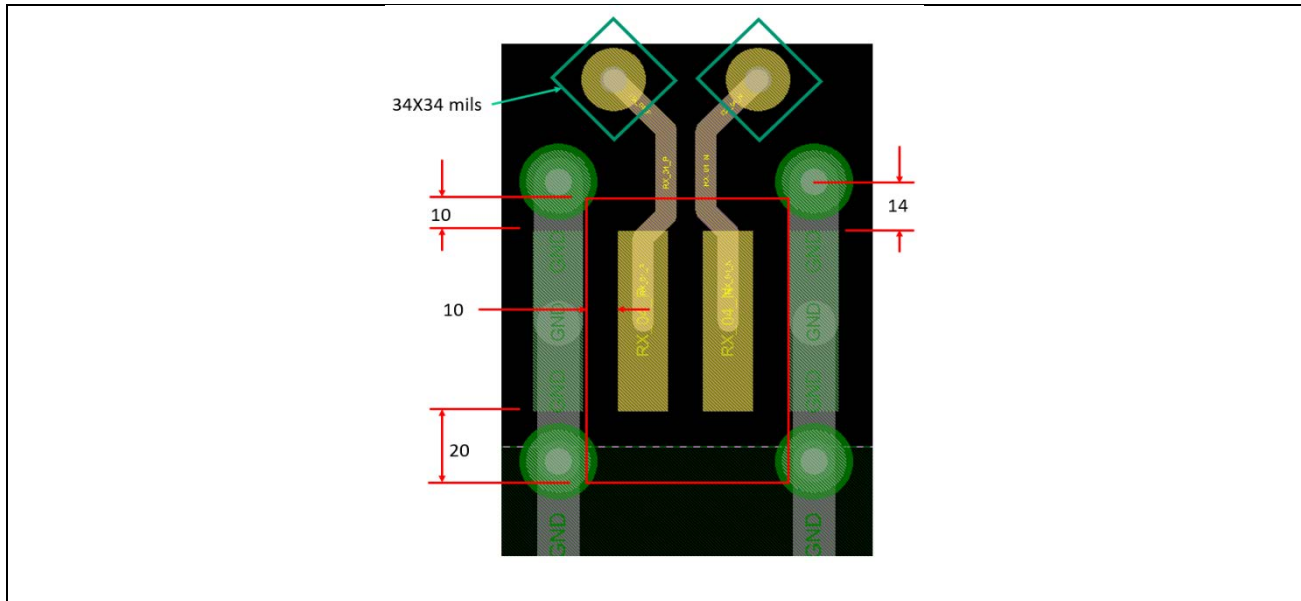
4.2.1 SlimSAS connector escape rules

The SlimSAS connector escape rules are as follows:

- Signal via stub length = Blind via
- Top layer pad diameter = 18 mils
- Bottom layer pad diameter = 18 mils
- Anti-pad size = 34 mils x 34 mils square shape
- Via drill size = 10 mils
- Pin anti-pad void on reference ground immediate to the pad = 60 mils x 80 mils
- Flooding copper on top layer to connect all ground vias; maintaining a minimum of 10 mils air gap against other elements

4.3 OpenCAPI card general layout rules

Under the ASIC/FPGA, the break-out rules are as follows:

- The signal wiring under the ASIC must be centered between the via.
- The wire must not be "neck-down" but maintain the same open-area wiring dimensions for the 85 Ω impedance specified by the stack-up of choice.
- The diagonal wiring pair-to-pair spacing must maintain 8 mils minimum spacing between pairs-to-pairs carrying the signals traveling in the same direction.
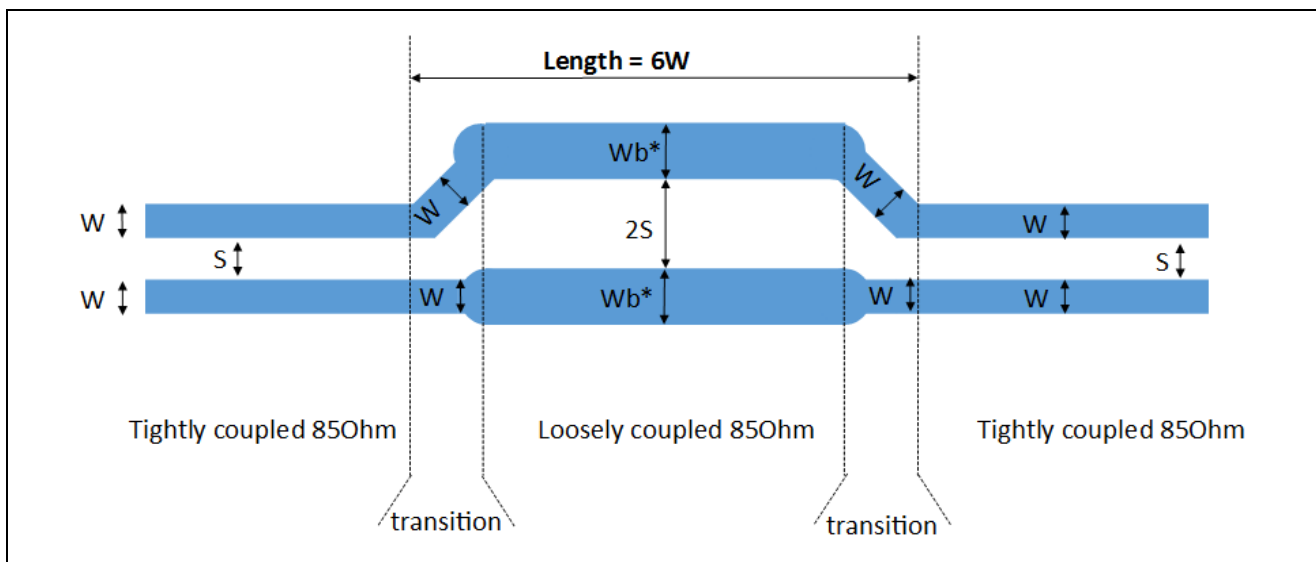
_____

Version 1.0
17 March 2021
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

Advance accelerator card design guidelines
Page 20 of 23

- For pairs-to-pairs traveling in the opposite directions, Tx and Rx, an empty wiring channel of separation is required.
- No differential wiring between signals or signal via is allowed.

The open area wiring guideline is as follows:

- Open area wiring impedance = 85 Ω ±10%
- Trace length = approximately 5-inch Meg6 (to meet 6 dB total loss budget of the card)
- Maximum uncoupled length = 300 mils
- 15° diagonal wiring/jogging is required.
- Pair-to-pair matching length = 500 mils
- Serpentine rule:
    - Dynamic phase tolerance = 10 mils in each 0.5 inch
    - Static phase tolerance: 2.5 mils
    - Follow the sawtooth wiring guideline (see *Figure 4-2*) when trying to match P/N skew

Figure 4-2. Sawtooth wiring guideline

Note: Wb must be selected to achieve 85 Ω differential impedance assuming 2S in-pair spacing.

Version 1.0 Advance accelerator card design guidelines
17 March 2021 Page 21 of 23
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

_____

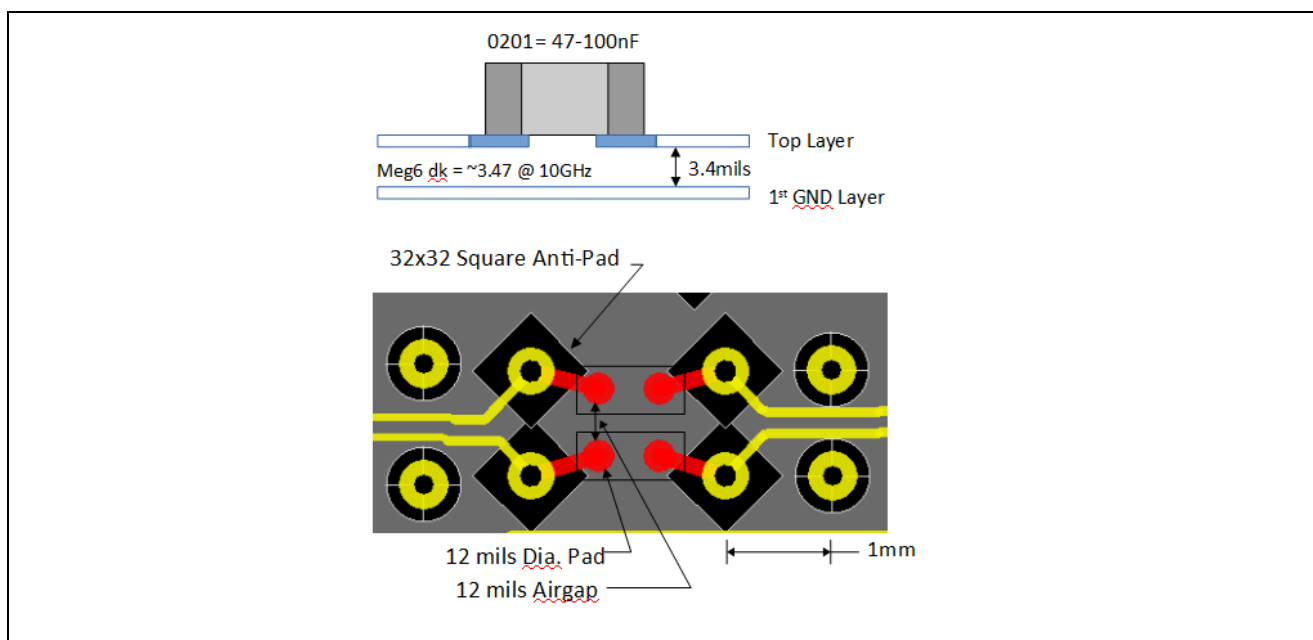*Table 4-2* shows the corresponding spacing rules.

Table 4-2. Spacing rules

Spacing Type	Spacing Amount
Pair-to-pair within group of same signal direction	5W or 20 mils (whichever is greater)
Tx-to-Rx pair	10W or 40 mils (whichever is greater)
Pair-to-power shape	10W or 10 mils (whichever is greater)
Noisy-switching 12 V, 48 V, 54 V via	8W or 30 mils (whichever is greater)
Pair-to-bolt whole / anti-pad edge open area	5 mils to edge or 10 mils to pad
Same line to same line spacing	5W or 20 mils (whichever is greater)
Pair to other bus/nets	10W or 40 mils (whichever is greater)
Spacing from edge of board	100 mils (if not, shielding ground via is required)

4.4   AC coupled capacitor layout

Capacitor discontinuity should be kept to within ±5% of the 85 Ω reference. *Figure 4-3* is an example of an IBM reference design to reduce the discontinuity. Unless the OpenCAPI adapter is using the same material and construction, the adapter card layout must be modeled and designed to meet this specification.

Figure 4-3. IBM reference design to reduce discontinuity

_____

Version 1.0   Advance accelerator card design guidelines
17 March 2021   Page 22 of 23
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only

_____

4.5   Layer transitional via

The signal trace can change layers once through a transitional via. Via size and via anti-pad must be tuned to meet ±5% of the fundamental impedance of the trace. Via pad references to the GND layer below should be voided to the same size of the via pad itself to minimize impedance mismatch. If the signal transmits more than one signal layer, a GND stitch via must be placed near the signal via to within 1 mm radius.

_____

Version 1.0   Advance accelerator card design guidelines
17 March 2021   Page 23 of 23
Approved for Distribution to OpenCAPI Members
Approved for Distribution to Non-Members for Learning Purposes Only