ESP-JETA
ESP Journal of Engineering & Technology Advancements
ISSN: 2583-2646 / Volume 5 Issue 4 October 2025 / Page No: 58-63
Paper Id: JETA-V5I4P109/ Doi: 10.56472/25832646/JETA-V5I4P109

Original Article

Interface and Partitioning Techniques for Chiplet SoCs

Srikanth Aitha
Osmania University, Hyderabad, Telangana, India.

Received Date: 31 August 2025 Revised Date: 21 September 2025 Accepted Date: 18 October 2025

Abstract: *This rapid generation of semiconductor technologies has caused the models based on chiplets to succeed and the physical constraint of monolithic Systems-on-Chip (SoC) systems. Chiplet-based SoCs encourage modular integration where a design can be compiled of heterogeneous parts which are produced in another technology node. The conceptual basic technologies of the chiplet-based design of SoC, specifically the interface protocols and partitioning, are critiqued in the paper. Certain prominent guidelines of interconnect are Chiplet Interface Protocol (ChIP), Universal Chiplet Interconnect Express (UCIe), and Bunch of Wires (BoW), which we list, how they have been stacked, how the communication was done, and how interoperability issues were managed. The partitioning techniques, which consist of the ChipletPart, CATCH, keeping in mind the trade-offs of the cost-performance, thermal, and design reuse, are also employed. Among the critical ones, the problems of runtime security, power integrity, timing closure, and safety are also listed. The importance of safe inter-chiplet communication processes and the trust which is ensured by the hardware is also accentuated. The paper wraps up its discussion by offering a future outlook of the trends, which are photonic interconnects, computer-aided design partitioning by AI, and standardization that are also being replaced by open chiplet system. The article is part of the research and practitioners in the context of designing scalable, secure, and energy-efficient designs of chiplet-based SoC designs.*

Keywords: *Chiplet-Based Soc Designs, Safe Inter-Chiplet Communication Processes, Runtime Security, Chipletpart.*

I. INTRODUCTION

The increasing intricacy of the sophisticated design of semiconductors and the decreasing size of improvements in the scale of large single-chip design (monolithic System-on-Chip (SoC)) semiconductor design have caused the semiconductor industry to consider an alternative paradigm of chip architecture: the chiplet-based chip design. Chiplets provide a chip design that is modular, and the single-chip monolithic SoC is mounted into small and usable components that can be packaged as a chip. The more superior design approach is more flexible, reusable, yields better results, and reduces time to market. Chiplet-based systems have critical issues in interfaces and partitioning. They exploit high-quality interface protocols to make information exchange seamless between smart partitioning methods in establishing the best chiplet boundaries, performance-cost-versatility tradeoffs. The synergies of these factors are required; without them, we cannot achieve scalable, efficient, and secure chiplet-based SoC solutions.

II. CHIPLET-BASED ARCHITECTURES AND EMERGING NEEDS

The socket-to-chiplet SoCs also recreate the traditional lines of integrated circuit design. Unlike monolithic architecture, chiplet architecture is designed such that functionality is not made in the form of a monolithic die on which all the functionality is implemented, but rather in small dies that are interconnected in a package that holds complex interposers or package substrates. This allows heterogeneous integration, in which distinct chiplets can be manufactured using the most popular preferred technology node and is based upon the performance or power needs [1].

Chiplet-based designs have been defined by numerous merits over the older SoC designs, such as greater silicon yield, reduced cost with large designs, and the ability to intermix IP blocks across vendors or processes. In combination with such advantages, the new design challenges include inter-chiplet communication space, security, partitioning, and embedding the system on a system level [1].

III. SECURITY CONSIDERATIONS IN CHIPLET-BASED SYSTEMS

Security is now in the limelight as chiplet-based architectures have continued to grow, especially in a heterogeneous environment, where the discrete chiplets need not even be from the same vendor, or even equal to other product lines. An attempt to address the issue of runtime security in heterogeneous System-in-Package (SiP) systems has been made by proposing new chiplet roots of trust structures by scholars. The mechanisms can provide chiplet-to-chiplet isolation of security and monitoring, secure boot, access control, and lifetime attestation [2]. The role of secure interconnects, as well as the role of data isolation between chiplets, is also discussed in mobile SoC security architectures. The mobile systems that chiplets should create must ensure that user information is protected and that users are not deprived of the benefits of useful subsystems like cellular basebands or image processors. As a result, the security architectures are constructed upon

hardware-level partitions and secure interfaces to ensure that subsequent movement between the chiplets laterally in the event of a breach is constrained [3]. The technical problem, which makes the chiplets a modular element, also affects the supply chain problem to the extent that the chiplets will require intensive authentication and attestation when integrating them.

IV. CHIPLET INTERFACE PROTOCOLS

The interfaces are of importance in communication protocols since they enable chiplets to communicate using low-latency, high-rate communication. The physically isolated dies (dies-that-died) need to be interconnected with high-speed interrelationships using silicon interposers (2.5D) or vertically-mounted (3D). Effective and successful communication protocols in data provision can coherently synchronize with low overheads. Chiplet Interface Protocol (ChIP) has become a possible standard for applications on an ultra-large scale to meet the growing demands of modularity and interoperability in chiplet-based systems. This protocol is intended to physically and logically isolate communication and provide chiplets the ability to communicate with traditional communication interfaces and hardware-specific execution abstraction [4]. ChIP has flow control, error detection, synchronization mechanisms, and extensibility to support various interconnection topologies. With such properties, a system may be scaled up by adding additional chiplets.

The interface protocols also play a significant role in the chiplet system since they specify the interface between the discrete functional dies in a conventional package. Because the chiplets are independently fabricated and then installed, standard interfaces are necessary to provide interoperability, performance, and design portability in vendor-to-vendor scenarios.

One way to achieve this is the Chiplet Interface Protocol (ChIP), which provides a layered communication infrastructure supportive of ultra-large-scale modular systems [4]. ChIP is associated with definite abstraction layers, examples of which are physical, link, transport, and application. The two layers have the following activities:

- The physical layer handles signal transmission and integrity.
- The link layer ensures reliable communication through error detection and basic flow control.
- The transport layer provides packet sequencing and delivery assurance.
- The application layer manages chiplet-level transaction protocols such as memory reads/writes and command dispatching.

This form of modularity of layering makes chiplets available to be developed and tested separately; nevertheless, there is standardized communication all through the system. The benefits of ChIP are that it can make use of non-homogeneous chiplets, including processors, accelerators, memory controllers, and domain-specific units of logic. It aids in asynchronous communication, i.e., chiplets are free to run other clock domains and organize on the interface level. In addition, ChIP protocol-independent handshaking enables IP reuse and external chiplet integration, which does not involve lock-in. The abstraction itself is applied in migrating to the open chiplet market ecosystems, where reusability, absence of dependence on verification, and reduction of latency are significant concerns. Other standards include UCIe (Universal Chiplet Interconnect Express) and Bunch of Wires (BoW), which are in the development process, but ChIP is a scalable and configurable interface stack that can be operated with 2.5D or 3D integration. Unlike the low power and high throughput-oriented standards such as UCIe, ChIP is protocol modularity-based with a wider range of use.

The following diagram illustrates the layered architecture of the ChIP interface protocol, which supports modular integration across multiple chiplet types:

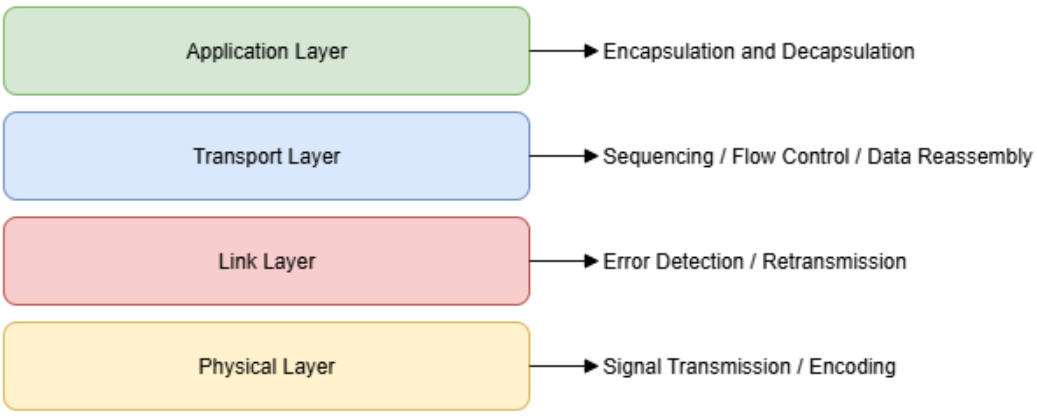

Figure 1 : ChIP Protocol Layered Architecture for Chiplet Communication.

The more developed chiplet architectures with standard interfaces such as ChIP are not only rewarded in the efficiency of design, but also in the scalability of integration solutions, the elimination of integration hazards, and the capacity to maintain heterogeneous integration approaches in the long term.

V. PARTITIONING TECHNIQUES FOR CHIPLET-BASED SOCS

Division of a monolithic SoC architecture into extremely small functional chiplets is called partitioning. It is a physical breakdown, but it is a multifaceted analysis of system-level factors such as power, performance, cost, and area (PPAC). Thoughtful partitioning choices influence thermal, signal, and power flow — all of which are key determinants of the stability of the system's operation.

ChipletPart is a non-3D architecture that implements 2.5D systems but makes partitioning cost-conscious. It implements chiplet design and communicative cost considerations and places packaging and interconnection under consideration. ChipletPart is effective in terms of partitioning by refining probable areas of inter-chiplet communication cost versus local computation — two key areas of partitioning effectiveness [5]. The methodology provides guarantees for keeping latency-sensitive blocks as close as possible, and blocks that require minimal data interactions — which are heavy — would be further aggressively subdivided.

The impact of partitioning on system performance is quantified by performance metrics such as bandwidth, latency, and throughput. Table 1 below shows comparative results from partitioned versus monolithic designs using ChipletPart's evaluation:

Metric	Monolithic SoC	Chiplet-Based SoC (Partitioned via ChipletPart)
Interconnect Latency	1.2 ns	1.5 ns
Peak Bandwidth	250 GB/s	210 GB/s
Power Consumption	85 W	78 W
Thermal Hotspot	110°C	98°C
Design Time	18 months	12 months

Table 1 : Comparative Analysis of Monolithic vs. Partitioned Chiplet SoC [5]

Though some latency penalties are introduced, these are offset by reduced thermal hotspots and improved modularity, enabling faster design cycles.

IV. PERFORMANCE ANALYSIS OF CHIPLET-BASED SYSTEMS

The architectures grounded on chiplets should be compared in terms of their performance with different workloads and settings to complement the partitioning strategies. It is the performance checks that are supposed to be performed on the bandwidth saturation, memory coherence, cache hierarchies, and the execution latency among the chiplets.

Further contemporary work in the chiplet design area has demonstrated that chiplet modularity may be used to permit the most optimal use of area and power allowances, particularly when tasks may be allocated to special-purpose chiplets such as AI accelerators or DSPs. Nevertheless, intercommunication between the chiplets becomes a choke point in case it is not fulfilled in practice [6].

Some of the simulations and empirical results included the fact that task scheduling and memory hierarchy were dependent on chiplet topology. The trade-offs in modularizing the SoCs are reflected in deteriorating performance as the number of chiplets increases, as presented in the graph below:

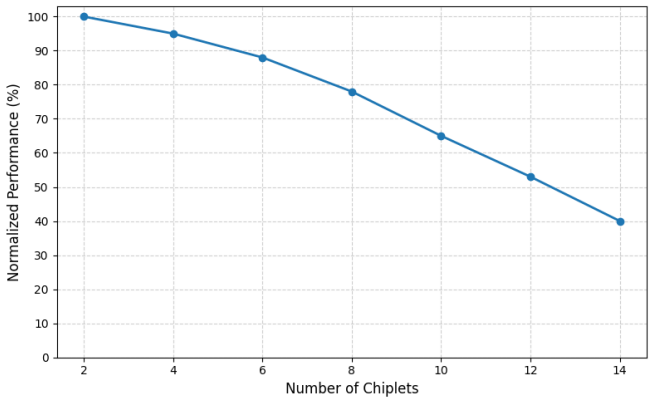

Figure 2 : Performance Decline with Increasing Chiplet Count.

It has also been analyzed that performance reduces significantly when eight chiplets are considered; it is observed to depend on the latency of the communication and the increase in the overheads in synchronization. It implies that the scaling of chiplet systems must be achieved through efficient chiplet architectural co-design and network-on-package optimization.

VII. THREE-DIMENSIONAL SOC DESIGNS

Beside the 2.5D integration system, chiplet systems may also be provided with access to the vertical integration of three-dimensional (3D) SoC designs. The systems are grounded on through-silicon vias (TSVs) as a tool to deliver interconnection between stacked-up dies. This reduces the inter-chiplet communication paths and optimizes bandwidth, which makes the implementation of 3D SoCs suitable for high throughput and low form factor-based applications [7].

The interest in thermal management in 3D SoCs is, however, of significant concern since there is large co-location of active dies. Thermal-conscious partitioning and power delivery networks are currently being developed with consideration of vertical heat dissipation. The next generation of high-performance computing (HPC) and edge computing devices will be replaced by 3D chiplet-based SoCs due to their reliance on TSV capability, as well as heat dissipation.

VIII. SAFETY AND COHERENCE IN AUTOMOTIVE APPLICATIONS

The functional safety and consistency of safety-critical systems that provide the automotive electronics chiplet-based architecture need flexible design strategies to support them. Global standards, including ISO 26262, should govern the combination of various chiplets that may be executing the major functions of sensor data processing, control decision-making, and connectivity.

An automotive-oriented multi-chip architecture provides spheres of safety and safe communication channels that can guarantee that the failure of a single chiplet does not spread and impact the system as a whole. The mechanisms that can be applied in these designs include redundancy and cross-domain error checking mechanisms [8]. It is designed with communication fabrics to ensure consistency of information in the chiplets, particularly in shared memory and sensor data integration applications.

Even more, there is a need to have large combinations of manageable functionality in the heterogeneous chiplets of vehicle systems (from the previous analog regulators to the latest neural accelerators). The partitioning and interfaces must also address the latency requirements of real-time activities; low error and high-integrity data transfer must be guaranteed amid environmental variation, temperature fluctuations, and electromagnetic interference.

IX. TIMING AND POWER INTEGRITY AT SCALE

One of the gravest problems of large-scale chiplet-based SoCs is timing closure and power integrity. The skewness control, voltage variation, and clock distribution are complex, as the chiplets are still physically mounted on an interposer or a package substrate. In the case of high-performance computing (HPC), failure, or even a single malfunction in signal synchronization or the untrustworthiness of the power supply, can be disastrous to the performance of the whole system or even result in a system crash.

The high-performance-based SoCs can take benefit of chiplet partitioning, whereby core compute units, as well as memory interfaces that are coupled with power management units, can be partitioned. Nevertheless, the challenge of incorporating chiplets within the timing constraints lies in the advanced clock tree synthesis method and dynamic voltage scaling techniques, which can only be utilized in distributed systems [9].

Crosstalk parasitics and package sympathy should also be considered by the signal integrity analysis between interfaces of the chiplets. Embryonic tools and methodologies are also available in such a way that co-designing of chiplet SoCs may be conducted at an early stage with a view to determine power and timing limits and subsequently the physical layout. They are practically used to model dynamic profiles of power consumption and ensure the power delivery network (PDN) can provide for the worst-case power consumption without voltage drop or overheating.

X. COST OPTIMIZATION IN CHIPLET SYSTEMS

Inasmuch as fun is part of the technical nature of chiplet-based systems, cost economic sustainability also remains a major concern towards large-scale adoption. In order to achieve the design of cost-effective chiplet systems, the entire lifecycle, IP reuse and production, packaging, and manufacturing shall be considered as one.

This model is termed CATCH (Cost Analysis Tool to Co-optimize Chiplet-based Heterogeneous System), which is used to express the cost level at different levels of abstraction and synthesize design-time decisions and cost impacts [10].

CATCH is a trade-off application that looks into the effects of partitioning choices, packaging complexity, interposer cost, as well as die yield of monolithic and chiplet-based architectures. It enables the architect to design systems in which features that are commonly changed, such as AI accelerators, will be instantiated in the form of chiplets, whereas other non-

varying components, such as memory controllers, are monolithic. It is a technology that impacts area, thermal, and reusability constraints [10].

The design teams will be permitted to have the following scenarios under CATCH:

- It may also be a replacement of a general-purpose processor with partial usage by a special chiplet accelerator.
- Comparison of the various interconnect technologies, including silicon bridges and organic substrates.
- Price calculation for the reuse of the chiplet in the long term with numerous generations of products.

This co-optimization is required particularly when the industry converts to open chiplet marketplaces, where vendor-neutral systems will be implemented to fit third-party chiplets.

XI. TRENDS AND FUTURE RESEARCH DIRECTIONS

Standardization, automation, and sophisticated simulation are the foundations of further development of chiplet-based SoCs. Universal Compute Interconnect Express (UCIe) is the standard around the sector and the one enabling the industry to become compatible at a board level. It is now enabling other vendors' chiplets to be interoperated together in a package. This will enable integration to become very easy, and innovation in chiplet integration using a plug-and-play model would be possible.

One can suppose that additional research will be carried out in the following spheres:

- Machine learning partitioning: AI can be applied to compute the best strategy for partitioning based on the design and performance targets.
- Photonic interconnects: Photonic interconnects connect chiplets with each other, eliminating electrical interposer bandwidth constraints.
- Thermal-sensitive interface protocols: This will involve the development of protocols that dynamically control data transfer rates according to real-time thermal profiles to prevent overheating.
- Security co-design: When designing chiplets, roots of hardware-based trust will be utilized, especially in finance, defense, and automotive systems.

Also, training and education of the workforce will ensure that the potential of chiplet technologies is realized. The general education offered on the production of SoCs will not be useful in the production of chiplet systems since they require expertise in packaging, heterogeneous integration, and cross-domain testing.

XII. CONCLUSION

Chiplet-based System-on-Chip design is an emerging project in semiconductor design and can be scaled, modularly and cost-effectively, to a wide range of consumer electronics, high-performance, and automotive computing applications. The above architectures are implemented successfully on the foundation of high-quality interface protocols and clever partitioning applications, which extract the optimum out of the performance, power, cost, and thermal constraints.

This review calls into perspective the primary contributions and new solutions of chiplet interface and partition methods, which involve security-enhanced protocols, cost-sensitive partitioning algorithms, performance modeling structures, and coherent multi-chip architectures. ChIP, ChipletPart, and CATCH are only examples of the decisive actions addressing the technical and economic issues of chiplet integration.

The trends of the industry will evolve towards the co-optimization of cross-layers, standardization of communication protocols, and holistic design methods that will take into account reliability, safety, and the economic viability of the design step. Along with such developments, the future of computing in edge and cloud architecture will lie in chiplet-based SoCs.

XII. REFERENCES

[1] Mohammed, H. U. H. (2025). CHIPLET-BASED ARCHITECTURES: REDEFINING THE FUTURE OF SYSTEM-ON-CHIP (SOC) DESIGN.

[2] Shuvo, A. M., Sami, M. S. U. I., Zhou, J., Farahmandi, F., & Tehranipoor, M. (2025, May). Enhancing Runtime Security in Heterogeneous System-in-Package Through a Chiplet-Based Root-of-Trust. In *2025 IEEE 75th Electronic Components and Technology Conference (ECTC)* (pp. 2150-2157). IEEE.

[3] Lee, E., Park, K., Lee, J., & Park, D. J. (2025, January). Security Architecture for Heterogeneous Chiplet-Based Mobile SoC. In *2025 IEEE International Conference on Consumer Electronics (ICCE)* (pp. 1-6). IEEE.

[4] Delavari, A., & Vaisband, B. (2025). Chiplets Interface Protocol (ChIP) for Ultra-Large-Scale Applications. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*.

[5] Graening, A., Gupta, P., Kahng, A. B., Pramanik, B., & Wang, Z. (2025). ChipletPart: Cost-Aware Partitioning for 2.5 D Systems. *arXiv preprint arXiv:2507.19819*.

[6] Mallya, N. B., Strikos, P., Goel, B., Ejaz, A., & Sourdis, I. (2025, March). A Performance Analysis of Chiplet-Based Systems. In *2025 Design, Automation & Test in Europe Conference (DATE)* (pp. 1-7). IEEE.

[7] Chakravarthi, V. S., & Koteshwar, S. R. (2025). Designing Three-Dimensional System on Chips (3D SOCs). In *SOC-Based Solutions in Emerging Application Domains* (pp. 39-49). Cham: Springer Nature Switzerland.

[8] Rao, S. S., Thangavelu, B., Joshi, S., Kumar, A., & Srivastava, G. (2025, March). A Safe, Secure and Coherent Multi-chip Architecture for ISO Compliant Automotive Solutions. In *2025 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)* (Vol. 3, pp. 1-6). IEEE.

[9] Ganesan, S. (2025). High-Performance Computing SoCs: Ensuring Timing and Power Integrity at Scale. *Journal of Computer Science and Technology Studies*, *7*(8), 519-525.

[10] Graening, A., Talukdar, J., Pal, S., Chakrabarty, K., & Gupta, P. (2025). CATCH: a Cost Analysis Tool for Co-optimization of chiplet-based Heterogeneous systems. *arXiv preprint arXiv:2503.15753*.