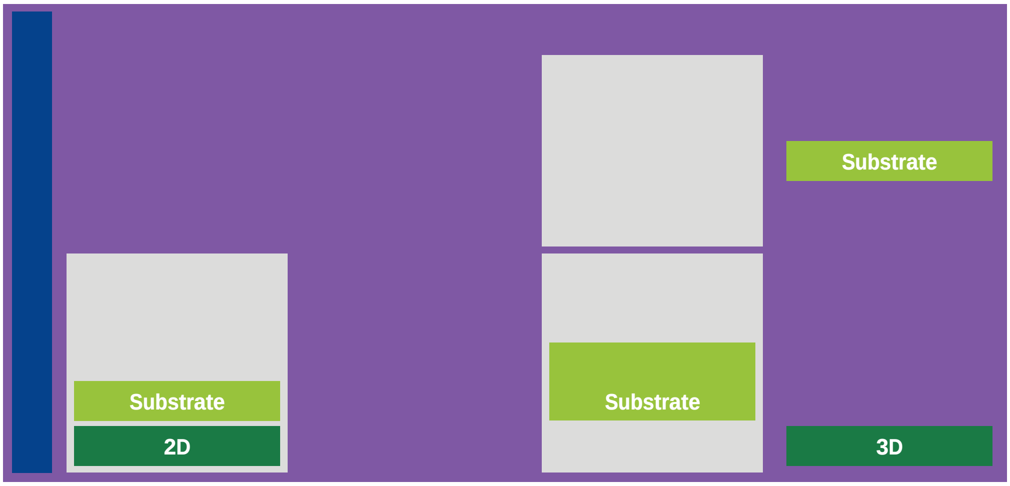
From autonomous cars to surgery-performing robotics, our smart everything world is driving increased and new demands for semiconductors. The unprecedented market shifts brought on by the global pandemic and ensuing supply chain pressures have highlighted chip shortages at a time when users are expecting their electronic products to deliver increasingly sophisticated functionality. Such an environment brings promising opportunities in the electronics industry, with new players entering the semiconductor landscape. What design teams are finding, however, is that traditional, monolithic semiconductor designs are no longer meeting the cost, performance, or functionality needs of certain compute-intensive, workload-heavy applications. Following the path of Moore's law and migrating to smaller process nodes also has its limits.

How can the electronics industry continue as Moore's law slows, system complexity increases, and the number of transistors balloons to trillions?

Multi-die systems have emerged as the solution to go beyond Moore's law and address the challenges of systemic complexity, allowing for accelerated, cost-effective scaling of system functionality, reduced risk and time to market, lower system power with increasing throughput, and rapid creation of new product variants. For applications like high-performance computing

First, let's define exactly what we mean by a multi-die system. Simply put, a multi-die system is a massive, complex, interdependent system comprised of multiple dies, or chiplets, in a single package. There are different approaches to creating this type of architecture. One approach consists of disaggregation—the partitioning of a large die into smaller dies to improve system yield and cost compared to monolithic dies. The disaggregated approach applies to heterogeneous designs as well as homogeneous designs.


Substrate
Substrate
2D
Substrate
3D

Die-to-die interfaces are integral in bringing multi-die systems to life. Consisting of a physical layer (PHY) and a controller block, they provide the data interface between two silicon dies assembled in the same package. Disaggregated chips rely on several die-to-die connectivity architectures that support high data rates, which is why UCIe stands out here. Other key characteristics for die-to-die interfaces include:

- Modularity
- Interoperability
- Flexibility
- High bandwidth efficiency
- High power efficiency
- Low latency
- Robust, secure known good dies
- Short-reach, low-loss channels without any significant discontinuities

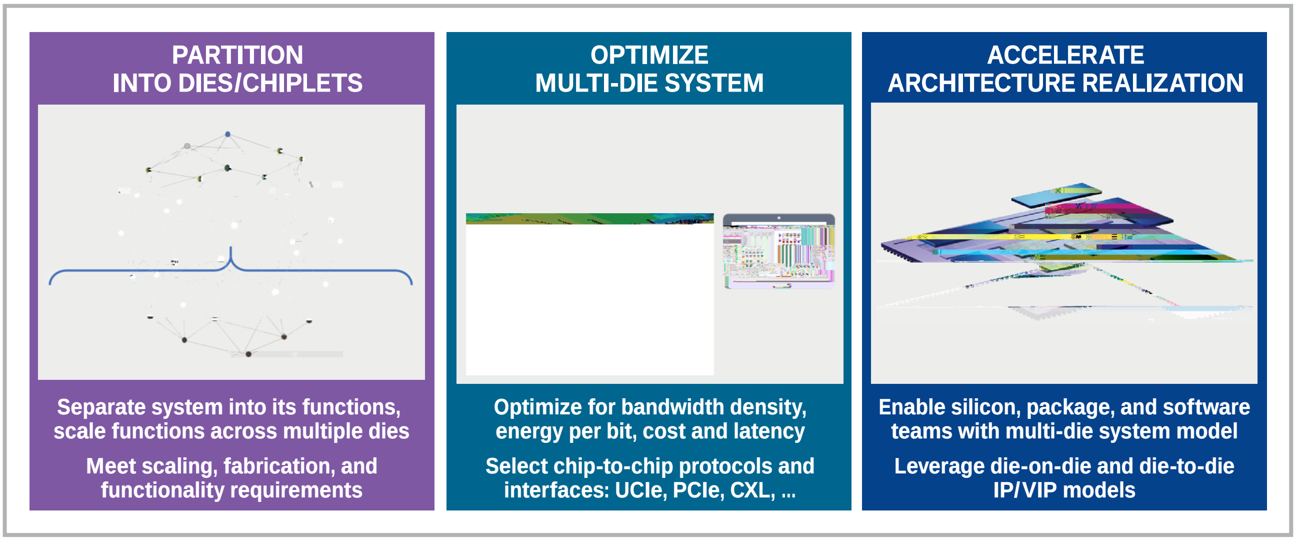
Die-to-die controller and PHY IP can help ensure that the interfaces are designed to deliver on these criteria. Controller IP with error

Traditional chipmakers aren't the only ones getting into the multi-die system space. Hyperscalers with their massive data centers, carmakers developing autonomous functions, and networking companies are among the businesses that are designing their own

The design starting point, architecture exploration, must take an analysis-driven approach that considers macro-architecture decisions such as IP selection, hardware/software partitioning, system-level power analysis, and interconnect/memory dimensioning. Additionally, there are multi-die macro-architecture decisions pertaining to aggregation (assembling the system from dies) and disaggregation (partitioning the application onto multiple dies).

To understand the questions that must be answered during this phase, consider a complex application like a hyperscale data center. How many die of each type would be needed, on what process nodes should they be on, and how would they be connected? For each die, how would the functionality of different subsystems be partitioned into local processing elements? How would the system, with its different memories and compute dies, be assembled? Even if you've assured that the dies are correctly designed, how do you ensure that the entire system will meet your power and performance targets once it has been assembled? An analysis-driven approach would allow you to iterate through your many choices early to optimize your multi-die system as well as costs.

For safety-critical applications like automotive, predictability is an important criterion. Ultimately, a data-driven architectural specification approach utilizing modeling, analysis, simulation, and experimentation will guide the way.



PARTITION
INTO DIES/CHIPLETS

Separate system into its functions, scale functions across multiple dies

Meet scaling, fabrication, and functionality requirements

OPTIMIZE
MULTI-DIE SYSTEM

Optimize for bandwidth density, energy per bit, cost and latency

Select chip-to-chip protocols and interfaces: UCIe, PCIe, CXL, ...

ACCELERATE
ARCHITECTURE REALIZATION

Enable silicon, package, and software teams with multi-die system model

Leverage die-on-die and die-to-die IP/VIP models

Early architecture decisions on several key areas can enhance the design process:

- Multi-die system partitioning into dies to optimize chip-to-chip traffic
- Chip-to-chip communication considerations to ensure effective throughput and latency
- Trade-offs between interface power consumption, throughput, and die placement
- Performance impact of different fabrication and packaging technologies
- Die-to-die protocols and interfaces

Aside from making these early architecture decisions, engineering teams must also address chip-to-chip performance bottlenecks. Modeling latency and performance based on partitioning and die-to-die interface choices can help here. Finally, the other big challenge is to meet power and thermal key performance indicators (KPIs) by addressing system power consumption as well as the thermal impact of multiple dies in one package.

What's helpful to know is that automation available in today's tool flows has elevated architecture exploration beyond the manual, spreadsheet-based predictions of years past. Looking ahead, unified design space exploration could further elevate the accuracy and productivity of this process.

An easier and more productive transition from 2D to 2.5D/3D designs will benefit from consistent data management across dies and technologies. This is where a disjointed flow consisting of point solutions can be particularly detrimental to outcomes as well as productivity. To address the unique requirements of multi-die systems, what's needed is a unified approach for die/package co-design that spans design, analysis, and signoff. Ideally, an integrated environment should:

- Provide the integration capacity and efficiency for >100s of billions of transistor connections
- Support faster design closure via a concurrent workflow at all stages of the design, as well as a common data model and database with common tech files and rules
- Foster productivity with a single software environment and GUI for multi-die/package co-design
- Deliver fast convergence on optimal PPA, while accelerating time to package
- Optimize the design—and costs—early on and at a system-wide level

When it comes to validation, it's much too simplistic to consider a multi-die design as being a much larger system than an SoC. It is, but effectively emulating very large systems brings capacity into question. Multi-die systems also tend to be heterogeneous, with dies developed on different process nodes and, in some cases, reused, limiting access to any proprietary RTL.

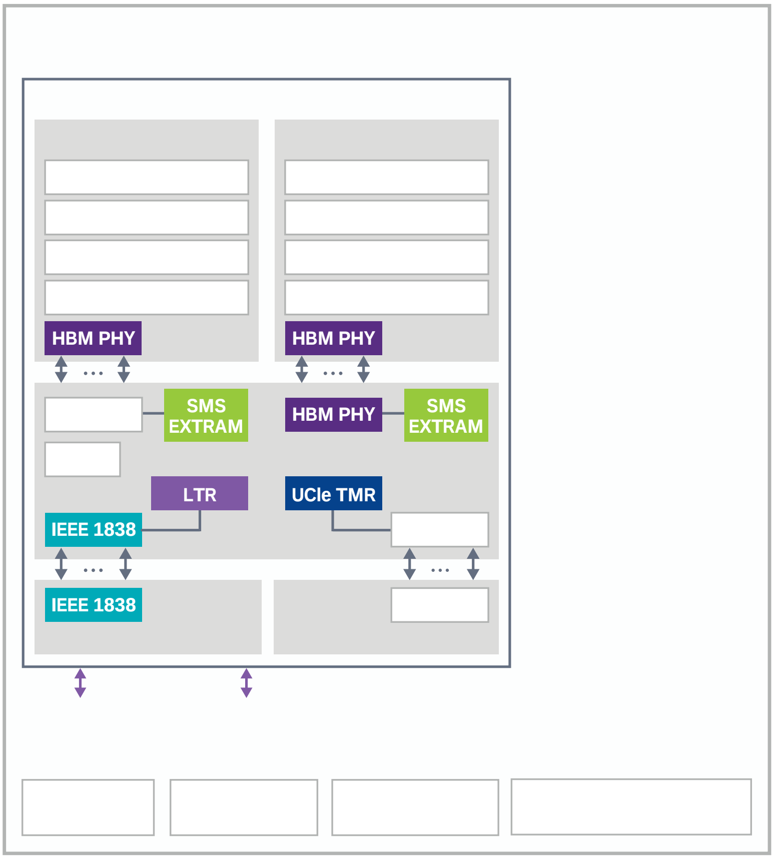
For multi-die software development and software/hardware validation, there are a few key considerations and solutions:

- Software bring-up of one die with software dependency on other dies. Multi-abstraction system modeling can leverage fast, scalable execution platforms that make use of virtual prototyping and hardware-assisted verification.
- Validation of the die-to-die interface. Pre-silicon validation can take advantage of IP blocks verified and characterized using analog/mixed-signal (AMS) flows. Pre-silicon validation and compliance testing can also be handled via a UCIe controller IP prototype with a UCIe protocol interface card.
- Multi-die system software/hardware validation. Each die can be mapped onto its own emulation setup and connected via die-to-die transactors (UCIe, etc.). Realistic application workloads executed with hardware-assisted verification can yield insights into multi-die system performance and support fast turnaround time on power validation. A die under test can also be connected via a speed adaptor to prototypes of mature dies.

Let's dive in a little deeper on these points. Given such a complex system running very complex software, it's essential to start the validation process early, creating virtual prototypes of the multi-die system to support software development. Specifying system behavior up front with a virtual model, running software on that model, allows the system specs to become more solidified and the software to become better defined before emulation.

In multi-die systems, it's important to optimize the die-to-die connections at the protocol level (the digital parts) and the analog level (the PHY). AMS emulation helps to reduce the risks that something will go wrong post-silicon.

Heterogeneous setups can facilitate validation of multi-die systems. Consider a design consisting of three dies developed by one semiconductor supplier, who provides the RTL, and a fourth die from another supplier, without RTL access but with an existing die. The three dies with RTL can be emulated in a large-scale setup with idatibom anotuC BT (or lex softwar)yst[Multi-die sy.)]TErb tom (TL can b



Silicon health can also be evaluated via silicon lifecycle management (SLM) technology. SLM involves integrating monitors into components of the design to extract data throughout a device's lifecycle, even while it's in the field. The deep, actionable insights gathered from silicon to system allow for continuous analysis and optimization.

With multi-die systems, the monitoring infrastructure should be unified across multiple dies. The idea is to capture a profile of environmental, structural, and functional conditions throughout the lifecycle of the chip. The challenges lie in complexity-driven reliability, power management, and interconnect concerns.

Given the system interdependencies, design teams will need to know, for instance, where to place two dies with very different thermal characteristics so that the heat dissipating from one die won't negatively impact the operation of the other—or that of the system. Once in the field, chips are affected by aging and temperature, making continuous monitoring a valuable function. Access to the individual dies once they've been packaged is also more challenging in the disaggregated world. If dies are stacked vertically, for example, there needs to be an efficient way to access them for in-field characterization.

SYNOPSYS®