Contents lists available at ScienceDirect

Integration, the VLSI Journal

journal homepage: www.elsevier.com/locate/vlsi

Linear Clock Tree Topology for Dynamic Source Synchronous and Fully Synchronous 3-D Interfaces[☆]

Andres Ayes[*], Eby G. Friedman

Department of Electrical and Computer Engineering, University of Rochester, Rochester, NY, United States of America

ARTICLE INFO

Index terms:
3-D IC
Synchronization
All-digital delay-locked loop
Source synchronous
Through silicon via (TSV)

ABSTRACT

Reliable timing verification is a primary challenge in three-dimensional (3-D) integrated circuits. Process, temperature, and voltage variations between tiers within an integrated system introduce timing uncertainty into the clock and data paths, resulting in reduced operating speed. A clock tree topology that supports synchronous and source synchronous timing schemes is proposed for 3-D interfaces. The interface operates in a source synchronous scheme for fast data transfer and transitions to a synchronous scheme for bidirectional data flow. Phase compensation for the synchronous scheme is achieved via a delay locked loop.

1. Introduction

Three-dimensional (3-D) integration is a promising technology offering higher transistor density, shorter interconnects, lower power, and heterogeneous integration [1]. These enhancements are possible due to through silicon vias (TSV) which allow the individual silicon tiers to be vertically connected. Each tier within a 3-D system exhibits different process, voltage, and temperature (PVT) characteristics. The TSV transmission line impedances also have to be considered since the via resistance and coupling capacitance introduce additional latency.

Two timing schemes are discussed here, fully synchronous and source synchronous. An important issue in synchronous timing architectures is the clock skew. Clock skew is the difference in the arrival time of the clock signal between two sequentially-adjacent registers [2–4] The fully synchronous scheme exhibits little to no skew between the clock arrival times between sequentially-adjacent registers. Clock distribution networks can be designed to minimize skew at every clock destination [1]; however, typical approaches lead to overly thick and long clock lines, resulting in greater coupling capacitance, overuse of resources, and higher power dissipation. Alternatively, a delay locked loop (DLL) can be used to synchronize the clock signal by introducing delay into the clock line, using feedback to properly synchronize the many data paths.

A source synchronous scheme sends the clock signal along with the data signal. This scheme supports timing verification since both clock and data experience similar PVT variations. The source synchronous scheme allows the clock to operate faster since the relationship between the clock arrival time and the data arrival time at the destination register can be accurately set. This scheme, however, produces certain tradeoffs. Data can only travel in one direction; meaning that two interfaces need to be included when data is passed between 3-D tiers. In an interface composed of hundreds or more bits, it is impractical to pass a local clock with each bit.

A third approach is an asynchronous scheme. This scheme allows the interface to exchange data without requiring a shared clock signal. The exchange is typically achieved via a handshake protocol, where the source sends the receiver a request to send the data, and the receiver acknowledges the signal when the receiver is ready to accept new data [5]. This architecture eliminates certain timing hazards but also increases latency due to the additional logic. The focus of this paper, however, is on synchronous schemes.

The purpose of this effort is to exploit the benefits of higher clock frequencies offered by a source synchronous scheme while supporting bidirectionality by switching from a source synchronous scheme to a fully synchronous scheme. The synchronous scheme is achieved by a DLL whose output becomes the clock source of the interface when data travels in the opposite direction as the source synchronous clock signal. This concept is illustrated in Fig. 1.

This paper is structured as follows: Related work in 3-D clocking is summarized in Section 2. The proposed linear clock tree for 3-D interfaces is described in Section 3. The performance of the linear clock tree is presented for both source synchronous and fully synchronous schemes in Section 4. Some conclusions are drawn in Section 5.

[☆] This work was supported by Qualcomm Corporation, San Diego, California, USA.

[*] Corresponding author.
E-mail addresses: aayes@ece.rochester.edu (A. Ayes), friedman@ece.rochester.edu (E.G. Friedman).

https://doi.org/10.1016/j.vlsi.2023.102066
Received 16 March 2023; Received in revised form 7 June 2023; Accepted 28 July 2023
Available online 2 August 2023

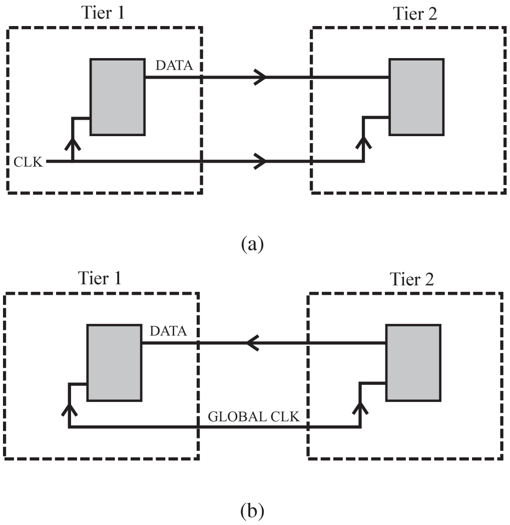


Fig. 1. Data exchange between tiers. (a) Source synchronous scheme. Clock and data travel in the same direction. (b) Fully synchronous scheme. Data can travel in the opposite direction of the source synchronous clock signal.

2. Background

Significant work exists that addresses the challenges of 3-D timing. Closed-form expressions, models, and electrical characterization of 3-D systems are presented in [6,7]. Clock distribution networks that minimize clock skew among all clock destinations in 3-D ICs are discussed in [8,9]. 3-D H-trees exhibit low skew while local mesh topologies dissipate less power. A combination of an H-tree and local meshes leads to low skew and moderate power consumption.

A DLL for 3-D systems compensates for TSV variations [10]. A digital DLL architecture where all of the components are placed on one tier is proposed in [11]. This single tier DLL is effective for heterogeneous integration since the DLL is manufactured in one process while synchronizing two tiers manufactured in different technologies.

A fast locking DLL is proposed in [12]. The DLL uses successive register approximation (SAR) to quickly converge to a delay for the clock line. Eight bits control a delay line consisting of a coarse delay section and a fine delay section.

Power and area models and clock tree synthesis (CTS) design flows for 3-D interfaces are presented in [13]. The models predict power consumption, timing, and area within 10% error for three synchronization topologies: fully synchronous, source synchronous, and asynchronous. The synchronous model, however, assumes skew balancing is performed via a clock tree topology as opposed to using a DLL.

3. Linear topology

The proposed clock tree topology supports two synchronization schemes: source synchronous and synchronous with phase compensation. In this topology, registers are linearly arranged from the source, only allowing for small branches from the primary clock path. After the clock signal has propagated through the portion of the clock tree within the source tier, the clock signal moves onto the next tier, propagating through the rest of the clock tree. Due to the smaller branches, the delay of the data lines can be more accurately matched to the primary clock path. This topology is shown in Fig. 2, where the dashed path represents the data path of the closest register to the clock source, and the dotted path represents the data path of the farthest register to the clock source.

This structure inherently supports a source synchronous scheme. However, given the structure of a linear clock tree, a fully synchronous scheme can be achieved by synchronizing the clock signals between sequentially-adjacent paths [2,3], *e.g.*, the clock signals for FF1 and FF3 or FF2 and FF4. Synchronization between these nodes can be achieved via a DLL that receives a clock signal from one tier as the reference signal and another clock signal from another tier as the feedback signal. The choice of which signals to synchronize is explored here in Section 4. The purpose for switching from a source synchronous scheme to a fully synchronous scheme is to allow data to flow in the direction opposite of the clock while considering PVT variations between tiers and without adding a second interface which would double the area overhead.

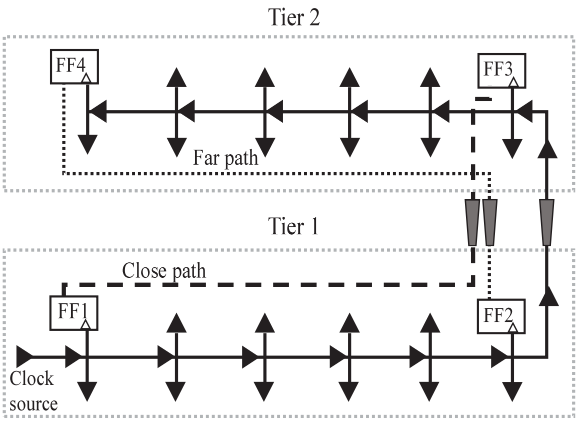


Fig. 2. Linear tree topology. The dashed line represents the data path of the bit closest to the clock source, and the dotted line represents the data path of the bit farthest from the clock source. Both paths are matched to the main clock path, illustrated as a solid line. The branches from the primary clock path are relatively short as compared to the primary path of the clock tree.

4. Performance evaluation

In this section, the performance of a linear clock tree is discussed when operating in these synchronization schemes. A linear tree in a source synchronous scheme is evaluated in Section 4.1. The DLL used in this test circuit and the linear tree in a fully synchronous scheme are described in Section 4.2.

4.1. Source synchronous

The source synchronous timing topology sends the clock signal along with the data signal. By matching the delay of the clock and data lines, the destination register captures the data regardless of wirelength and PVT variations. Race conditions, where the datum is captured at the destination register during the same clock cycle as when the datum is launched from the source register [3], are possible in this scheme. Race conditions can be mitigated by adding sufficient delay to the data line to ensure the datum is captured during the subsequent clock cycle [13].

As described in Section 3, the linear topology inherently supports the source synchronous scheme since the data and clock travel the same distance and are in close proximity. The delay of the clock branches also adds delay to the data lines, ensuring the data are captured in the subsequent clock cycle.

To investigate the performance of the clock tree topology in the source synchronous scheme, a circuit emulating a field of clock destinations is considered. The circuit for one tier is shown in Fig. 3. Note that the root of the clock tree is located on the left and propagates to the right. The parasitic interconnect impedances are represented by an RC π network. The capacitors function as clock sinks. Only two clock sinks for each branch are shown in Fig. 3, but eight sinks are present for each branch in the test circuit. The input capacitance of the buffer is approximated as 2 fF. Note that both the closest sink and the farthest

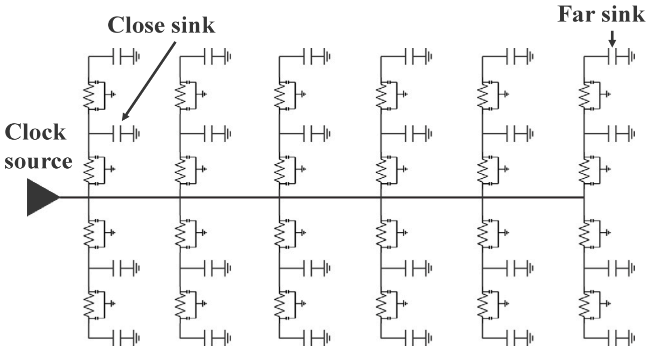


Fig. 3. One tier of the clock tree within a 3-D system. The clock sinks are represented by capacitors. The closest sink to the clock source and the farthest sink from the clock source are noted.

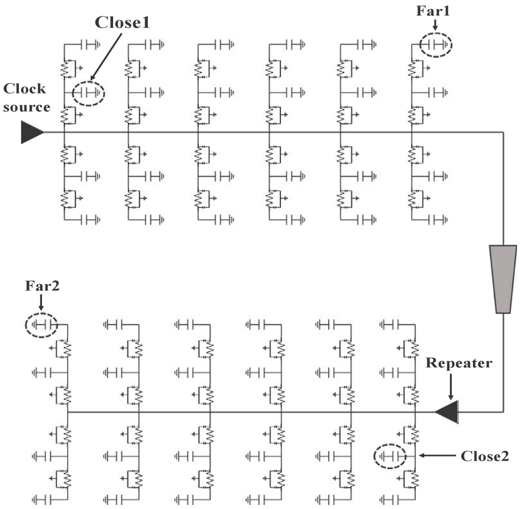


Fig. 4. 3-D clock tree connected by a TSV. Two clock trees, one in the source tier and one in the destination tier, are connected by a TSV. Note that the clock source for the destination tier is the output of the clock repeater placed at the end of the TSV.

sink from the root are noted in Fig. 3. These two leaf nodes experience the extremes of delay from the root of the clock tree. The clock tree within the source tier and the destination tier is shown in Fig. 4. The root of the 3-D clock tree is located in the top tier. The two 3-D planes are connected through a TSV. Note that the clock sink closest to the source in tier 1 is referred to as close1, and the clock sink closest to the clock TSV in tier 2 is referred to as close2. Similarly, clock sinks far1 and far2 refer, respectively, to the farthest clock sink in tier 1 and the farthest clock sink in tier 2.

The output waveforms of the source synchronous clock and flip flop are shown in Fig. 5. In this scheme, the clock signals are unaligned but the data delays match the clock delays. The clock skew for the different clock paths in the source synchronous scheme is listed in Table 1. These clock skews are related to the insertion delay of the clock network from one sink to the next sink. Note that close1 to close2 and far1 to far2 exhibit similar clock skews. These matched clock skews are due to the clock paths experiencing similar clock tree delays. Matched clock skews are an advantage of the linear clock tree topology.

The simulations utilize Cadence Virtuoso with SPECTRE as the circuit simulator. A 10 nm predictive technology model (PTM) library is assumed [14]. The parasitic interconnect impedances are represented by an RC π model, where the sheet resistance is 0.1 Ω/□ and the capacitance is 16 aF/□ [15]. The pitch for the TSVs and registers is 10 μm. The TSV model is described in [16]. The maximum clock frequency achieved by the source synchronous architecture is 2.24 GHz when both tiers operate at low standby power and 6.0 GHz when both tiers operate at high speed. The far-to-far clock skew for the former configuration is 241 ps and is 126 ps for the latter configuration.

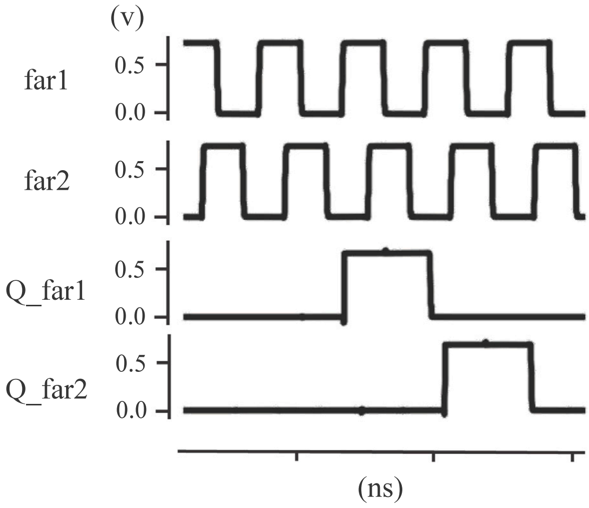


Fig. 5. Source synchronous clock and data waveforms for far1 and far2. The skew between the clock signal at far1 and the clock signal at far2 is caused by the delay of the primary clock path. Since the data lines are matched to the clock line, the data are captured in the destination tier during the subsequent cycle.

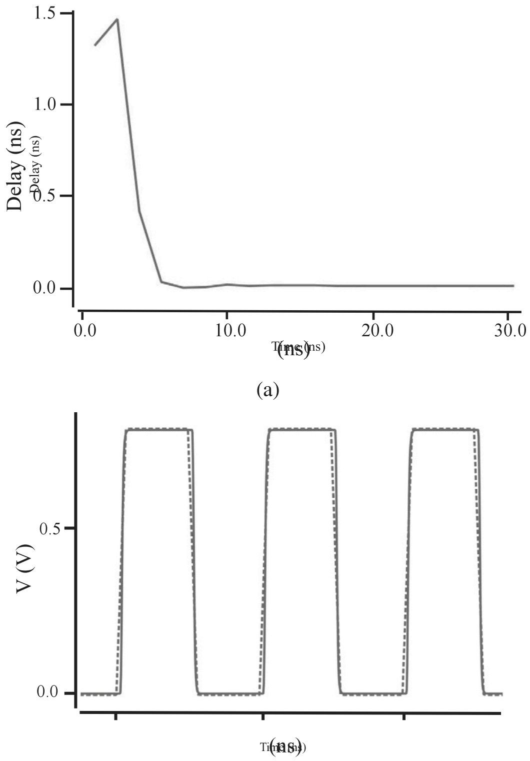
Table 1
Clock skew for each extreme path of the source synchronous topology.

Clock skew (ps)			
Close1 to close2	Close1 to far2	Far1 to close2	Far1 to far2
242	350	134	241

A significant drawback of this topology is the lack of bidirectional capability. In the case of a passive clock line, where no logic components are added to the clock signal, the data are restricted to travel in the same direction as the clock. Thus, for an interface sending data back and forth between tiers, two interfaces are necessary. The synchronous scheme allows the interface to transmit data without a second interface.

4.2. Fully synchronous with delay locked loop

The delay locked loop used in this testbench is based on the aforementioned SAR-based DLL proposed in [12]. This DLL provides fast locking capability [12]. Upon initialization, the most significant control bit is set to logic 1 (which causes the most significant delay element to turn off). During the subsequent cycle, the feedback signal is compared to the reference signal. If the phase of the reference signal lags behind the feedback signal, the most significant control bit is set to logic 0 (which turns on the most significant control element). Otherwise, if the reference signal leads the feedback signal, the control bit remains at logic 1. The next most significant control bit is set to logic 1, and the process is repeated. If the N control bits do not exist after N cycles, the delay inserted into the control line is approximately a multiple of the clock period. Half of the control bits control a coarse delay line, and the other half of the control bits control a fine delay line. Once the DLL has approximated the delay, the DLL adds or removes delay to/from the clock line by one unit of the fine delay element until the phase of the reference and feedback clocks is aligned. In this 10 nm technology, one unit of the fine delay element is between 6 and 10 ps. The delay of one coarse delay element is between 28 and 32 ps. Convergence of the delay inserted into the clock line and the resulting clock waveforms are illustrated in Fig. 6 for a 1 GHz clock signal.



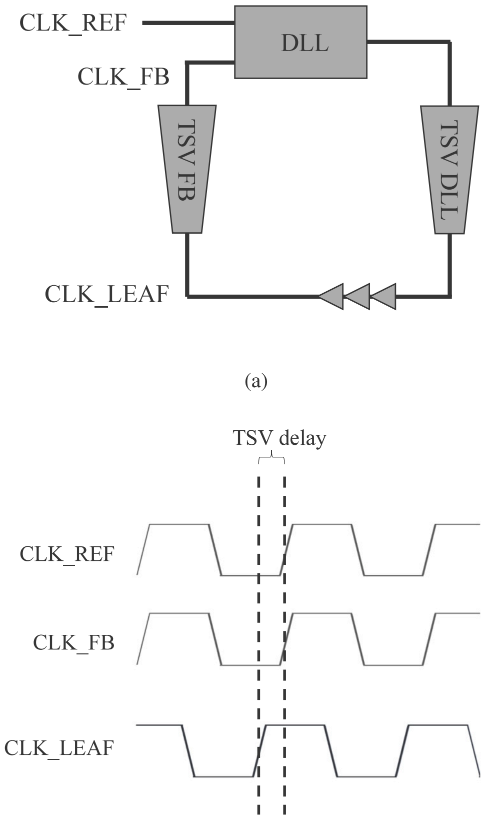
(a)

(b)

Fig. 6. SAR-based DLL converging on a delay multiple of the clock period (1 GHz), (a) delay between the reference signal and the feedback signal normalized to the clock period, and (b) waveform of the reference and feedback signals. The dashed signal is the reference, and the solid signal is the feedback.

The reference clock node is located in tier 1 (the top tier shown in Fig. 4), while the target clock node is located in tier 2 (the bottom tier shown in Fig. 4). The feedback path from the target clock node from tier 2 to the input of the DLL requires a TSV. The impedance of this TSV introduces a delay offset, as illustrated in Fig. 7. Since the synchronous scheme is used for data that travels from the bottom tier to the top tier, this offset produces positive clock skew [3], where the clock signal arrives at the source register before the clock signal arrives at the destination register [2]. To ensure that no hold violation is produced, the delay of the data path must be greater than the delay offset created by the feedback TSV. This condition is satisfied in the linear clock tree topology since the data lines travel longer distances and include the TSV impedance, as illustrated in the close path and far path shown in Fig. 2.

The inputs of the DLL are determined by evaluating the skew of the most extreme paths, close1 to close2 and far1 to far2, when the DLL synchronizes one of these two paths. In the first case, the reference clock signal is close1 and the feedback clock signal is close2. In the latter case, the reference clock signal is far1 and the feedback clock signal is far2. The combination of inputs that leads to similar skew for the two extreme paths is preferable as all clock paths in the clock tree experience similar skews. These results are illustrated in Fig. 8. The filled columns represent the skew between far1 and far2, while the empty columns represent the skew between close1 and close2. Note that synchronizing far1 and far2 leads to a similar skew for the extreme paths. As described in Section 4.1, this similarity in clock skew is a product of the linear topology. Reducing the clock skew for one clock path also reduces the clock skew for the other paths. Note that the clock skew illustrated in Fig. 8 is negative [3] since the clock skew is measured from far1 to far2 and from close1 to close2. For the remaining simulations, the DLL synchronizes the far1 to far2 path.



(a)

(b)

Fig. 7. Three-dimensional DLL with TSV along feedback path. (a) Single tier 3-D DLL. (b) TSV delay along the feedback path causes a delay offset between the input and output of the DLL.

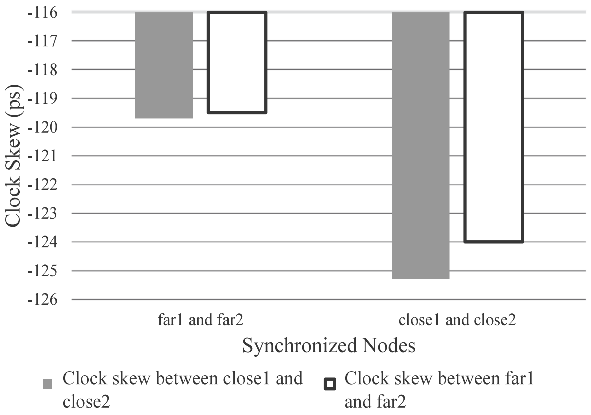


Fig. 8. Clock skew for both combinations of synchronized nodes. For each synchronization case, the clock skew of the close path (close1 to close2) and the far path (far1 to far2) are considered. Synchronizing far1 and far2 leads to a similar clock skew for the linear clock tree topology.

The maximum clock frequency achieved by the DLL is 1.7 GHz when the source tier operates in the low standby power process, and 2.6 GHz when the source tier operates in the high speed process. The process technology of the destination tier does not affect the maximum frequency achieved by the DLL.

In 3-D ICs, different tiers can be manufactured in different processes. For high speed and low standby power processes, the two tiers are

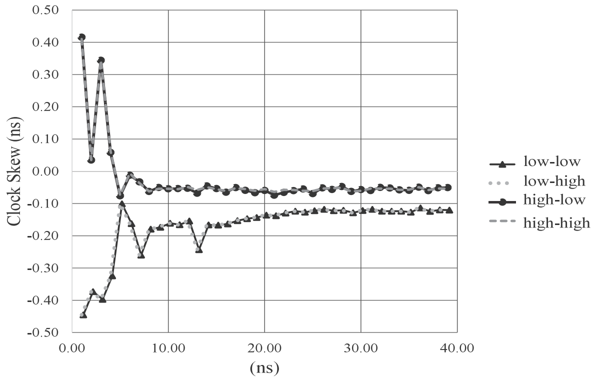


Fig. 9. Clock skew of DLL for the far-to-far clock path for different combinations of process characteristics. Four process combinations are considered, where the first process corresponds to the source tier, and the second process corresponds to the destination tier: low–low, low–high, high–low, and high–high. In all cases, the DLL is placed within the source tier.

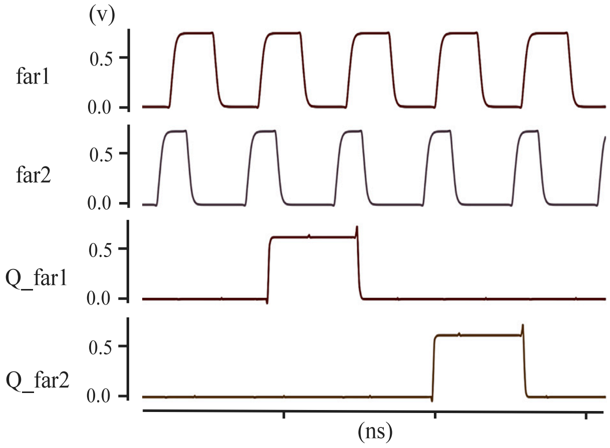


Fig. 10. Extreme source synchronous clock and data waveform for voltage supply = 0.7 V, temperature = 125 °C at 6 GHz.

evaluated for the four process combinations: low–low, high–low, low–high, and high–high. The clock skew between far1 and far2 for the synchronous scheme for different process combinations is shown in Fig. 9. Note that the skew converges to a negative value. This result is due to the delay offset produced by the TSV within the feedback path. The performance and far1-to-far2 skew of both topologies under different process combinations are listed in Table 2. The lock time metric specific to the DLL for the synchronous scheme is also listed in Table 2.

The interface in this test circuit has 512 TSVs (one for each bit) plus two TSVs for the clock signals: one to send the clock signal from the source tier to the destination tier, as shown in Figs. 2 and 4, and a second TSV for the DLL feedback path, as shown in Fig. 7(a). The pitch between the TSVs is 10 μm. The DLL requires approximately 0.8% of the area of the clock tree. The synchronous scheme does not achieve the same high frequencies as the source synchronous scheme. The relatively small area overhead, however, makes this approach an attractive alternative as compared to a second source synchronous interface, which doubles the area overhead.

The performance of the synchronous scheme depends upon the DLL used to align the two tiers. This scheme can achieve a clock speed of 1.7 GHz for the low power process and 2.6 GHz for the high speed process assuming a 10 nm technology node. Alternatively, the source synchronous topology can achieve 2.24 GHz at low power and 6.0 GHz at high speed. While the source synchronous topology can achieve over twice the speed of the synchronous topology, the area to include a second interface for the data traveling in the opposite direction is considerably larger than a single tier DLL which switches the interface from a source synchronous scheme to a fully synchronous scheme.

Table 2
Comparison of performance and area of full and source synchronous schemes.

		Process combination			
		Low–low	Low–high	High–low	High–high
DLL	Max frequency (GHz)	1.7	1.7	2.6	2.6
	Far-to-far skew (ps)	−124	−124	−53	−53
	Lock time (cycles)	24	24	10	10
	Clock TSVs	2			
Source synchronous	Max frequency (GHz)	2.2	2.5	4.7	6
	Far-to-far skew (ps)	241	222	142	126
	Clock TSVs	1			

Table 3
DLL frequency ranges due to voltage and temperature corners.

Corners			Frequency (GHz)	
Process	Voltage supply (V)	Temp (°C)	Min	Max
High speed	0.7	125	1.0	2.3
		27	1.0	2.6
		−50	1.1	2.8
	0.8	125	1.0	2.4
		27	1.0	2.6
		−50	1.1	2.8
	0.9	125	1.0	2.5
		27	1.1	2.7
		−50	1.1	2.7
Low power	0.7	125	0.6	1.2
		27	0.7	1.5
		−50	0.9	1.9
	0.8	125	0.7	1.5
		27	0.8	1.7
		−50	0.9	2.0
	0.9	125	0.8	1.7
		27	0.9	1.9
		−50	0.9	2.2

A nominal voltage of 0.8 V and a nominal temperature of 27 °C are assumed. To determine the performance of the linear clock tree for voltage and temperature corners, the linear clock tree is tested under the following corners: 0.7 V to 0.9 V and −50 °C to 125 °C. The performance of the source synchronous scheme is affected by the reduced duty cycle. The maximum frequency is, however, not significantly impacted by the extreme corners. A waveform for the extreme case of 0.7 V and 125 °C at 6 GHz is shown in Fig. 10. Note that the waveform exhibits the same behavior as shown in Fig. 5.

The fully synchronous scheme is affected by PVT corners due to the corners shifting the delay range of the DLL. The minimum and maximum frequency of the fully synchronous scheme for different corners is listed in Table 3. Note that this limitation is set by the DLL and is not due to the clock tree topology.

5. Conclusions

A linear clock tree topology that supports both source synchronous and fully synchronous schemes is proposed as a 3-D interface. The source synchronous scheme allows the interface to send data from the source tier to the destination tier at high frequencies while matching data and clock PVT variations. To support data transmission from the destination tier to the source tier, the interface switches from a source synchronous scheme to a fully synchronous scheme using a DLL to compensate for PVT. The area of the DLL is less than the area of a separate source synchronous interface. Beneficial positive clock skew occurs due to the delay offset from the TSV along the feedback path. The skew is beneficial since the clock signal delay is smaller than the delay of the data lines. The source synchronous scheme can achieve a clock frequency ranging up to 6 GHz, while the synchronous scheme can achieve a clock frequency ranging up to 2.6 GHz.

CRediT authorship contribution statement

Andres Ayes: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Visualization. **Eby G. Friedman:** Resources, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: University of Rochester reports financial support was provided by QUALCOMM Inc.

Data availability

Data will be made available on request.

References

[1] V.F. Pavlidis, I. Savidis, E.G. Friedman, Three-Dimensional Integrated Circuit Design, second ed., Morgan Kaufmann, 2017.

[2] E. Salman, E.G. Friedman, High Performance Integrated Circuit Design, McGraw-Hill Publishers, 2012.

[3] E.G. Friedman, Clock distribution networks in synchronous digital integrated circuits, Proc. IEEE 89 (5) (2001) 665–692.

[4] E.G. Friedman, Clock distribution design in VLSI circuits - An overview, in: Proceedings of IEEE International Symposium on Circuits and Systems, 1993, pp. 1475–1478.

[5] M. Singh, S.M. Nowick, MOUSETRAP: High-speed transition-signaling asynchronous pipelines, IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 15 (6) (2007) 684–698.

[6] I. Savidis, E.G. Friedman, Electrical characterization and modeling of 3-D vias, in: Proceedings of the IEEE International Symposium on Circuits and Systems, 2008, pp. 784–787.

[7] I. Savidis, E.G. Friedman, Closed-form expressions of 3-D via resistance, inductance, and capacitance, IEEE Trans. Electron Devices 56 (9) (2009) 1873–1881.

[8] V.F. Pavlidis, I. Savidis, E.G. Friedman, Clock distribution networks for 3-D integrated circuits, in: Proceedings of the IEEE Custom Integrated Circuits Conference, 2008, pp. 651–654.

[9] V.F. Pavlidis, I. Savidis, E.G. Friedman, Clock distribution networks for 3-D integrated circuits, IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 19 (12) (2011) 2256–2268.

[10] C.-C. Chung, C.-Y. Hou, All-digital delay-locked loop for 3D-IC die-to-die clock synchronization, in: Proceedings of the International Symposium on VLSI Design, Automation and Test, 2014, pp. 1–4.

[11] M. Sadi, S. Kannan, L. England, M. Tehranipoor, Design of a digital IP for 3D-IC die-to-die clock synchronization, in: Proceedings of the IEEE International Symposium on Circuits and Systems, 2017, pp. 1–4.

[12] M.E. Quchani, M. Maymandi-Nejad, Design of a low-power linear SAR-based all-digital delay-locked loop, in: Proceedings of the Iranian Conference on Electrical Engineering, 2019, pp. 118–124.

[13] S. Bang, K. Han, A.B. Kahng, V. Srinivas, Clock clustering and IO optimization for 3D integration, in: Proceedings of the ACM/IEEE International Workshop on System Level Interconnect Prediction, 2015, pp. 1–8.

[14] Available from:http://ptm.asu.edu. [Online].

[15] Etienne Sicard, Introducing 14-Nm FinFET Technology in Microwind, 2017.

[16] Y. Zhang, X. Zhang, M.S. Bakir, Benchmarking digital die-to-die channels in 2.5-D and 3-D heterogeneous integration platforms, IEEE Trans. Electron Devices 65 (12) (2018) 5460–5467.