Technical Disclosure Commons

Defensive Publications Series

October 2020

Three-dimensional Integration of Compute Core and I/O in High-performance ASIC

N/A

Follow this and additional works at: https://www.tdcommons.org/dpubs_series

Recommended Citation
N/A, "Three-dimensional Integration of Compute Core and I/O in High-performance ASIC", Technical Disclosure Commons, (October 05, 2020)
https://www.tdcommons.org/dpubs_series/3655

This work is licensed under a Creative Commons Attribution 4.0 License.
This Article is brought to you for free and open access by Technical Disclosure Commons. It has been accepted for inclusion in Defensive Publications Series by an authorized administrator of Technical Disclosure Commons.

Three-dimensional Integration of Compute Core and I/O in High-performance ASIC

ABSTRACT

In certain ASICs, the input-output (I/O) interfaces, such as high-bandwidth memory (HBM) physical layer (PHY), serdes, etc., occupy substantial area on the die. Validation of I/O interfaces fabricated in recent process nodes, e.g., 3 nm technology, is generally more involved and has corresponding time to market costs. For high-performance ASICs whose compute function is divided amongst cores, communications between an HBM and the core farthest from it can be complex. This disclosure describes techniques to partition the compute cores and the I/O interface into separate dies and to implement these in suitable process nodes which may be different, e.g., compute core in 3 nm technology and I/O interface in 7 nm technology. By doing so, the time to validate the I/O interface is reduced. Additionally, communication lines between an HBM and a compute core far away from each other are simplified. Another benefit is that the area of compute core dies can be maximized due to no other I/Os.

KEYWORDS

- Compute core
- I/O interface
- High-bandwidth memory (HBM)
- HBM PHY
- PCIe
- GPIO
- Chiplet
- Die-to-die (D2D) interconnect
- Through silicon via (TSV)
- Active interposer

BACKGROUND

In certain high-performance application specific integrated circuits (ASICs), input-output (I/O) interfaces such as high-bandwidth memory (HBM) physical layer (PHY), Serializer/Deserializer (serdes), PCIe, general purposes input/output (GPIO), etc. occupy

substantial area on the die. The I/O interfaces, which are an overhead to the main compute function of the ASIC, don't scale well, especially in the recent process nodes, e.g., 3 nm or other recent technology. Design, development, and Silicon validation of I/O interfaces fabricated in recent process nodes is generally more involved which imposes a corresponding time to market cost. Another feature of high-performance ASICs is that the compute function is divided amongst two or more cores to get a higher die count per wafer with better wafer yield. However, this can complicate communications between an HBM and the core farthest from it, as explained below.

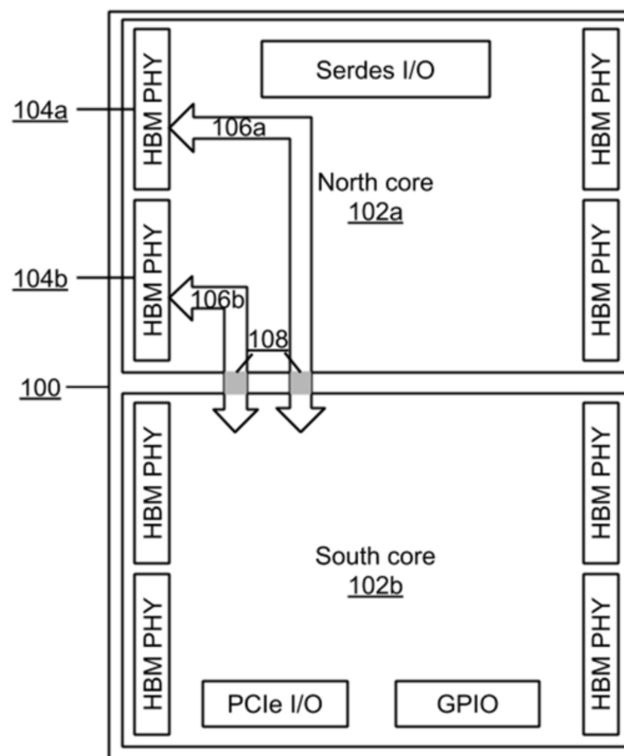

Fig. 1: Illustrating the complexity of communication between an HBM and the core farthest from it

Fig. 1 illustrates an example of the complexity of communication between an HBM and the farthest core. There are two cores, referred to as north core (102a) and south core (102b) within a traditional ASIC (100). The I/O interfaces are integrated within the ASIC as illustrated,

e.g., HBM PHYs (104a-b) at the eastern and western edges of both cores, a serdes at the north core, and a PCIe and a GPIO at the south core. Communications between the south core and HBM PHY located in the north core (106a-b) traverse the length of the north core which presents a trace-route complexity. There also exists a crossover between the dies using a high-bandwidth die-to-die (D2D, 108) interconnect which is another source of complexity.

DESCRIPTION

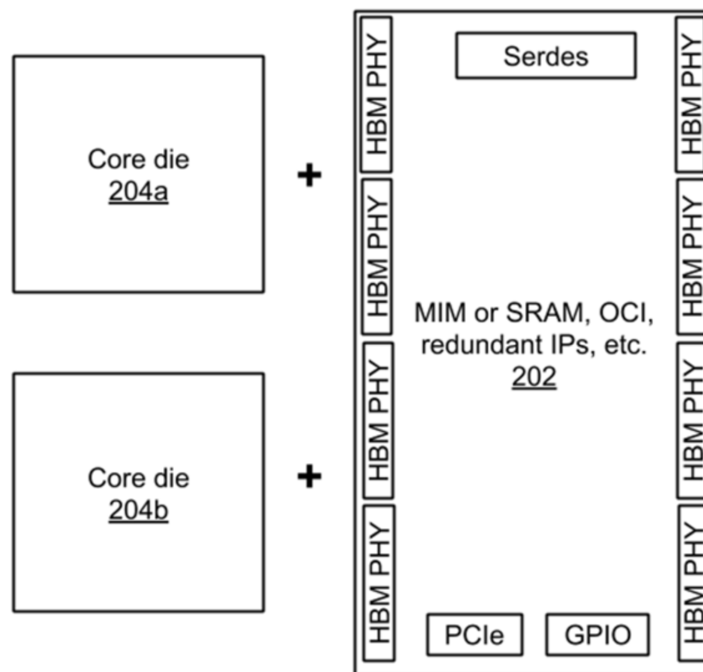

Fig. 2: Separation of the I/O interfaces and the compute cores into distinct chiplets

Per the techniques of this disclosure, illustrated in Fig. 2, the I/O interfaces and the compute cores are partitioned into distinct dies, or chiplets. The smaller, denser, scale-oriented compute core dies (204a-b) are designed and manufactured in a more recent process node, e.g., 3 nm or other technology, while the I/O interface (202) leverages an older but more thoroughly tested process node, e.g., a 7 nm, 10 nm, 14 nm, 16 nm, or other suitable technology. The I/O interface chiplet can include several types of I/O interfaces, e.g., HBM PHY, PCIe, GPIO,

serdes, OCI/repeaters, etc. The compute cores can each be of a suitable size that is smaller than the I/O interface. As such the I/O interface die has redundant space, which can be utilized for SRAM, metal-insulator-metal capacitors (MIMcaps), OCI, or redundant silicon IP to improve wafer yield. By separating the compute cores from the I/O interface and by using different process technology for the compute cores and the I/O interface, the techniques reduce validation effort on I/O interfaces coupled with compute cores, thereby reducing time-to-market.

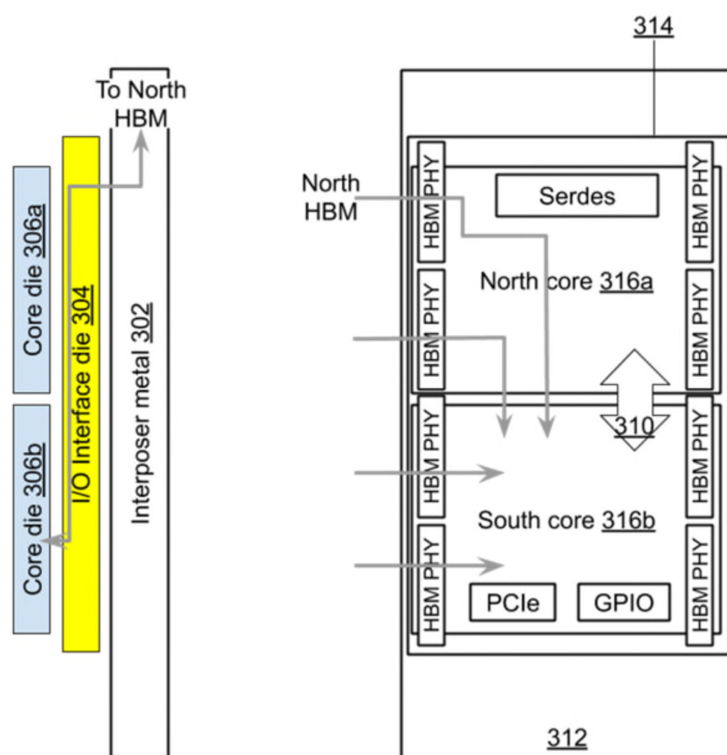

Fig. 3: (a) Side view, and (b) top view of a stacked package comprising the core and I/O interface dies and the interposer

Per the techniques, a package is constructed by stacking the compute core and interface dies atop interposer metal to achieve a three-dimensional structure. A cross-sectional side view of the package is illustrated in Fig. 3(a), in which the compute cores (306a-b) are placed next to each other and atop the I/O interface die (304), which is itself atop interposer metal (302). A top view of the package is illustrated in Fig. 3(b), which again shows the compute cores (316a-b)

next to each other. Although the I/O interface (314) is below the compute cores, for clarity the different types of I/O interfaces, e.g., HBM PHY, PCIe, GPIO, serdes, etc. are illustrated as being visible. The I/O interface itself rests on interposer metal (312). Core-to-core data transfer occurs as before (310).

In both Fig. 3(a) and 3(b) the grey arrows represent communication lines between an HBM and a core. Per the techniques, an HBM is coupled to a compute core via datapath lines that pass through the I/O interface die, e.g., without traversing any compute core. Unlike a traditional design, this is true even for an HBM, e.g., the north HBM, far away from a compute core, e.g., the south core. Unlike a traditional design, the disclosed techniques enable low-complexity HBM-core communication, with routes that do not traverse a compute core and lines without D2D interconnect. The HBM PHY and the on-chip interconnect in the I/O interface die can serve multiple remote-access lines between HBMs dies and compute core dies without D2D IP in the compute core die.

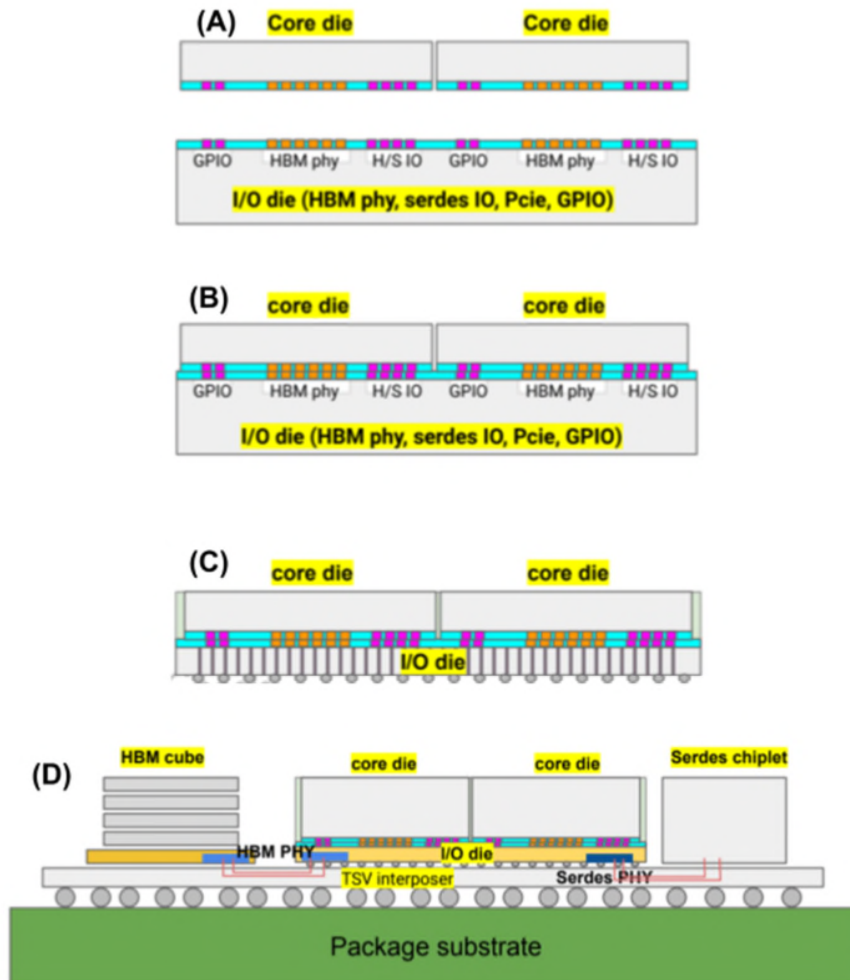

Fig. 4: Physical layout of a package during its manufacturing stages

Fig. 4 illustrates the physical layout (side cross-sectional view) of a package comprising distinct compute core and I/O interface dies during various manufacturing stages. Fig. 4(A) illustrates two compute dies just prior to stacking with an I/O interface die. Fig 4(B) illustrates the compute cores and the I/O interface after stacking. The compute cores and the I/O interface dies are bonded together with a fine-pitch hybrid bonding, which offers high bandwidth and low latency compared to other types of bump bonding. Fig. 4(C) illustrates a fully-stacked die, e.g., the stacked cores with through-silicon vias in the I/O die, which connect to C4 solder bumps.

Fig. 4(D) illustrates the complete package (system-on-chip), e.g., with compute cores, I/O interface, high-bandwidth memory cube, serdes chiplet, etc. The two cores are mounted atop the I/O interface, which is itself mounted atop a through-silicon via (TSV) interposer. The HBM cube and the serdes chiplet are also mounted atop the TSV interposer. The TSV interposer rests on a package substrate. The three-dimensional stacked dies are thinned and bumped for assembly on the substrate. As explained before, datapath lines connecting the compute cores to the HBM cube and the serdes chiplet run through the TSV interposer and through the I/O interface, avoiding traversal through the core(s).

CONCLUSION

This disclosure describes techniques to partition the compute cores and the I/O interface into separate dies and to implement these in suitable process nodes which may be different, e.g., compute core in 3 nm technology and I/O interface in 7 nm technology. By doing so, the time to validate the I/O interface is reduced. Additionally, communication lines between an HBM and a compute core far away from each other are simplified. Another benefit is that the area of compute core dies can be maximized due to no other I/Os.