Bunch of Wires: An Open Die-to-Die Interface

Shahab Ardalan
Ayar Labs
Santa Clara, CA
shahab@ayarlabs.com

Halil Cirit
Facebook
Menlo Park, CA
hcirit@fb.com

Ramin Farjad
Marvell
Santa Clara, CA
rfarjadrad@marvell.com

Mark Kuemerle
Marvell
Essex Junction, VT
mkuemerle@marvell.com

Ken Poulton
Keysight Technologies
Santa Clara, CA
ken.poulton@keysight.com

Suresh Subramanian
Apex
Santa Clara, CA
suresh@apexsemi.com

Bapiraju Vinnakota
Broadcom Inc
San Jose, CA
bapiraju.vinnakota@broadcom.com

Abstract— **Bunch of Wires (BoW) is a new open die-to-die (D2D) interface. BoW's objective is to allow designers to gracefully trade-off performance for design and packaging complexity across a wide range of process nodes. BoW bandwidth can range from 80 Gbps per 16-bit BoW slice with a simple design to 512 Gbps with complex design and packaging. With this flexibility, BoW directly enables heterogeneous integration, a primary advantage of chiplets. In turn, this will enable economies of scale for services and technologies for BoW. In this paper, we discuss progress on the BoW specification based on extensive design and performance studies by engineers from multiple companies. These studies aim to specify BoW so as to make it easy to use in a system. This open innovation project will deliver a low-complexity D2D interface with competitive power-performance metrics with the economies of scale associated with an open ecosystem.**

Keywords—BoW, Bunch of Wires, Die-to-Die, interface, chip-to-chip, chiplet, MCM.

I. Introduction

Commercial interest in die-to-die (D2D) interfaces for chiplet-based designs continues to grow. Bunch of Wires (BoW) is a new open D2D interface introduced at Hot Interconnect in 2019. The basic objective of BoW is to enable designers to gracefully trade-off performance for design effort and packaging complexity. Basic BoW uses CMOS IOs [1] and is designed to be easy to port to multiple process nodes. For bandwidth similar to 112G SerDes, implementers can choose to invest in advanced design or packaging. The base specification calls for a "slice" operating over a 16-bit wide source-synchronous DDR bus, with optionally terminated lines [2]. Two modes, Basic and Fast, provide raw bandwidths of 80 and 256 Gbps per slice, respectively. In both, slices can be stacked 4 deep to achieve a beachfront bandwidth from 320 and 1024 Gbps/mm with regular bumps in basic packaging to up to ~1 Tbps/mm with microbumps in advanced packaging.

The BoW interface is being developed by the Open Domain-Specific Architecture (ODSA) [2] project in the Open Compute Project. Engineers from a diverse group of companies have worked to define a BoW interface available as an open specification with no technology licensing costs [3]. The open specification and additional collateral will enable economies of scale for services and technologies for BoW such as interface IP, packaging and test, accelerating adoption of chiplet based systems. In this paper, we discuss progress on the definition and the results of design and performance studies by this group. The studies evaluated designs, implementation complexity and performance to use a BoW link in a system and the transport common data protocols over a BoW link: (a) at mature (65 nm) and advanced (5 nm) process nodes; (b) with regular-sized bumps on a organic substrates and microbumps on advanced packaging (Si interposers, EMIB, InFO, etc);

With this flexibility, BoW uniquely enables heterogeneous integration, a primary advantage of chiplets, across a wide range of process nodes and packaging technologies for products in a diverse price-performance range (high-performance accelerators, NICs, data converters, etc). No other open interface offers this tradeoff.

II. BoW Definition

BoW has the following target attributes [3] (while an implementation may achieve higher performance):

- A set of backward-compatible D2D interfaces unencumbered by technology license costs.
- Inexpensive to implement with the flexibility to trade off throughput per wire for design, and packaging complexity.
- Portable across: (a) process nodes ranging from 65 nm to 5 nm; (b) Multiple bump pitches and packaging technology.

2332-5569/20/$31.00 ©2020 IEEE
DOI 10.1109/HOTI51249.2020.00017

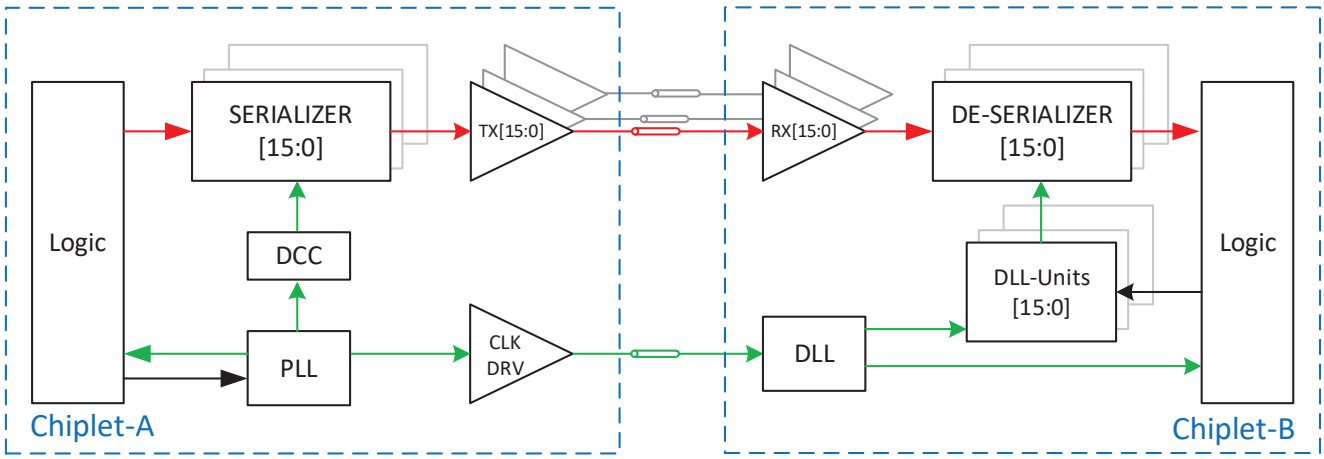


Figure 1: BoW reference design

- Throughput per Chip Edge target range (Rx+Tx): (a) 100 Gbps/mm with simple interface circuitry and all packaging options. For example, be able to achieve this goal at a bump pitch of 150 um and with a die edge stack depth no greater than 2 signal layers and 3-2-3 organic laminate packaging; (b) 1 Tbps/mm with advanced interface circuitry or an advanced packaging. For example, be able to achieve this goal with a bump pitch of 50 um and with a die edge stack depth no greater than 4.
- Competitive power-performance metrics: (a) power, < 1 pJ/bit, as defined by Tx IO Pad, wire and Rx IO Pad; (b) Latency: <5 ns without FEC, <15 ns with FEC. Based on experience, the 5 ns target meets the requirements of high performance applications and has been demonstrated to be achievable.

BoW is a simple source-synchronous architecture. In a BoW link, the sending slice (Chiplet-A in Fig 1) transmits clock and data signals. The receiving slice (Chiplet-B in Fig. 1) receives both clock and data where phase alignment between clock and data is adjusted. The source-synchronous (clock forwarded) architecture provides a low power link solution where there is no clock generator block (such as a phase lock loop) at the receiving chiplet. Figure 1 shows an example implementation of a 16-bit slice. The logic in Chiplet-A delivers a reference clock to the transmitter PLL. At the receiver a Delay Locked Loop (DLL) skews the clock from the transmitter to align with each channel.

A BoW link between two chiplets is made up of wires, slices, and stacks. The basic unit is a **slice**, with 18 or 20 signal bumps: 2 for the differential clock, 16 single-ended data and optional signals FEC and AUX. The long edge is parallel to the chip edge. Multiple slices may be placed in a **stack**. A 4-stack is approximately 10 x 10 bump pitches, or about 1.3 x 1.3 mm for 130 um bump pitch. A logical **link** from one chiplet to another is composed of one or more stacks placed along the chip edge. A link is typically configured with equal numbers of Rx and Tx slices, but may be asymmetric or one-way.

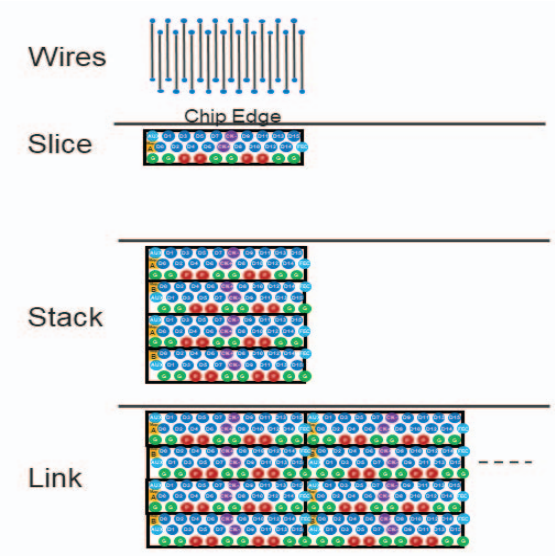


Figure 2: BoW link components

Two modes are defined, more may be added later. The rates are the "maximum-minimum" data rates for interoperability. Two ends of a BoW link may be configured to operate at higher clock and data rates.

Table 1 summarizes the target rates from [3].

- **BoW Basic**: All implementations must support the Base configuration with a minimum clock rate of 2.5 GHz, 5 Gbps/pin DDR. Analysis since the original proposal has shown that BoW basic will require source termination for the distances >2mm typically associated with organic substrates. A maximum clock rate of 4 GHz, 8 Gbps/pin DDR can also be supported for shorter traces (see Table 1)
- **BoW Fast**: This mode (also referred to as BoW Terminated in [3]) is expected to use termination at the destination to minimize signal reflection and improve the data rate per line. A Fast configuration shall support a minimum clock rate no greater than 8 GHz, 16 Gbps/pin DDR.

TABLE I. BOW MODES AND TARGETS

Configuration	Interop Data rate	Trace Length	Termination
Basic (Interposer)	<= 5 Gbps <= 8 Gbps	<= 2mm <= 1mm	Optional
Basic (Laminate)	<= 5 Gbps <= 8 Gbps	<= 10mm <= 5mm	Source (Tx) Termination
Fast	<= 16 Gbps	<= 50mm	Double Termination

III. DESIGN STUDIES

We review studies for key attributes. The studies focused on choosing options that made package and system integration in heterogeneous systems easier while preserving performance.

A. *Operational Voltages*

To support heterogeneous integration, one of the key objectives of BoW is to be implementable in a variety of technologies which makes voltage selection critical. BoW may be implemented using any supply or signaling voltage, but must support a mode that supports signaling levels consistent with a 0.75V supply with a 5% tolerance. It should be noted that BoW does not dictate a power supply voltage for the implementation, but by specifying a standard signaling level for basic interoperability the implementer has greater freedom to optimize their design to a given technology.

B. *Package Integration*

In a BoW link, each slice is configured to be Tx or Rx and is paired with an Rx or Tx slice on the far end. To keep package circuit design simple, wires must be short, no more than 10 mm to maintain low wire loss and wire-to-wire skew. Figure 3 shows only the wires for the slices at the chip edge. The slice is oriented parallel to the chip edge in order to allow flexibility for different levels of packaging complexity.

Note that the bump patterns are not prescribed in BoW, only the wire order. In organic laminate packages, a bump pitch of 130 um and a wire pitch limit of 50 um are common today. But since only the wire order is specified, chiplets with somewhat different bump pitches can be interconnected. In these packages, the wires for one slice will typically occupy one signal routing layer, the next layer will be ground for signal integrity and the next slice in the stack will be on the next deeper signal layer.

An organic package with a low layer count will allow use of just one or two slices in a stack, while a package with more layers supports use of more slices in a stack. With interposers [4], InFO [5] and other advanced interconnect technologies that allow bump pitches of 50 um or less and wire pitches below 10 um, the wires for multiple slices might be interleaved on a single signal layer in the interconnect.

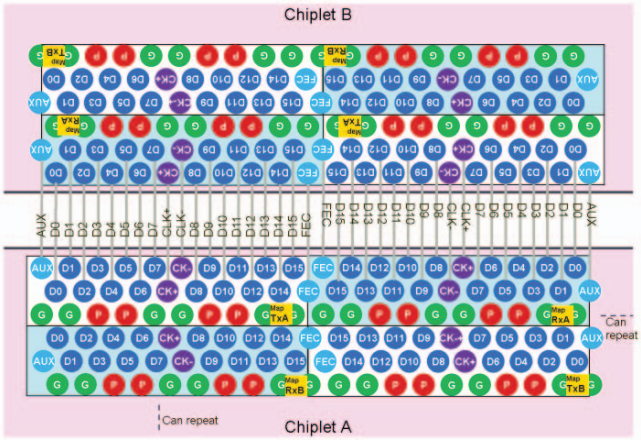


Figure 3: Package wire ordering in a BoW link

A basic BoW slice, configured as Tx or Rx is wired to a corresponding Rx or Tx in the far end chiplet. For a bidirectional link, the preferred arrangement of slices is in a checkerboard of Tx and Rx slices, with Tx wire D0 of slice 0 on the upper left corner viewed from the center of the chiplet and Rx wire D0 of slice 0 on the upper right. The slices nearest the edge on the two chiplets are wired to each other; the stacked slices second-nearest to the edge are also wired to each other, and so on. The current proposal is that on all four edges of a chiplet, the Tx wires and slices are numbered ascending in the clockwise direction and the Rx wires and slices are numbered counter-clockwise. This allows interoperation of chiplets at any n*90-degree rotation.

The *example* bump patterns shown in Figures 2 and 3 are staggered patterns that allow most-compact bump packing. Other arrangements are possible as long as signal/wire ordering is maintained. Most alternatives will use more bump area to optimize other characteristics. The power and ground bumps could be reassigned, or placed in two rows. The signal bumps may be placed in more than two rows to minimize wire pitch. The signal bumps may be square packed instead of hexagonal. Slices may be assembled into asymmetric links or unidirectional links or deeper stacks (4 may be a practical limit for organic substrates) Two chiplets with different stack depths may be connected by configuring the active slices at powerup.

A basic BoW slice that is configurable to operate as either Rx or Tx simplifies packaging and can be used for self-test. The additional circuit area is small compared to the bump array area. In a future extension, full-duplex slices may be supported.

C. *System Logic Integration*

A BoW interface instance will interface with the logic in a chiplet through a MAC or a controller. A transaction protocol controller for protocols such as PCIe, CXL and CCIX will typically contain both the transaction and link layers. The OCP/ODSA aims to reuse existing standard interfaces as an abstraction interface between a D2D PHY and a Controller [2].

Specifically, the ODSA aims to specify the use of the PIPE and LPIF interfaces as abstraction layers for common transaction protocols [6]. A transaction protocol will adapt its transactions to the abstract interface and a BoW interface will implement an adapter to the interface [2]. Figure 4 shows an example configuration where an off-the-shelf PCIe controller can be used for D2D transactions.

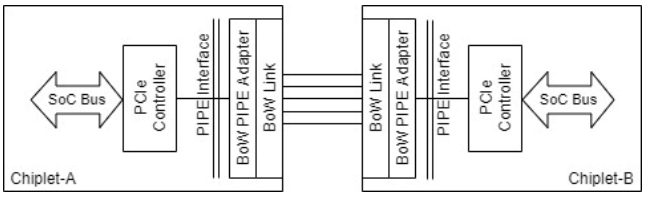


Figure 4: PCIe over BoW

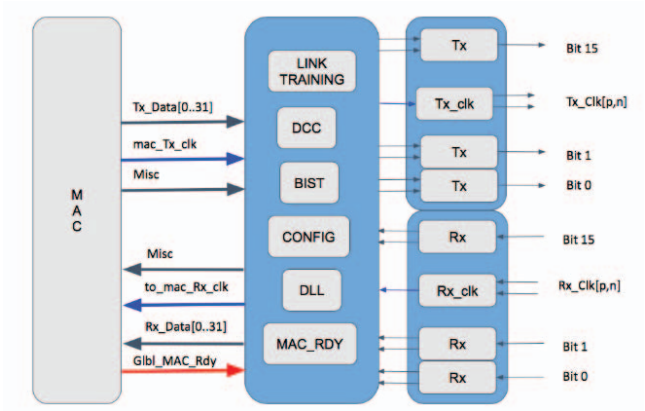


Figure 5: Custom transaction protocol over BoW

The ODSA also aims to implement an open link layer protocol for custom transaction protocols. The data and sideband interface signals between a MAC and the PHY, for a single slice, are shown in Figure 5. The sideband signals, (still to be defined in detail) exchange status information with the MAC for functions such as the calibration within the PHY and readiness to receive/transmit data.

D. Link/System Initialization

When powered on, all the BoW links in a system will need to be initialized, tested and trained to operate. To achieve BoW link ready status with its partner, each chiplet will go through four distinct phases.

- Power-on reset: During power-on reset all I/Os will either be tri-stated with a weak pull down enabled or driven low. Power-on reset does not depend on any configuration specified features.
- Configuration: The PHY is configured according to the commands in a local register. A register map table will be defined to specify the function of each bit (for example, frequency of operation, PoR Status, Config Done, Link Status)
- Calibration: The transmitter will need to calibrate its DCC followed by the receiver calibrating the DLL. Sequencing will require a handshake between the two ends.
- Link Training: The transmitter will send out a known pattern that will enable the receiver to place the receiver clock optimally to capture the data. The tap resolution and de-skew capabilities (per bit or slice level) is left to the designer.

In a system with multiple BoW interfaces, each link pair in the system must achieve link ready status in each of its component slices. Once done, the link has to signal readiness to the rest of the system. If any BoW link is down (either at the MAC or the PHY level), it has to communicate this to the appropriate link partner as well as to the rest of the system.

Calibration and training will require the two endpoints of a link to exchange control information. BoW envisions two options for this exchange. In the first, the two end points communicate directly. For each slice, the AUX bump is used as a MODE bit. When MODE is asserted, the two endpoints exchange control information directly on the signal bumps. BoW will also support an alternative virtual-wire control model. In this model, the status register information is transported over a protocol such as SPI or JTAG. The hardwired model is faster and more responsive than virtual wires. The virtual wire model requires more complex sequencing that can be orchestrated by having one of the chiplets in the system serve as a system master or optionally with an external master control chiplet.

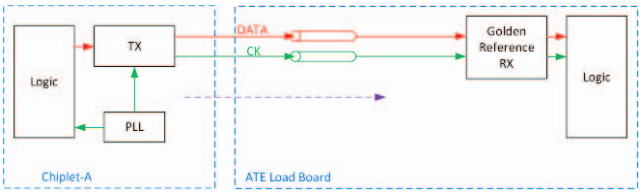
E. BoW Testing

A BoW interface will be used for loopback testing [7] in two use cases: (a) at wafer-sort time for chiplet test; (b) for full-system bring-up and debug validation. Wafer sort tests are currently only practical for the BoW interface with regular bump pitches (~130um), where ATE (automatic testing equipment) probe boards with matching pin pitches are available. Microbump probes will require additional effort [8].

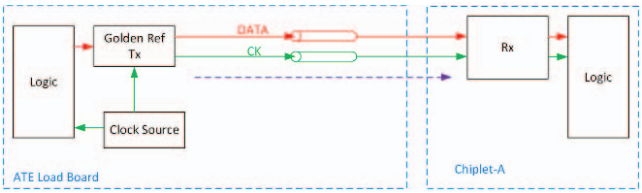
Unidirectional links will need open-loop testing. In Tx-Open-Loop testing, shown in Figure 6a, Chiplet-A transmits a known test pattern to a golden reference receiver through the ATE load board. The received pattern is verified in the ATE load board. Rx-Open-Loop testing, shown in Figure 6b is used for a link where the DUT is only a receiver. A golden reference Tx transmits a known pattern through the channel to the chiplet. The received pattern will be analyzed for quality and functional tests.

With bidirectional links, loopback mode tests can be implemented. In the short loopback mode (Figure 7) data is looped back within the chip. The short loopback can be triggered by the ATE. In the long loopback mode (Figure 8), the PRBS pattern is generated by chiplet-A, sent over the modeled channel on the ATE load board which loops it back. The received pattern will be passed to a bit error rate tester

(BERT) to analyze the performance of the link with off-chip data and clock wires.



(a)



(b)

Figure 6: Open loop testing (a): Tx, (b): Rx

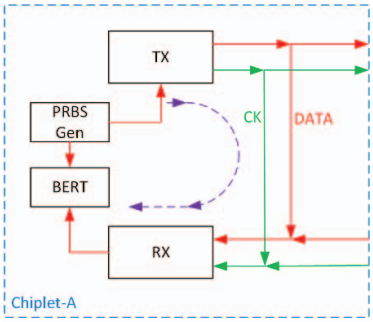


Figure 7: Short loopback testing

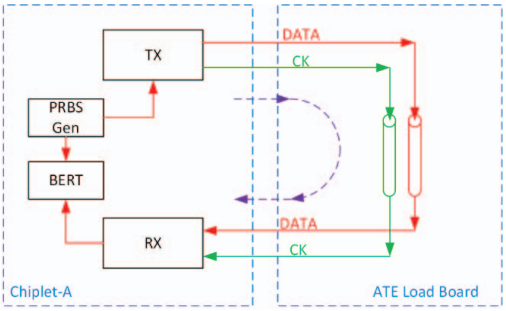


Figure 8: Long loopback testing

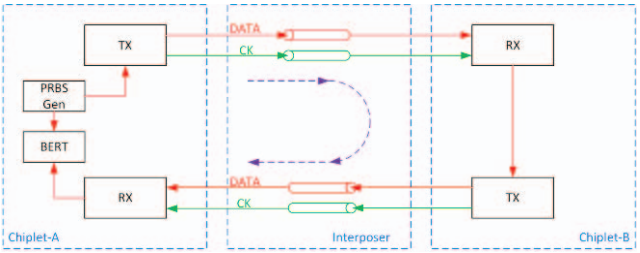


Figure 9: Loopback testing between two chiplets

Both loopback modes can potentially be used for in-field validation bring-up and test. Cooperation across chiplets will be required to execute these tests in the field. Open-loop testing requires the use of a fixed test pattern recognized by both ends and is the only option for unidirectional links. Long loopback mode can be implemented on interposer or organic laminate for validation/verification purposes. Figure 8 shows how a long loopback mode is executed across two chiplets for in-field validation and test.

F. Interoperation with Parallel D2D Interfaces

BoW Basic can potentially interoperate with parallel D2D interfaces such as AIB [9], HBM [10] and Open HBI [2], that use advanced packaging and have been used in chiplet-based products. As an example, Table 2 compares AIB+ and BoW. As can be seen, BoW includes several features to scale across substrates.

In the data plane, a BoW slice can be electrically connected with an AIB Channel with up to 18 signal wires. In the control plane, the two interfaces also share common parameters as is to be expected. The "virtual wire" model discussed above can be used to interoperate with AIB and other parallel interfaces.

TABLE II. AIB-BOW COMPARISON

Parameter	***AIB+***	***BoW Basic***
Dataplane		
Voltage (V)	0.7-0.9	0.7-0.9
Bump Pitch (um)	<=55	<=150
Clock	1 GHz	2 GHz
Data Rate	DDR	DDR
Channel Length (mm)	4	10
I/Os	Unidirectional	Unidirectional
BW per bump (Gbps)	2	4
Basic unit	Channel	Slice
Bits/unit	20	16+2
Bump Map	Specified	Reference
Chip edge orientation	Perpendicular	Parallel
Substrate	Advanced	Laminate/Advanced
Clocking	Differential	Differential
Clock loopback Rx	Yes	Yes
DCC	Yes	Yes
DLL	Yes	Yes
Control plane		
Power on reset	Yes	Yes
NS_MAC_RDY	Yes	Yes
FS_MAC_RDY	Yes	Yes
Sideband control	Yes	Optional
Redundancy	Yes	No
Config done	Yes	Yes
Link training	No	TBD
BIST	No	Yes
Rx termination	No	Optional
Tx termination	No	Optional

FEC	No	Optional
DBI	No	Optional

A custom transport protocol will require a link layer that can be supported by the MACs at either end. In this configuration, link layer control information can also be exchanged through virtual wires. A standard transport protocol such as PCIe, can be adapted to this connection by using a PIPE adapter and standard PCIe controllers at each end as shown in Figure 10. The PIPE adapter for each PHY will be custom to it.

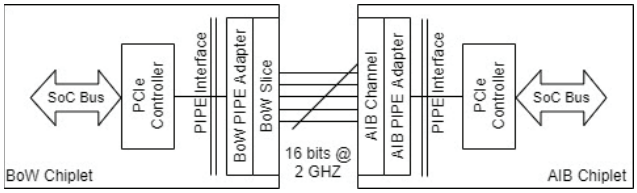


Figure 10: PCIe interoperation over BoW-AIB

The specific packaging technology used will be a function of the bump density in the BoW and AIB chiplets. By defining interoperation, chiplets with BoW interfaces can connect to chiplets that support the AIB interface, such as the Stratix FPGA product line from Intel, providing further economies of scale.

IV. Performance Studies

This section discusses design-performance and package-performance trade-offs with BoW

A. Scaling with Design

Figure.11 is a schematic of the channel topology used to simulate BoW performance. A behavioral model (PWL) of a simple CMOS driver was used. The source termination and driver capacitive parasitics (e.g., ESD, bump) are modeled as simple R, C elements. The receiver is modeled as a high impedance load along with capacitive parasitics.

For package (organic substrate) traces, a 2D field solver is used to extract the RLGC model of a 5-bit bus with typical trace width (25 um) and spacing (25 um) and height (18 um) of above/below reference planes. The nominal impedance for this stripline configuration is ~39 Ohms, using GX92 dielectric material. The model includes crosstalk components for the bus. Each I/O is simultaneously stimulated with an independent PRBS source [n=27-31]: Bit period - 250 ps; Trise and Tfall - 100 ps.

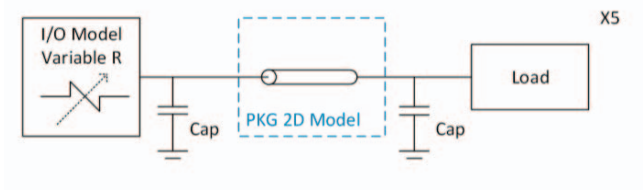


Figure 11: Channel model for simulation

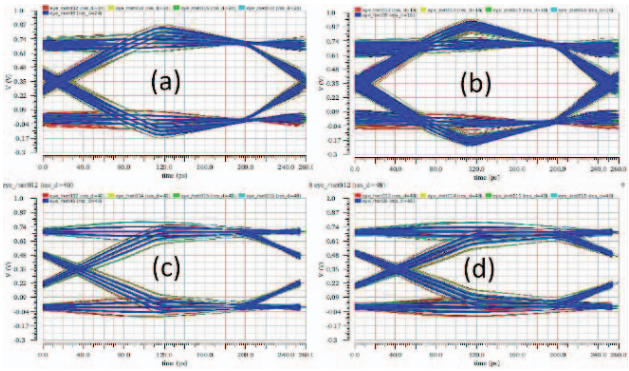


Figure 12: BoW Basic with source termination, Driver and receiver caps are 100fF and load is10kΩ: (a): 24Ω, (b) 16Ω, (c) 40Ω, (d) 48Ω

Figure.12 shows the output of a simulation where the source termination was varied from 16 to 48 Ωs for a 10 mm channel length. A source termination of 16 Ωs produces significant overshoot and undershoots (Top Right), violating the requirements for reliable operation. The best eye, as expected, is obtained when the source termination (40 Ω) matches the impedance of the channel (Bottom Left). To allow for manufacturing variation of +/- 20% on the driver, results for a source impedance of 32 Ωs and 48 Ωs (Middle and Bottom right) are also shown. Even with 20% mismatch in the source impedance, the eye opening is minimally degraded and produces acceptable signal levels at the receiver. The transmitter and receiver parasitics are also varied by 3x, 4x, and 5x w.r.t the previous simulations. The results are shown in Figure 13. Performance is acceptable for 4Gb/s signaling over 10 mm even at 400fF of parasitic capacitance at either end.

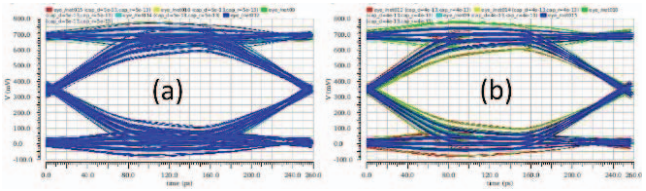


Figure 13: BoW Basic with channel simulation, Source Termination is 40Ω, load is 10kΩ: (a) 500fF, (b) 400fT

We also investigated termination at the far-end with no source termination. As shown in Figure 14, there is little signal degradation even at distances of 50mm of nominal organic substrate routing.

In summary, a simple CMOS driver with source termination can deliver 4 Gbps over 10 mm with standard package materials and trace geometries. These simulations suggest that to achieve the target bandwidth, the I/O design can be kept very simple without having to resort to significant design effort.

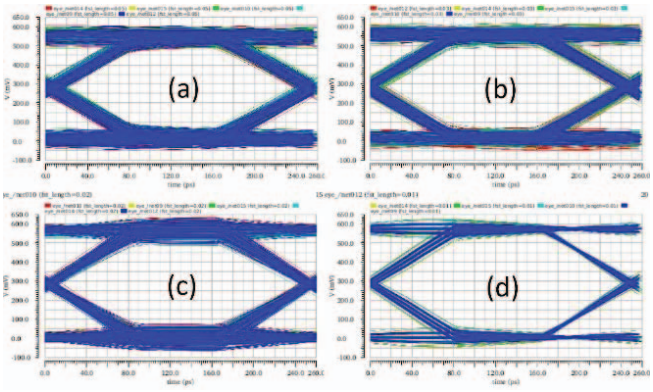


Figure 14: BoW Fast with destination termination, receiver cap is 100fF and load termination is 40Ω, different channel length: (a) 50mm, (b) 30mm, (c) 20mm, (d) 10mm

B. *Scaling with Packaging*

As with other parallel interfaces, the complexity of the IO circuitry can be reduced significantly to fit under fine pitch micro-bumps. This enables dense integration of the BoW IOs for operation over interposer with micro-bumps and dense signal routing. This approach provides a high throughput per beachfront for BoW with simple IO circuitry, but requires expensive interposer technology.

Wafer-level Integrated Fanout (WLFO or InFO) technology, where multiple dice are combined together in a mold and connected using the redistribution layer (RDL), is a lower cost alternative to interposers [5]. The cost of InFO is comparable to organic MCM packaging, as it eliminates the expensive silicon interposer and corresponding complex assembly. It provides about 3x denser routing than organic packages, but 3x-4x less than that of interposer. The routing density is limited to the number of RDL layers that can be fabricated, currently 3 to 4 layers. To achieve similar throughput per edge as in interposers, BoW Fast can be used with higher throughput of 16 Gbps/wire.

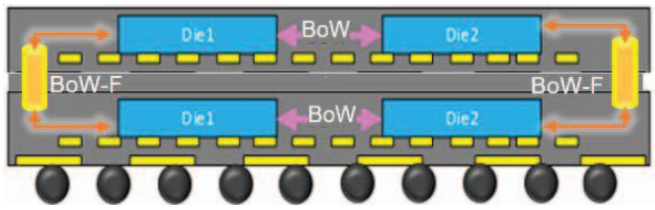


Figure 15: BoW used with advanced packaging

Low-cost 3D packaging can be achieved by stacking RDL-based modules on top of each other and connecting them with Through-Mold Vias (TMVs) as shown in Figure 15. TMVs are considerably lower cost than that of Through-Silicon Vias (TSVs) in interposers. However, TMVs density is again lower by 3x-4x compared to TSVs. To achieve the same throughput as in interposer, BoW Fast at 16 Gbps can be used. To properly operate BoW Fast at 16 Gbaud over the InFO RDL traces, the traces need to behave like transmission lines. By routing reference signals (e.g., ground) next to the signal routes, we can achieve transmission line behaviors over RDLs in the packaging.

C. *Performance Discussion*

The tradeoff between bandwidth density and IP complexity enabled by BoW is evident with further study across package technologies for key data rates. For this comparison, bandwidth density per mm of chip edge and per 0.1 mm^2 of chip area (for better visualization) is evaluated for data rates of 5 Gbps, 8 Gbps and 16 Gbps. The increasing data rates represent increasing complexity of BoW implementation. For the analysis, the bump layout in Figure 16 was used and scaled by bump pitch.

Combinations that may require expensive implementation are included in the analysis but are greyed out, including 16 Gbps/lane implementations at very fine bump pitches. One observation is that nearly equivalent bandwidth density per mm of chip edge is achievable by either implementing a lower data rate BoW (5 Gbps) at a denser 40-um bump pitch or a higher data rate BoW (16 Gbps) with a less-dense 130-um bump pitch, with the latter occupying less overall silicon area. BoW is unique as it enables a choice to invest in package technology or higher performance circuit design to achieve compelling bandwidth density metrics.

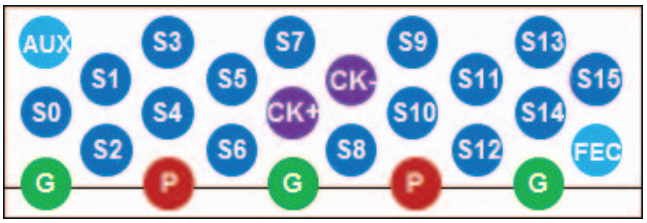


Figure 16: Bump map for performance analysis

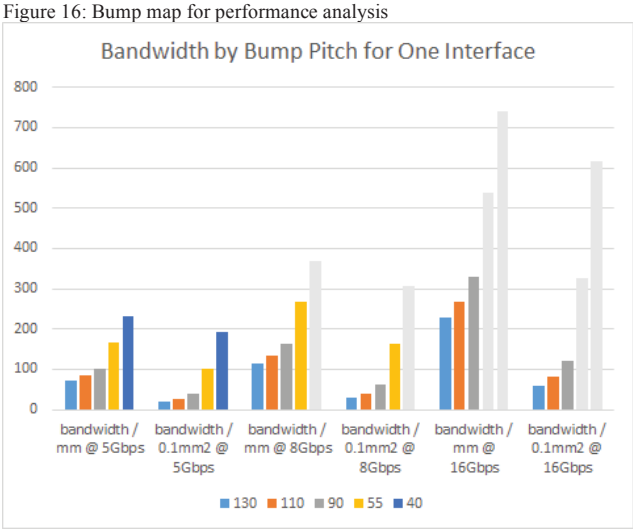


Figure 17: BoW performance comparison

V. Conclusion

Table 3 summarizes progress on BoW attributes against targets set in the initial paper or subsequently.

TABLE III. BoW Status Summary

Parameter	*Original Target*	*Current Status*
BoW Basic/Wire	4 Gbps, 10 mm Unterminated	5 Gbps, >10 mm with source termination
BoW Fast/Wire	16 Gbps, 50 mm Destination Term	16 Gbps, 10 mm Destination Term
Process Nodes	28 nm to 5 nm	65 nm to 5 nm
Packaging	Basic/Advanced	Basic/Advanced
Physical Spec	Bump Map	Wire order
Control/Operation	Mode bit	Virtual wire
Testing	Not defined	Loopback testing
Txn protocol	Standard (e.g.PCIe), Custom protocols	PIPE/LPIF adapter MAC interface

The analyses show that it is possible to meet the objectives of creating a high-performance interface that enables: (a) a performance- cost trade-off; (b) heterogeneous integration; (c) multi-vendor integration. The ODSA also has a set of activities to develop collateral to make the interface easier to use including: (a) a test chip and a packaging prototype; (b) Interface adapters to carry off-package (PCIe, CXL, CCIX) and on-die (AXI, TileLink) protocols on D2D interfaces; (c) A proposal to build a test probe card based on the reference bump pattern.

The open innovation model for the BoW interface has grown active participation and enabled a wide range of use cases to be considered in further developing the BoW. The shareable collateral made possible by an open definition and scalability of the interface will enable economies of scale. The performance, scalability and economics of the BoW interface should make it a strong contender to meet the D2D requirements of a wide range of use cases.

References

[1] R. Farjadrad and B. Vinnakota, "A Bunch of Wires (BoW) Interface for Inter-Chiplet Communication," Hot Interconnect, 2019

[2] ODSA Wiki - https://www.opencompute.org/wiki/Server/ODSA

[3] BoW GitHub repo: https://github.com/opencomputeproject/ODSA-BoW

[4] H. Lee, et al, "Multi-die Integration Using Advanced Packaging Technologies," 2020 CICC, Boston, MA, USA, 2020, pp. 1-7, doi: 10.1109/CICC48029.2020.9075901.

[5] C. Tseng, C. Liu, C. Wu and D. Yu, "InFO (Wafer Level Integrated Fan-Out) Technology," 2016 IEEE 66th ECTC, Las Vegas, NV, 2016, pp. 1-6, doi: 10.1109/ECTC.2016.65.

[6] Intel PCI Express Architecture, https://www.intel.com/content/www/us/en/io/pci-express/pci-express-architecture-devnet-resources.html

[7] A. Loke et al, “Loopback Architecture for Wafer-Level At-Speed Testing of Embedded HyperTransport™ Processor Links, 2020 CICC.

[8] M. Hutner et. al., “Test Challenges in a Chiplet Marketplace,” VLSI Test Symposium, 2020.

[9] Intel AIB Bus: https://github.com/intel/aib-phy-hardware