A case for three-dimensional stacking of tightly coupled data memories over multi-core clusters using low-latency interconnects

Article *in* IET Computers & Digital Techniques · September 2013

DOI: 10.1049/iet-cdt.2013.0031

CITATIONS
7

READS
927

3 authors, including:

Erfan Azarkhish
Centre Suisse d'Electronique et de Microtechnique
13 PUBLICATIONS **134** CITATIONS

SEE PROFILE

Luca Benini
University of Bologna
1,157 PUBLICATIONS **35,548** CITATIONS

SEE PROFILE

Published in IET Computers & Digital Techniques
Received on 15th February 2013
Revised on 15th May 2013
Accepted on 28th May 2013
doi: 10.1049/iet-cdt.2013.0031

ISSN 1751-8601

A case for three-dimensional stacking of tightly coupled data memories over multi-core clusters using low-latency interconnects

Erfan Azarkhish, Igor Loi, Luca Benini

DEI, University of Bologna, Bologna, Italy
E-mail: erfan.azarkhish@unibo.it

Abstract: Shared tightly coupled data memories are key architectural elements for building multi-core clusters in programmable accelerators and embedded systems, as they provide a convenient shared memory abstraction while avoiding cache coherence overheads. The performance of these memories largely depends on the architecture of the interconnect used between processing elements (PEs) and memory banks. The advent of three-dimensional (3D) technology has provided new opportunities to increase design modularity and reduce latency and manufacturing cost. In this study, the authors propose two 3D network architectures: C-logarithmic interconnect (LIN) and Distributed logarithmic interconnect (D-LIN) (designed in synthesisable RTL), which allow modular stacking of multiple L1 memory dies over a multi-core cluster with a limited number of PEs. The authors have used two through-silicon-via technologies: the state-of-the-art micro-bumps and the promising and dense Cu–Cu direct bonding. The overhead of electrostatic discharge protection circuits has been considered, as well. Architectural simulation results demonstrate that, in processor-to-L1-memory context, C-LIN and D-LIN perform significantly better than traditional network-on-chips and simple time-division multiplexing buses. Furthermore, post-layout results show that the proposed 3D architectures achieve comparable speed against their 2D counterparts, whereas enabling modularity: from 256 kB to 2 MB L1 memory configurations with a single mask set.

1 Introduction

The increasing focus on energy-efficient architectures coupled with a slowdown in clock speed improvement has created a growing interest in parallel computing, where a large number of simple cores are integrated onto the same die. General purpose graphics processing unit (GP-GPU) such as NVIDIA Fermi [1], HyperCore [2] and ST-microelectronics platform 2012/STHORM [3] are the most visible examples in this trend. All of these architectures follow cluster-based many-core designs with a limited number of processors (up to 32) in each cluster sharing tightly coupled L1 data memories (TCDMs), a.k.a scratchpad memories. TCDMs are used because they yield much higher storage density per unit area, lower power consumption and lower access latency compared with cache memories [4].

Network-on-chips (NoC) designs have been advocated as an alternative to bus-based architectures. Thanks to their scalability which makes them suitable for inter-cluster communication and in L2 and upper levels of memory hierarchy. However, their high average latency and latency variability, and increased design complexity to guarantee correctness and fairness (e.g. avoiding deadlock, livelock and starvation) [5] brings their usefulness in processor-to-L1-memory context under question. On the other hand, crossbar-based interconnects, like the one in IBM BlueGene/Q [6], can provide a uniform and ultra-low memory access latency within a cluster, which is unachievable in multi-stage NoC systems. The design of crossbar networks for high-performance usually relies on custom circuit design techniques, such as pass transistors and low-swing drivers (e.g. [7]). Full-custom approaches are not suitable for architectures featuring soft cores and third-party intellectual property (IP) blocks, and their reusability is limited across technology nodes. Therefore processor-to-L1-memory interconnects provided as a parametric synthesisable IP are highly desirable in this context.

In this paper, we take advantage of three-dimensional (3D) technology to increase the shared L1 memory size in a modular fashion, that is, stacking memory dies on top of a logic die, without the need to re-spin silicon [as it would be needed for traditional 2D technology]. We focus on a single cluster (see Fig. 1*b*) with a typical size [16 processing elements (PEs)] sharing a tightly coupled multi-banked L1 memory, and propose two 3D network architectures, Centralized 3D-LIN (C-LIN) and Distributed 3D-LIN (D-LIN) (designed in synthesisable RTL), which can be configured based on user specifications and technology constraints to provide low-latency memory access. Our modular stacking strategy allows system integrators to stack multiple memory dies and create arbitrary L1 memory sizes through different height stacks with identical dies, without

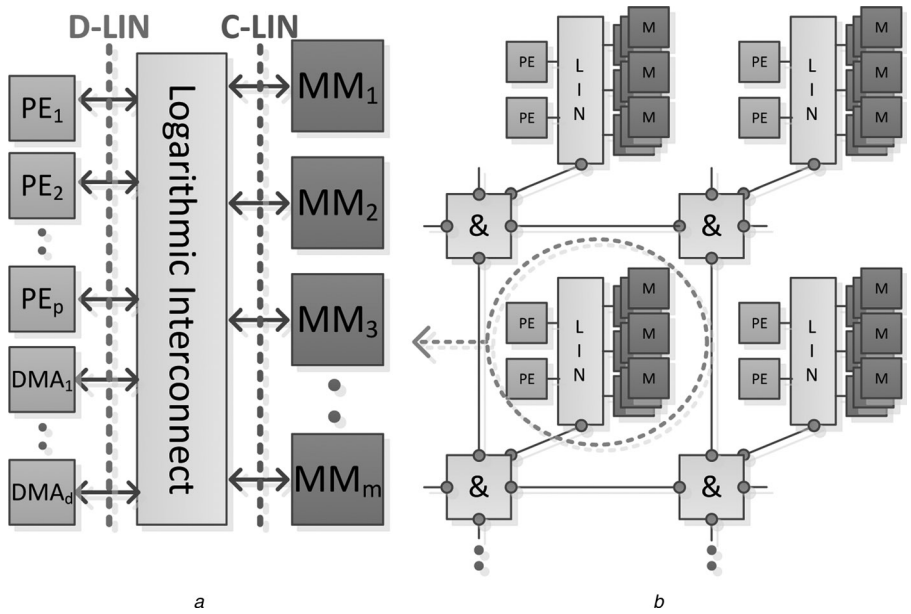

Fig. 1 *Logarithmic Interconnect (LIN) and its usage inside a many core platform*

a Abstract view of the Logarithmic Interconnect (LIN)
b LIN inside a cluster-based many-core platform

the need for different masks for dies at different levels in the stack. The designs have been implemented using two bonding techniques with consideration of the electrostatic discharge (ESD) protection circuits.

Architectural simulation results demonstrate that in the processor-to-L1-memory context, our proposed LIN-based designs outperform traditional NoC and simple time-division multiplexing buses. Furthermore, post-layout results on realistic floorplans and show that the proposed 3D architectures increase design modularity over their 2D counterpart (2D-LIN), and provide the opportunity for further cost optimisations, whereas achieving comparable speed. Thus, even though current through-silicon-vias (TSVs) are still not much better in terms of speed than global on-chip wires, they can provide more freedom in heterogeneous integration of dies with cost-optimised technologies, since they are definitely much better than traditional off-chip links. Related research efforts are presented in Section 2. 2D-LIN and its 3D extensions are described in Sections 3 and 4. Issues related to the 3D integration are discussed in Section 5, and finally, experimental results and concluding remarks are brought in Sections 6 and 7.

2 Related works

Performance limitations of the interconnection networks have led to a renewed interest in interconnect research and a transition from traditional bus-based systems to more sophisticated topologies, including mesh NoCs [8], hierarchical bus models [9], flattened butterfly on-chip networks [10] and crossbars [11–13]. The ability of crossbars to provide uniform access latency makes them an appealing option in processor-to-L1 memory interface for limited-cardinality clusters (16 PEs, typically), because predictable access latencies allow for quality-of-service guarantees and ease of programming. Custom designed crossbar-switches can provide very high bandwidths (e.g. 1 Tbit/s in [7]); however, lack of configurability and their incompatibility with standard technology libraries provided by silicon foundries make them unsuitable to system on chip (SoC) design.

Moreover, crossbar networks for tightly coupled shared memories have been used in the HyperCore architecture [2], which contains a shared on-chip memory accessed through a series of combinational switches; and in [11, 12] using mesh of trees (MoTs) and swizzle switch networks, respectively.

3D stacking of scratchpad memories to replace fast on-chip secondary random accesss memories (SRAMs) has been studied in [14, 15]. In [14], the authors proposed a configurable memory layer that consists of many uniform memory elements connected directly to processors. Although, in [15], a prototype of 3D stacked TCDM has been published, which is a two-layer 3D IC, with logic die consisting of 64 general-purpose processor cores running at 277 MHz and connected through a mesh NoC, and the memory die with 256 kB of SRAM. While these works have simply focused on the use of private memory banks, our work proposes a solution for sharing L1 memory, and shows that NoC solutions are not suitable in this context.

In addition, 3D extension of low-latency crossbars for shared L1 clusters has been investigated in [11, 13], whereas [16] uses time-division multiplexing buses for this purpose. The key difference between our proposed approach and these works is modularity, which allows stacking of several memory dies on a logic die without the need for new masks for each stacked die. Moreover, our solutions offer better scalability compared with [13] (more in-depth discussion is performed in Sections 4 and 6.2). Additionally, physical synthesis on realistic 3D floorplans make our obtained results more accurate.

Lastly, several vertical interconnect technologies have been explored in the literature, including wire bonding, micro-bump, contactless (capacitive or inductive) and TSV vertical interconnect [17]. Among them, the TSV approach offers the best vertical interconnection density, and thereby, has gained popularity. Among the TSV technologies MIT Lincoln Laboratory (MITLL) [17] and Tezzaron TSV [18] offer high density (over 15 000 via/mm^2) and low resistance

doi: 10.1049/iet-cdt.2013.0031

($< 0.5\ \Omega$) and capacitance (< 2 fF); however, their number of stacked layers is limited to three and two, respectively, and they are used in technologies larger than 90 nm and in low-volume production. The current state-of-the-art in high-volume production-ready TSV technology uses more conservative spacing (about 500 via/mm^2) and physical and electrical interfaces $\simeq$30 fF [19–21]. In this paper, we assume the second type TSV process [19] coupled with STMicroelectronics CMOS-28 nm low-power technology library.

3 2D Logarithmic interconnect

The basic 2D-LIN is a low-latency and flexible crossbar that connects multiple PEs to multiple SRAM memory modules (MMs) (see Fig. 1*a*). The IP is optimised to provide fast arbitration and single-cycle access to TCDM banks, and synchronisation mechanisms for inter-process communication. It is built following the MoTs approach, where the network is created combining binary trees [13]. Each tree provides a unique combinational path between the PEs and one MM, and vice versa. Therefore the request and the response paths are decoupled in 2D-LIN to maintain non-blocking communication (see Fig. 2*a*).

The key property of this soft IP is the reconfigurability. User has control on the following parameters: number of PE channels (p), number of direct-memory-access (DMA) channels (d), number of MMs (m), size of each MM in kilobytes (s) and width of data bus (w). Furthermore, bank/word level interleaving are both supported, and arbitration can be performed using either pseudo-least-recently-granted (LRG) or pseudo-round-robin (RR) methods, which are modified versions of LRG and RR to become suitable for implementation inside binary trees. The fraction m/p is defined as 'BankingFactor'. When this parameter is less than or equal to one, there will be a high number of collisions between PEs while accessing memory banks, and performance drops severely. Simulation results with random loads show that a banking factor of two offers over 94% of the performance of the ideal case where no collision exists.

Each clock cycle, all the requests made from masters (PEs and DMAs) are propagated through the request blocks. Collisions because of multiple requests directed to the same memory bank are avoided by the arbitration performed in each node. PEs losing the arbitration are stalled, and the winners conclude the transfer in a single clock cycle in case of a store, whereas, in case of a load, the read data are returned the next clock cycle. This architecture supports atomic test-and-set operation as well. It should be noted that, apart from the topology, also the flow-control mechanism presented in this work differs from [12] in the way that it operates only at one edge of clock; therefore the clock period can be reduced further, and even though in this method read operations takes two cycles to complete, pipelining of reads allows an average performance of one read per cycle.

4 3D Logarithmic interconnect

Within the 2D-LIN architecture, when more storage capability is needed, the increase in network size and routing congestion limit the maximum achievable operating frequency. This happens because crossbar-switches are inherently non-scalable. Centralised 3D-LIN (C-LIN) and distributed 3D-LIN (D-LIN) are two extensions of the 2D-LIN to be integrated in a 3D-stacked chip multiprocessor. These topologies allow designers to overcome the 2D limitation by automatically splitting the design into one logic layer and several memory layers and stacking them over each other. Both networks have been designed based on the assumption that memory layers with identical layouts will be stacked over each other, forming vertical memory cones with all their parameters automatically configured during the boot procedure. This allows for reduction in the chip cost and design effort, and adds design flexibility. To allow stacking of identical memory dies, all components on different memory layers share the input data and control signals from the logic die, and tri-state data buses for their responses, as well. Lastly, both C-LIN and D-LIN support configurable parameters of 2D-LIN, plus a parameter l which represents the maximum number of stackable memory layers, although the number of stacked memories can be chosen freely after the 3D chip assembly step.

It should be noted that in the 3D network presented in [13], interconnects are replicated completely in each memory layer,

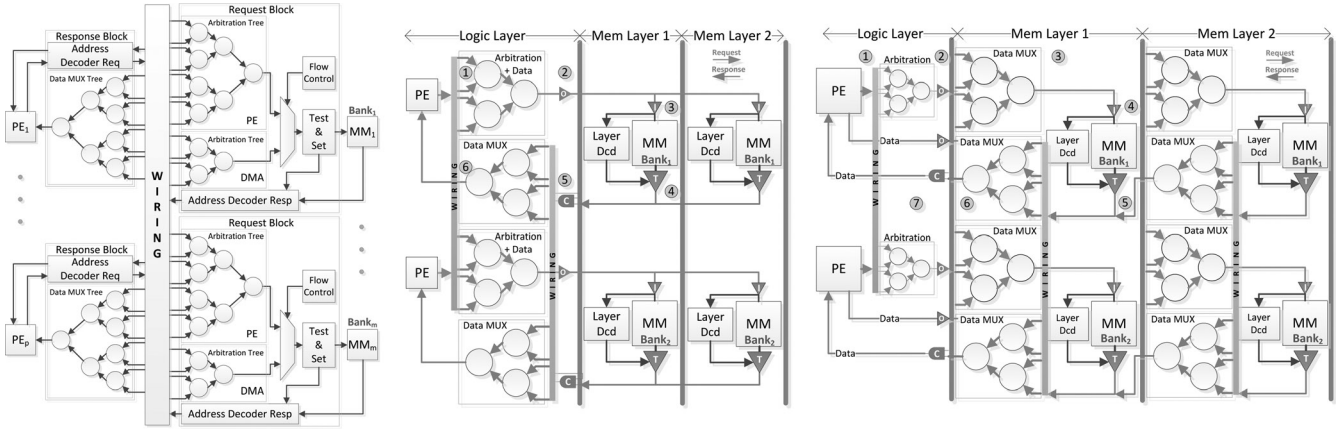

Fig. 2 *Block diagrams of the original 2D-LIN and the proposed 3D alternatives*

a 2D-Logarithmic Interconnect (2D-LIN)
b Centralized 3D-Logarithmic Interconnect (C-LIN)
c Distributed 3D-Logarithmic Interconnect (D-LIN)

and a copy of the arbitration circuits for each layer is placed on the logic layer. As a result, addition of new memory layers, increases 'BankingFactor', and therefore the size of the logic-die grows. Whereas, in our designs, addition of new memory layers does not affect 'BankingFactor' and only adds to the capacity of the existing banks. This makes our solutions more scalable (the comparison between obtained results is performed in Section 6.2).

4.1 Centralised 3D-LIN

C-LIN is the simplest extension to 2D-LIN. As illustrated in Fig. 1*a*, the 2D design is cut at the memory interface therefore PEs and the interconnection network are placed on the logic layer, whereas MMs along with small layer decoding logics are placed on memory layers (see Fig. 2*b*). One benefit of this architecture is that logic and memory elements are completely separated therefore different technologies and optimisations may be utilised for design of the logic and memory dies. In addition, memory layers in C-LIN can be designed as simple, small and inexpensive as possible. The network operation of C-LIN is similar to 2D-LIN with the difference that after the arbitration in the logic layer, the winner request will be sent to memory layers through the TSVs and request address will be matched with 'LayerID' (a number which uniquely identifies each memory layer) in the layer decoding logics of each layer. Therefore the address space is divided among the memory layers, and for each memory bank only one layer will be active at a time. This helps maintaining the combinational nature of the LIN and avoids insertion of buffers and FIFOs between layers.

4.2 Distributed 3D LIN

In the other alternative, D-LIN, 2D design is cut at the PE interface (see Fig. 1*a*) therefore interconnect is distributed among the layers as illustrated in Fig. 2*c*. Similar to C-LIN, flow-control is performed in the logic layer, whereas after arbitration in the logic layer filtered requests will be propagated to memory layers, and in the target memory layer, knowing that all collisions have been already resolved, simple multiplexer trees lead data into MMs. Response networks also act similar to C-LIN, with the difference that they are located in memory layers. In both C-LIN and D-LIN, outputs from different memory layers are resolved using tri-state logic.

The main benefit of D-LIN is reduction in number of TSVs. Since the number of TSVs in D-LIN is proportional to the number of masters ($p + d$), and because 'BankingFactor' is usually greater than one, the number of master channels will be less than the slave channels (m); hence, the reduction in the number of TSVs.

5 Dealing with 3D integration issues

This section presents architectural solutions to the issues related to 3D integration of C-LIN and D-LIN.

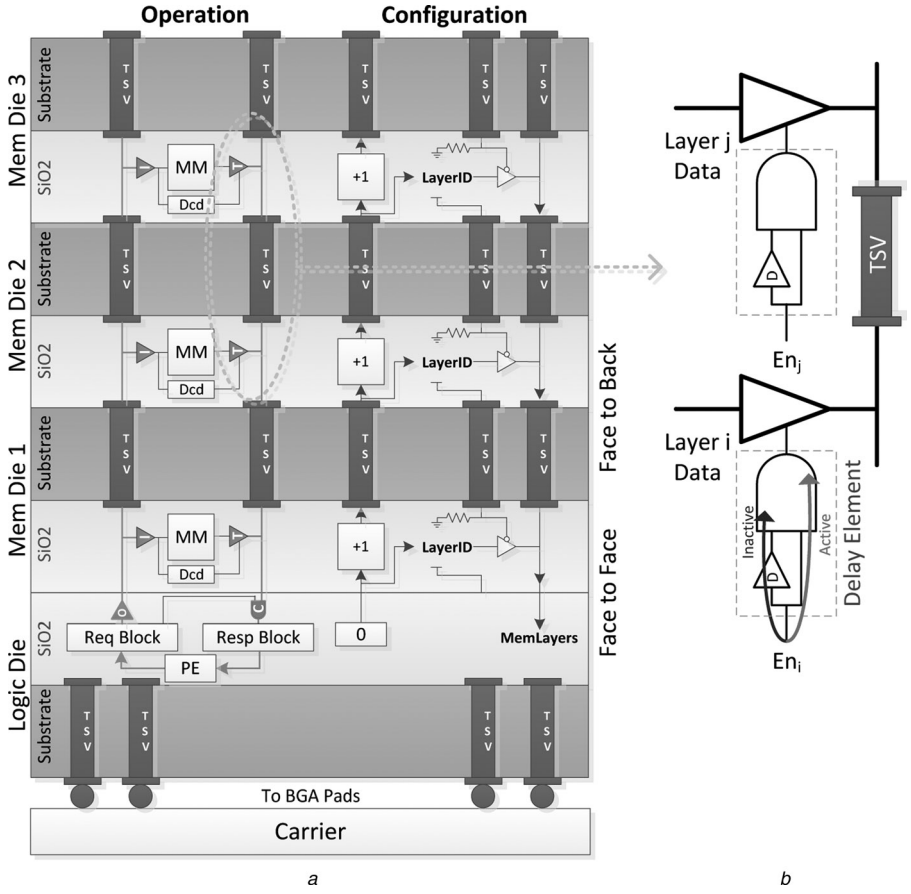

Fig. 3 *Structure of 3D stacking*

a Cross section of C-LIN and boot-time configuration circuitry
b Delay elements to remove the high-current glitches

doi: 10.1049/iet-cdt.2013.0031

5.1 ESD protection

The inter-die signal interfaces in a 3D-IC (integrated circuit) are vulnerable to electrical stress induced during stacking or testing steps. Approximately 60% of silicon IC failures are a result of electrical overstress or ESD (EOS/ESD) [22]. To cope with these issues, IOs passing through the TSVs will be protected by input and output protection circuitry. As illustrated in Figs. 2*b* and *c*, four types of IO drivers are designed for this purpose: *O* and *T* are simple and tri-state output buffers, respectively, with reverse diodes for protection at their outputs, and *I* and *C* are input buffer and clamp circuits with protection diodes at their input stages. The *O* and *C* cells require level-shifters as well, since they may operate between two different voltage domains. All these cells have been adopted from conventional IO pad drivers and are tuned to drive much less capacitance of stacked TSVs of at most eight layers. To avoid a long chain of buffers, the *I* and *T* cells are placed out of the chain at the inputs and outputs of each MM (see Fig. 3*a*). This way, the signals do not need to travel through multiple buffers to reach a MM.

5.2 Boot-time configuration

In order for the memory layers to have identical layouts, boot-time configuration circuits are required to assign unique 'LayerID' values to each layer. For this purpose, assuming via-first technology, 'LayerID' is incremented in each layer and sent to the next memory layer (see Fig. 3*a*). In order to provide the total stacked memory size to the operating system, last layer is identified by means of a pull-down resistor, and its 'LayerID' is returned to the logic layer as the number of memory layers.

5.3 Process/voltage/temperature variations

The importance of process, voltage and temperature variations intensifies in 3D circuits, since the dies from different process corners may be stacked over each other, and timing critical circuits such as the clock distribution network have to operate correctly under these conditions. One problem with such issues is the appearance of high-current glitches on outputs of tri-state drivers. As Fig. 3*b* illustrates, only one of the drivers should be active at a time; however, because of variations, one layer may start driving the bus before another has stopped therefore high-current glitches will appear on the output bus which may degrade the chip's life and performance. In order to solve this issue, the driver in each layer should guarantee that it will return to an inactive state before any other starts to drive. The simple delay elements illustrated in Fig. 3*b* can serve for this purpose, and by adjusting the delay between activation and deactivation glitches can be completely eliminated.

Another issue is the clock skew among memory layers, which has been analysed thoroughly in [23, 24]. In order to maintain the combinational nature of the interconnection network, we utilise only the simplest method, which is increasing clock margins during the clock tree synthesis phase.

6 Experimental results

In this section, we discuss the experimental results for the low-latency networks in terms of timing performance and silicon area. Our baseline 2D-LIN platform is a multi-core system composed by 16 [STxP70 is a cost effective 32-bit ASIP RISC core implemented using a seven-stage pipeline for reaching 600 MHz, which can execute up to two instructions per clock cycle (dual issue) [3]] PEs that share the on-chip TCDM with 32 memory banks ('BankingFactor' = 2) each having a size ranging from 8 to 64 KB. While in C-LIN and D-LIN, bank size is fixed (8 KB), and memory size increases modularly by changing the number of stacked layers from 1 to 8. For physical synthesis, we utilised a hierarchical design flow, in which, after a preliminary synthesis and timing budgeting of the whole design; topographical synthesis and place and route are performed separately for logic and one memory layer. Then, the layers are assembled together and a capacitive load of 30 fF is used to model each TSV [19] [the capacitance of the micro-bumps is negligible compared with the TSV itself (less than 2 fF) [25]]. Finally for the sign-off step, post place and route net-list along with the parasitic are fed into Primetime and a multi-corner static timing analysis is performed. If the obtained results are not suitable, the flow should iterate once more with possible changes in constraints.

We explored different configurations in terms of memory size embedded in the design, with metrics derived from the state-of-the-art technology and tools. Our design flow is based on the STM CMOS-28 nm low-power technology library, with a multi V_{TH} synthesis flow with Synopsys Design Compiler Graphical (2011.09), and place and route with Cadence S.C Encounter Digital Implementation (10.1), and the sign-off tasks in Primetime (2011.09). We assumed that memory dies in the 3D designs are stacked on top of the logic die, which provides power supply, clock and data/control signals to them. The logic die has been designed using 10 metal layers, while this number has been reduced to seven in memory dies because of lower routing complexity. [This reduction would lead to a significantly reduced mask and production cost, which is an example of how 3D integration enables cost optimisation [26]] Memories and PEs have been implemented using predesigned hard macros. As can be seen in Fig. 3*a*, the first memory layer is stacked over the logic layer using the face-to-face technology, as the others use face-to-back stacking. This eliminates the need for TSVs between the first two layers. In addition, the operating voltage of the memory layers has been increased slightly over the logic layer. This allows for removal of the level-shifter inside the *C* cell, and a reduction in its layout size and delay (considering the fact that for 32 memory banks about 1000 *C* cells are required). Lastly, we implemented our solutions with two different bonding techniques: micro-bumps and Cu–Cu direct bonding [27].

In this section, first we try to present the benefits of the LIN in comparison with existing interconnect solutions, then we explore the design alternatives. Finally, we discuss the obtained results.

6.1 Comparison with other topologies

First, we compare LIN with an extremely high-performance NoC to show the superiority of low-latency interconnects in processor-to-L1-memory context. We modelled our baseline LIN(16×32) and a Mesh-NoC(4×4) presented in [28] in a home-made cycle accurate trace simulator, fed with memory traces from MPARM [29] running a 16 PE configuration. For LIN, we assumed a clock frequency of

400 MHz, Pseudo-RR arbitration, and single-cycle access to memory banks. While for the NoC we assumed a clock frequency of 5 GHz (this is very optimistic for a low-power process such as STM CMOS-28 nm), flit size of 32 bits, six-port switches (*N*, *S*, *E*, *W*, *P* and *M*) with fall-through latency of five clock cycles and link latency of one cycle. *X*–*Y* routing is assumed, as well. In our comparison we use 8 kB memory banks from the STM 28 nm CMOS technology, with an access time of 1 ns. We have attached two of these banks to each NoC switch aggregating a total of 32 banks. In the first experiment, each PE sends uniform traffic to the address range of ['HomeBank' ± 'WorkingSetSize'/2], for 10 000 transactions. As Fig. 4 illustrates, when 'WorkingSetSize' is small, every PE accesses its home bank only and NoC performs slightly better than LIN. However, as PEs start to access remote banks, NoC's execution time increases rapidly. Although on the other hand, LIN's execution time recovers after an increase, because of load distribution among multiple banks.

For the second experiment, three embedded parallel image processing benchmarks [30] have been executed, and a time-division multiplexing bus has been modelled in our simulator, as well. Table 1 presents the results, where Features from Accelerated Segment Test (FAST) is a corner detection algorithm, SIFT is an scale invariant feature transform and CT is a colour tracking algorithm. In all cases, LIN performs better than NoC, and bus results are an upper bound to the execution time since all accesses result in contention.

The reason for this advantage is that even though mechanisms such as speculative and look-ahead routing allow for a bandwidth of about 1 'flit/cycle' in NoCs, yet they are not beneficial in processor-to-L1-memory context. Since traffics do not have a bursty nature, and small packets cannot benefit from the huge bandwidth provided by NoC

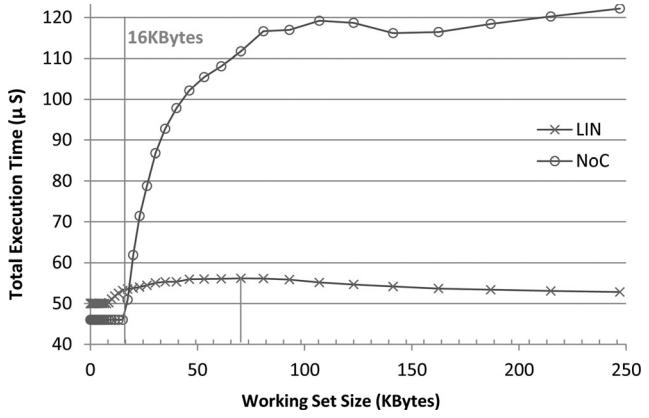

Fig. 4 *Performance comparison between NoC and LIN under different working set sizes*

Table 1 Performance comparison between LIN, NoC and bus executing different benchmarks

Benchmarks		FAST	CT	SIFT
LIN	execution time, ms	5.59	79.89	4464.07
	AMAT, ns	6.54	6.47	6.46
NoC	execution time, ms	8.21	106.92	4943.43
	AMAT, ns	7.53	7.09	6.57
bus	execution time, ms	46.40	730.51	30 799.99
	AMAT, ns	81.30	82.40	81.90

Table 2 Comparison of post-layout area between LIN and NoCs

Interconnect	Cardinality	Area, mm^2
LIN	($p = 16$, $m = 32$)	0.09
NoC-3.6 GHz [31]	4 × 4	0.29
MIRA (3DM) [32]	4 × 4	0.40
MIRA (3DM-E) [32]	4 × 4	0.98
NoC-5.1 GHz [28]	4 × 4	1.02

switches. In addition, memory access latency is critical in this context since it can lead to stalling the pipeline of the processors. Also it should be noted that there is an opportunity to further optimise the speed of LIN using custom circuit techniques, such as [11]. However, this will result in incompatibility with standard technology libraries provided by silicon foundries and remove the configurability features of LIN.

Next, Table 2 compares the post-layout area between LIN (16 × 32) and three other NoCs. All results have been scaled to 28 nm technology, and NoC switches have been scaled to flit size of $F = 32b$ by a factor of $(F_2/F_1)^{1.8}$ (extracted from Orion 2.0 [33]). As can be seen, NoCs require much larger areas than LIN, which is because of the large number of buffers and memory elements used in them.

6.2 Design alternatives

This subsection presents the results of timing performance and silicon area for our three designs: 2D-LIN, C-LIN and D-LIN; implemented using two different bonding techniques. The area for PE hard macros (STxP70) is about 0.25 mm^2 and for each 8 KB memory bank about 0.02 mm^2. Also for the micro-bumps, a minimum pitch of 40 μm × 50 μm, and for the direct bonding a more dense pitch of 10 μm × 10 μm have been used [27]. In addition, as a corner case, the ESD protection circuitry has been removed from the direct bonding technique. The resulting layouts after full placement and routing for 2D-LIN and D-LIN (Cu–Cu direct bonding) are depicted in Fig. 5.

Fig. 6*a* illustrates the silicon area (mm^2) for 2D and 3D implementations. As can be seen, the 2D die size increases with the embedded on-chip SRAM, except for the first two configuration, where PE obstructions and design geometry dominate the total design area (see also Fig. 5*a*). As the number of memory banks increases, total area grows and large channels should be allocated for wires to reduce the routing congestion. Relative distances increase, and a massive number of buffers are inserted by the synthesis tool. In the 3D configurations, memory dies are equipped with a large number of TSVs; therefore a large portion of the die is allocated for TSV placement (routing obstruction and keep-out region for placement). This is illustrated in Fig. 5*d*. It should be noted that in this experiment, C-LIN contains 2688 TSVs, whereas in D-LIN this number is reduced by 47% to 1424. The effect of TSVs is intensified when micro-bumps are used because of the large pitch.

Timing results are depicted in Fig. 6*b*, where eight memory layers have been stacked for 3D configurations. It can be seen that C-LIN and D-LIN improve the performance over 2D-LIN with the same memory size by small factors of 6.7 and 3.7%, respectively, whereas it is usually believed that TSVs can greatly reduce wire length and improve clock frequency. In order to explain this, consider the 2D planar design with 2 MB of memory, for which we obtained a maximum

doi: 10.1049/iet-cdt.2013.0031

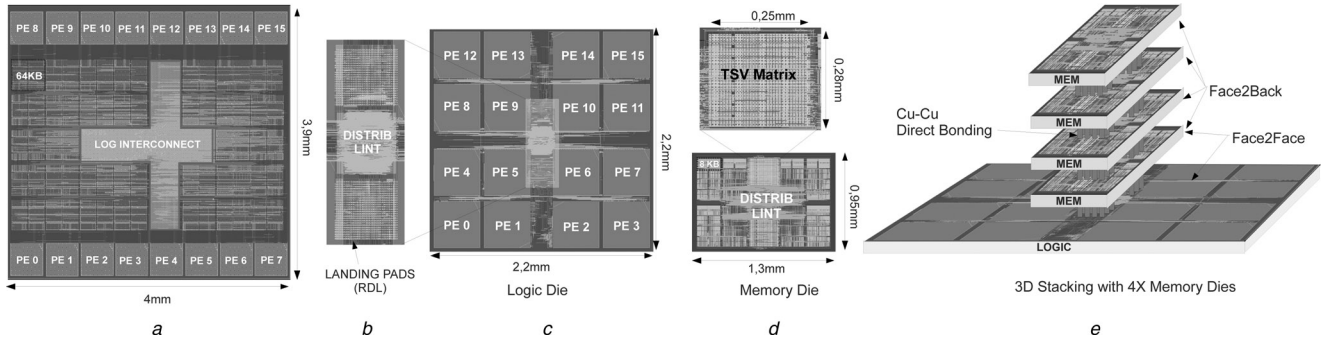

Fig. 5 *Physical implementation of the designs*

a 2-LIN with 2 MB SRAM
b Details of the landing pads on redistribution layer in D-LIN
c Logic die of D-LIN with Cu–Cu direct bonding
d Memory die of D-LIN with details of the TSV matrix
e 3D Stacking with four stacked memory dies

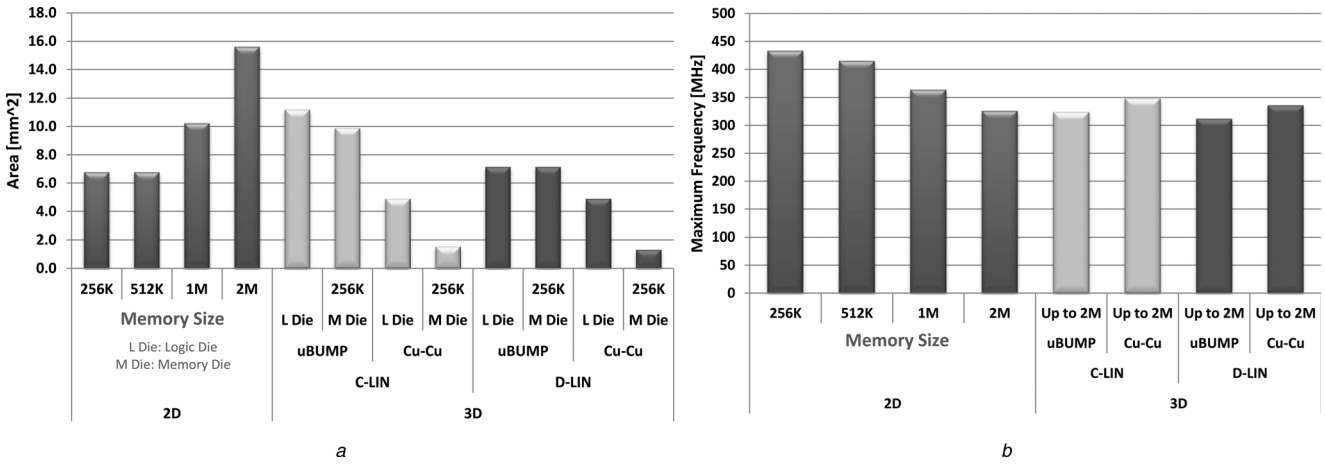

Fig. 6 *Design implementation results*

a Silicon area (mm^2) for 2D and 3D implementations
b Maximum achievable frequency (MHz)

frequency of 324 MHz. Whereas for the C-LIN design with Cu–Cu direct bonding, we obtained a maximum frequency of 348 MHz. If we assume that TSVs are ideal with zero capacitance, we can obtain a maximum frequency of 433 MHz for the 3D design (which is comparable with the frequency of a 2D design with 256 kB of memory). Comparing this clock frequency with 348 MHz for C-LIN gives that 0.56 ns of the critical path is devoted only to driving the TSVs, or in other words TSVs and their drivers drop the performance over the ideal case by a factor of 24%. This situation can be further explained considering the 30 fF capacitive load of TSVs, resulting in a total load of 240 fF in a stack of eight TSVs, which is roughly equal to 4 mm of 'Metal'8 wire having a coupling capacitance of 66 fF/mm. This explains that current TSVs are not yet scaled enough to provide a major performance boost over 2D planar designs. Moreover, from comparison of results between micro-bumps and direct copper bonding without protection circuits, it can be estimated that micro-bumps and protection circuits further drop the performance by a factor of about 9%, consuming 0.21 ns of the critical path.

One last point to mention is that, system latency in [13] is calculated as (network latency + memory access time), and by maintaining a fixed 'BankingFactor' (reduction in number of banks per layers by addition of new layers), it has been shown that both system and network latency decrease significantly. Whereas in our experiments, we showed that a major contributor to the performance drop is the latency of the TSVs and their drivers; and in order to support our argument, we performed timing characterisation of the whole 3D stack considering the TSV loads and their driver circuits, and derived maximum achievable frequency directly from the post-layout results of the physical synthesis tool.

6.3 Discussion

As our results demonstrated, the cost (size and speed) of the TSVs including the protection circuitry in current 3D technologies are not dominating over on-chip wires unless we go towards larger 2D die sizes. However, 3D TSV technology can help reducing significantly the overall cost of the die stack, by implementation of the memory dies at lower costs: Using reduced number of masks or different technology options (e.g. different thresholds or different oxide thickness for the memory transistors to minimise leakage) to have better memories compared with the ones that could be implemented on the same die as the logic [26]. Furthermore, long critical paths in our single-cycle design may suggest that current TSVs can be more

beneficial in higher levels of memory hierarchy, where latency is not critical and pipelining can break the critical paths [34]. Lastly, it should be noted that, as the network obtains larger, the place and route effects, such as long-wiring buffers, and routing congestion become increasingly important, and we believe that delays will increase even more in larger designs. Also, it should be noted that in a real design back-end, multi-corner and possibly multi-mode analysis should be performed, which will make convergence even more difficult. Therefore we suggest to build processing clusters with tightly coupled memories using LIN, and use NoC as another level of hierarchy for inter-cluster communication, to benefit from both low latency of LIN and scalability of NoC.

7 Conclusions

In this paper, we presented two synthesisable network architectures, C-LIN and D-LIN derived from the LIN, which can be integrated with 3D stacking technology to provide access to tightly coupled shared memory banks stacked over multi-core clusters. Architectural simulation results demonstrated that in processor-to-L1-memory context, LIN outperforms both traditional NoC and simple time-division multiplexing buses. Furthermore, we devised a modular design strategy which allows users to stack multiple memory dies and create different height stacks with identical dies, without the need for different masks for dies at different levels in the stack. The designs have been explored in terms of area and latency, and full-layout results show that for large 2D designs the main problems are routing congestion, signal integrity and the mask cost. Therefore our proposed 3D designs offer better scalability with a similar performance; however, in terms of delay, the 3D designs are not so competitive with the 2D planar design, unless we go towards larger 2D chips. As a conclusion, even though the current TSVs are still not much better in terms of speed than global on-chip wires, they can provide more freedom in heterogeneous integration of dies with cost-optimised technologies, since they are definitely much better than traditional off-chip links.

8 Acknowledgments

This work was supported, in parts, by the EU FP7 Project Phidias (GA no. 318013) and ERC-AdG Multitherman project (CA no. 291125).

9 References

1 'The next generation cuda architecture, code named fermi', White Paper, NVIDIA, September 2009
2 'The hypercore architecture', White Paper, Plurality, Ltd., January 2010
3 Melpignano, D., Benini, L., Flamand, E.: 'Platform 2012, a many-core computing accelerator for embedded socs: performance evaluation of visual analytics applications'. Proc. 49th Annual Design Automation Conf. ser. DAC'12, New York, NY, USA, 2012, pp. 1137–1142
4 Banakar, R., Steinke, S., Lee, B.S.: 'Scratchpad memory: a design alternative for cache on-chip memory in embedded systems'. Proc. Tenth Int. Symp. Hardware/Software Codesign, 2002 (CODES 2002), 2002, pp. 73–78
5 Borkar, S.: 'Networks for multi-core chips: a contrarian view'. Symp. Low Power Electron. Design (ISLPED), 2007
6 Haring, R.A., Ohmacht, M., Fox, T.W.: 'The ibm blue gene/q compute chip', *IEEE Micro*, 2012, **32**, (2), pp. 48–60
7 Satpathy, S., Zhiyoong, F., Giridhar, B.: 'A 1.07 Tbit/s 128×128 swizzle network for simd processors'. Proc. 2010 IEEE Symp. VLSI Circuits (VLSIC), 2010, pp. 81–82
8 Balfour, J., Dally, W.J.: 'Design tradeoffs for tiled cmp on-chip networks'. Proc. 20th Annual Int. Conf. Supercomputing (ser. ICS'06), New York, NY, USA, 2006, pp. 187–198
9 Das, R., Eachempati, S., Mishra, A.K.: 'Design and evaluation of a hierarchical on-chip interconnect for next-generation cmps'. Proc. IEEE 15th Int. Symp. High Performance Computer Architecture, 2009 (HPCA 2009), 2009, pp. 175–186
10 Kim, J., Balfour, J., Dally, W.: 'Flattened butterfly topology for on-chip networks', *Comput. Archit. Lett.*, 2007, **6**, (2), pp. 37–40
11 Sewell, K., Dreslinski, R.G., Manville, T.: 'Swizzle-switch networks for many-core systems', *IEEE J. Emerging Sel. Top. Circuits Syst.*, 2012, **2**, (2), pp. 278–294
12 Rahimi, A., Loi, I., Kakoee, M.R.: 'A fully-synthesizable single-cycle interconnection network for shared-l1 processor clusters'. Design, Automation Test in Europe Conf. Exhibition (DATE), 2011, 2011, pp. 1–6
13 Beanato, G., Loi, I., De Micheli, G.: '3D-LIN: A configurable low-latency interconnect for multi-core clusters with 3d stacked l1 memory'. Proc. 2012 IEEE/IFIP 20th Int. Conf. VLSI and System-on-Chip (VLSI-SoC), 2012, pp. 30–35
14 Saito, H., Nakajima, M., Okamoto, T.: 'A chip-stacked memory for on-chip sram-rich socs and processors'. IEEE Int. Solid-State Circuits Conf. – Digest of Technical Papers, 2009 (ISSCC 2009), 2009, pp. 60–61,61a
15 Kim, D.H., Athikulwongse, K., Healy, M.: '3d-maps: 3d massively parallel processor with stacked memory'. Proc. 2012 IEEE Int. Solid-State Circuits Conf. Digest of Technical Papers (ISSCC), 2012, pp. 188–190
16 Ito, K., Saen, M., Osada, K.: 'Hierarchical 3d interconnection architecture with tightly-coupled processor-memory integration'. Proc. 2010 IEEE Int. 3D Systems Integration Conf. (3DIC), 2010, pp. 1–6
17 Davis, W., Wilson, J., Mick, S.: 'Demystifying 3d ics: the pros and cons of going vertical', *IEEE Des. Test Comput.*, 2005, **22**, (6), pp. 498–510
18 Gupta, S., Hilbert, M., Hong, S.: 'Techniques for producing 3-d ics with high-density interconnect'. Int. VLSI Multi-Level Interconnection Conf., Waikoloa Beach, HI, USA, 2004
19 Van der Plas, G., Limaye, P., Mercha, A.: 'Design issues and considerations for low-cost 3d tsv ic technology'. Proc. 2010 IEEE Int. Solid-State Circuits Conf. Digest of Technical Papers (ISSCC), 2010, pp. 148–149
20 Radojcic, R.: 'Roadmap for design and eda infrastructure for 3d products', Available at http://www.eda.org/edps/EDP2012/Papers/3D Riko Radojcic Keynote.pdf, accessed April 2012
21 Vivet, P., Dutoit, D., Thonnart, Y., Clermidy, F.: '3d noc using through silicon via: an asynchronous implementation'. Proc. 2011 IEEE/IFIP 19th Int. Conf. VLSI and System-on-Chip (VLSI-SoC), 2011, pp. 232–237
22 Rosenbaum, E., Shukla, V., Keel, M.-S.: 'Esd protection networks for 3d integrated circuits'. Proc. 2011 IEEE Int. 3D Systems Integration Conf. (3DIC), 2012, pp. 1–7
23 Pavlidis, V., Savidis, I., Friedman, E.: 'Clock distribution networks for 3-d integrated circuits'. Custom Integrated Circuits Conf., 2008 (CICC 2008), 2008, pp. 651–654
24 Xu, H., Pavlidis, V.F., De Micheli, G.: 'Effect of process variations in 3d global clock distribution networks', *J. Emerging Technol. Comput. Syst.*, 2012, **8**, (3), pp. 20:1–20:25
25 Jain, A.: 'Research challenges and opportunities in 3d integrated circuits'. Freescale semiconductor, Available at http//:www.usu.edu/mrc/Ankur_Jain.pdf, accessed January 2009
26 Dong, X., Xie, Y.: 'System-level cost analysis and design exploration for three-dimensional integrated circuits (3d ics)'. Asia and South Pacific Design Automation Conf., 2009 (ASP-DAC 2009), 2009, pp. 234–241
27 Marinissen, E.J., Daenen, T., Dupas, L.: 'Wafer probing on fine wafer probing on fine-pitch pitch micro bumps for 2.5d and 3d sics'. IMEC, Available at http://www.swtest.org/swtw_library/2011proc/PDF/S04_03_Marinissen_ SWTW2001.pdf, accessed June 2011
28 Vangal, S., Singh, A., Howard, J.: 'A 5.1 GHz 0.34 mm^2 router for network-on-chip applications'. IEEE Symp. VLSI Circuits, 2007, 2007, pp. 42–43
29 Benini, L., Bertozzi, D., Bogliolo, A.: 'Mparm: exploring the multi-processor soc design space with systems', *J. VLSI Signal Process. Syst.*, 2005, **41**, (2), pp. 169–182
30 Sw/hw extensions for heterogenous multicore platforms (virtical), Available at http://www.virtical.eu/
31 Kumar, A., Kundu, P., Singh, A.P.: 'A 4.6 Tbits/s 3.6 GHz single-cycle noc router with a novel switch allocator in 65 nm cmos'. ICCD, 2007, pp. 63–70
32 Park, D., Eachempati, S., Das, R.: 'Mira: A multi-layered on-chip interconnect router architecture'. Proc. 35th Int. Symp. Computer Architecture, 2008 (ISCA '08), 2008, pp. 251–261

doi: 10.1049/iet-cdt.2013.0031

33 Kahng, A.B., Li, B., Peh, L.-S., Samadi, K.: 'Orion 2.0: A fast and accurate noc power and area model for early-stage design space exploration'. Design, Automation Test in Europe Conf. Exhibition, 2009 (DATE '09), 2009, pp. 423–428

34 Kang, K., Benini, L., Micheli, G.: 'A high-throughput and low-latency interconnection network for multi-core clusters with 3-d stacked l2 tightly-coupled data memory'. Proc. 2012 IEEE/IFIP 20th Int. Conf. VLSI and System-on-Chip (VLSI-SoC), 2012, pp. 283–286