GATE-SiP: Enabling Authenticated Encryption Testing in Systems-in-Package

Galib Ibne Haidar, Kimia Zamiri Azar, Hadi M Kamali, Mark Tehranipoor, Farimah Farahmandi
University of Florida, Gainesville, Florida, USA, 32611
{ghaidar,k.zamiriazar,ffarahmandi}@ufl.edu,tehranipoor@ece.ufl.edu,hadi.mardanikamali@ucf.edu

ABSTRACT

A heterogeneous integrated system in package (SIP) system integrates chiplets outsourced from different vendors into the same substrate for better performance. However, during post-integration testing, the sensitive testing data designated for a specific chiplet can be blocked, tampered or sniffed by other malicious chiplets. This paper proposes GATE-SiP which is an authenticated partial encryption protocol to enable secure testing. Within GATE-SiP, the sensitive testing pattern will only be sent to the authenticated chiplet. In addition, partial encryption of the sensitive data prevents data sniff threats without causing significant penalties on timing overhead. Extensive simulation results show the GATE-SiP protocol only brings 6.74% and 14.31% on area and timing overhead, respectively.

KEYWORDS

SiP, 2.5/3D, chiplets, authentication, partial encryption, secure test.

ACM Reference Format:
Galib Ibne Haidar, Kimia Zamiri Azar, Hadi M Kamali, Mark Tehranipoor, Farimah Farahmandi. 2024. GATE-SiP: Enabling Authenticated Encryption Testing in Systems-in-Package. In *61st ACM/IEEE Design Automation Conference (DAC '24), June 23–27, 2024, San Francisco, CA, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3649329.3656527

1 INTRODUCTION

The heterogeneous integration process enables the system in package (SiP) to integrate chiplets that are with different functionalities and are fabricated in different foundries on the same substrate. SiP can also enhance the overall yield rates and performance with low cost [21][11]. However, the chiplets sourced from different vendors are usually black boxes to the SiP designer. Hence, post-integration testing is important to ensure the quality of packaged SiP. To facilitate enhanced performance in terms of area, speed, and cost, 2.5D and 3D advanced packaging techniques are usually adopted. In 3D and 2.5D architecture, chiplets are stacked vertically and placed horizontally on top of an interposer respectively. However, in both cases, only the base die/chiplet will have access to board-level pins according to the established IEEE 1838 standard to facilitate post-integration testing [3]. Hence, when the testing patterns designated for upper-tier chiplet are sent through board-level pins, malicious middle-tier chiplets can perform man-in-the-middle, tampering, sniffing, and spoofing attacks during testing to acquire sensitive data and gain access to the internal design of individual chiplets through the scan chains [14], which damages the confidentiality, integrity, and availability (CIA) of sensitive testing patterns [8] [22] [13] [10].

Over the years, various solutions have been proposed to solve security issues in the 2.5D/3D SiP domain. However, there has been minimal research focused on securing the design during the post-integration testing phase. Logic locking [7] is one of the promising solutions to prevent reverse engineering and IP piracy. However, during post-integration testing, the key has to be transmitted to the designated chiplet for scan chain access. Any hostile chiplet inside the SiP can sniff this key during transmission and later use it to gain illegal access, demonstrating the inefficiency of logic locking during post-integration testing. Hardware trojans inserted in the malicious chiplets can be triggered to tamper or sniff the testing patterns designated for other chiplets. Detecting those trojans can effectively mitigate security threats. Nevertheless, recent hardware trojan detection methods [4] [14] developed for SiP fail to detect any internal hardware trojan of a 2.5D/3D IC. Hence, constructing secure communication to pass sensitive testing patterns through protocol [17] and root-of-trust modules [20] [18] were investigated. However, the area overhead, integration complexity, and performance penalty brought by the previous research are usually significant. In addition, the capability of detecting and mitigating the threats during the post-integration testing for SiP is not completely considered either.

To address the above issues, the GATE-SiP: authenticated partial encryption protocol is proposed in this paper. In this protocol, active test patterns are not transmitted until the authentication with the intended chiplet is complete. Further, dummy patterns are mixed with the authentication patterns so that untrusted chiplets cannot differentiate between the authentication and dummy patterns. Thus, the testing patterns are hard to be blocked or tampered with. To prevent the testing patterns from leaking sensitive data to untrusted chiplets, encryption is also applied to the test setup patterns only. The scan chain patterns are transmitted unencrypted, as they do not contain sensitive data. Hence the timing overhead caused by encryption would be reduced. We also perform the experiment to imitate the GATE-SiP protocol in different scenarios. The simulation results show that the proposed GATE-SiP incur at most 6.71% and 14.31% on area and timing overhead, respectively. The contribution of this paper is summarized as follows:

- A runtime evaluation protocol with adaptive control to tester to detect security breaches.
- An authentication protocol to prevent man-in-the-middle attacks.

DAC '24, June 23–27, 2024, San Francisco, CA, USA

ACM ISBN 979-8-4007-0601-1/24/06…$15.00
https://doi.org/10.1145/3649329.3656527

- A partial encryption protocol to prevent tampering, sniffing, and spoofing attacks.

2 BACKGROUND

2.1 Threat Model

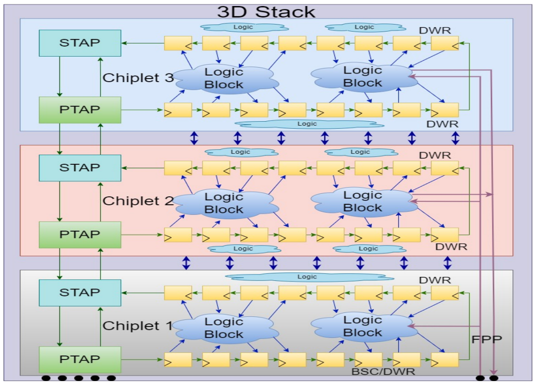

Figure 1: IEEE 1838 standard test infrastructure for 3D ICs

The IEEE 1838 standard was first introduced in 2020 to standardize the on-chip hardware components to enable testing in the 3D heterogeneous chiplet ecosystem [3], and similar testing infrastructure can also be applied on a 2.5D system. Instead of one TAP (Test Access Port) like its predecessor IEEE 1149.1 [2], this standard specifies the use of two TAPs, one primary TAP (PTAP) and one secondary TAP (STAP). Take 3D architecture shown in Fig. 1 as an example, only the bottom chiplet's PTAP will be connected to the board-level pins, and the PTAPs of other chiplets will be connected to the STAPs of the previous chiplets. Hence, in an SiP assembled with 2.5D/3D packaging techniques, the chiplets positioned higher in the hierarchy must receive test data from the lower ones. In this paper, it is assumed that the SiP designer and test engineers are trusted, and the SiP designer receives chiplets from both trusted and untrusted sources as shown in Fig. 2. Under this threat model, chiplets outsourced from untrusted vendors have the capability of breaching the security of higher-stacked chiplets. Assuming chiplet 2 in Fig. 1 to be malicious, the following attacks can be performed during testing:

2.1.1 *Man in the middle attack.*

As test data intended for chiplet 3 must go through the STAP of chiplet 2, chiplet 2 can block/loop back the test data to the board-level pins without transmitting it to chiplet 3. As a result, there will be a mismatch between the expected and generated response, which renders the chiplet 3 faulty and tarnishes the reputation of the chiplet 3 vendor in the process as well. In addition, the untrusted chiplet can pretend to be the tester to authenticate with the chiplet 3. In that case, the internal logic can be illegally accessed by untrusted chiplet 2.

2.1.2 *Tamper attack.*

Further, chiplet 2 can modify the output/input test data intended for chiplet 3. If the input test data is modified, then it will not generate the response expected by the tester. Similarly, if the output test data is modified by chiplet 2, it will not match the response expected by the tester. Again, the tester will render the chiplet 3 to be faulty without realizing the underlying security breach.

2.1.3 *Sniffing and spoofing attack.*

During post-integration testing for chiplet 3, sensitive information such as encryption-decryption keys, and scan chain access sequence, needs to be transmitted. Chiplet 2 can sniff and develop the secret data by monitoring the connection between TAP in different chiplets, and that information can be stored and used for gaining unauthorized access.

2.2 Related Work

Researchers have developed creative approaches to establish a secure ecosystem for chiplet-based 2.5D/3D SiP. Patanjali et al. proposed the trust enforcing entity (TREE) modules, security wrappers, and scan protection modules to manage security assets against different threats in [18]. Sami et al. proposed the use of POCA (Power on Chip Authentication) protocol to establish authenticated communication with the intended chip [17]. Although these protocols are effective against man-in-the-middle attacks, they cannot resist tampering, sniffing, and spoofing attacks as mentioned in section 2.1. Further, SST and CSST methods were proposed to establish a root-of-trust module in the test floor [5][15]. Other root-of-trust modules utilizing an active interposer [14] and monitoring plane [20] are also proposed. However, these root-of-trust modules fall short of addressing all the security threats mentioned in section 2.1. Additionally, significant overhead on time and area makes them hard to implement in real SiP. According to the above discussion, to the best of our knowledge, the proposed GATE-SiP is the first protocol that completely protects the post-integration testing process for 2.5D/3D SiP without bringing significant penalties on time and area overhead.

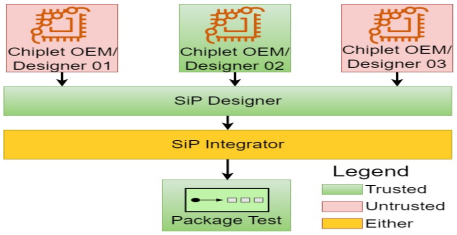

Figure 2: SiP supply chain

3 GATE PROTOCOL

3.1 Assumptions

To create a secure post-integration environment, the protocol relies on the following assumptions:

- During die/chiplet level testing the chiplet designers need to set specific PUF activation conditions, scan-chain lengths, and IDs to the internal PTAP registers of individual chiplets. In addition, a database (DB_{crp}) containing PUF activation conditions, challenge-response, and authentication stimuli-response pairs based on chiplet ID should be generated by chiplet designers. Due to the property of the PUF, chiplets with different IDs would have different challenge-response pairs.

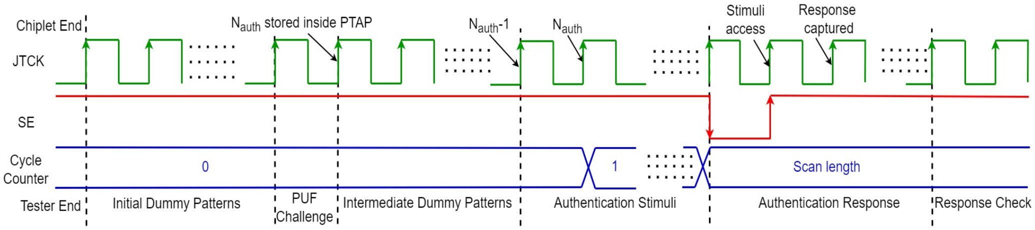

Figure 3: Authentication protocol

- The dataset that is to be created by the chiplet designer during die/chiplet level testing needs to be securely shared with the trusted SiP designer and test engineers before the post-integration testing starts. However, scan chain length that includes the proprietary information will not be shared. An example database from chiplet designer containing two different IDs is shown in Table 1.
- The pins used for scanning out the PUF responses and setting the PUF activation conditions will be disconnected after die/chiplet level testing so that any adversary can not use the pins to gain access to the inside logic of chiplet [16].

Table 1: Example of DB_{crp} database for GATE protocol

Chiplet ID	Activation condition	PUF		Authentication	
		Challenge	Response (N_{auth})	Stimuli	Response
16'h7f6d	sec_bit[3:0] =4'h9	16'h1433	16'd1173	011...110	101..100
		16'he09a	16'd741	010...101	000...110
		16'hbd1a	16'd4537	011...000	110...010
		16'h81ab	16'd51	001...101	100...110
16'h3a92	sec_bit[3:0] =4'hb	16'h1433	16'd1571	011...001	100...101
		16'he09a	16'd2603	111...101	000...110
		16'hbd1a	16'd353	101...011	100...111
		16'h81ab	16'd4037	110...111	110...001

3.2 Chiplet architecture

Our proposed chiplet architecture requires the integration of a custom PTAP, a PUF, and an ASCON cipher to enable secure post-integration testing as illustrated in Fig. 4. The standard PTAP has been modified with custom logic for controlling on-die PUF activation, authentication, and partial encryption protocols. In addition, the parameters used for authentication are also stored in the internal registers as shown in Fig. 4. The lightweight ASCON cipher is included in the design to enable on-chiplet secure encryption-decryption while incurring minimum area overhead [6]. Lastly, the PUF is integrated with the chiplet to generate the authentication cycle number and the crypto parameters (key, associated data, and nonce) for the ASCON cipher. The use of PUF enhances the IP security of individual chiplets further. Because the inherent property of PUF assures diverse responses to the same challenge across various chiplets [19], even if an adversary gets access to the challenge-response pairs of one chiplet sourced from one specific vendor, other chiplets of that vendor will not be compromised.

3.3 Authentication Protocol

To prevent test data block/loopback attacks performed by the untrusted middle-tier chiplets, initially authenticated communication with the intended chiplet is established. Further, to confuse the hostile chiplets, authentication patterns are mixed with dummy patterns. In addition, the custom PTAP does not allow dummy patterns to gain access to the internal logic, only the authentication patterns gain access. As the access to the internal design is decided inside the chiplet, it becomes difficult for the untrusted chiplets to differentiate between the dummy and authentication test patterns and thus, unable to gain access to the internal logic. The detailed authentication protocol is listed as follows:

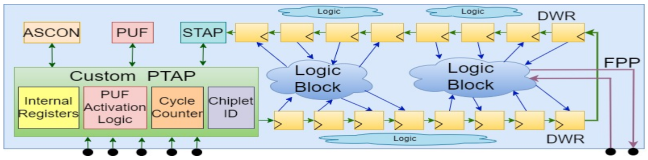

Figure 4: Proposed chiplet architecture for enabling secure testing

Initial dummy patterns: At the onset of the testing process, several dummy test patterns will be applied to confuse the untrusted chiplets so that they cannot differentiate between dummy and authentication patterns.

PUF challenge: Then to activate the on-chiplet PUF, the stimuli containing the activation condition needs to be sent to its PTAP according to the database DB_{crp} as shown in Table 1. At the same time, the PUF challenge is also transmitted to PTAP. The response of the activated PUF is the authentication cycle number, N_{auth}, and it will be collected by PTAP and stored in its registers. It means that PTAP will expect dummy patterns for another $N_{auth}-1$ clock cycles. At the exact N_{auth} cycle, authentication patterns will begin to shift in. Due to the pre-shared database (DB_{crp}), N_{auth} is common information between the chiplet and SiP tester. Thus, the SiP tester will transmit the authentication patterns at the exact N_{auth} cycle.

Intermediate dummy patterns: After establishing the number of N_{auth} through PUF, for the following $N_{auth} - 1$ clock cycles, dummy patterns are kept sending from the tester. The scan enable (SE) signals of the scan chain registers will be asserted high by the PTAP. Hence, the test patterns shifted in via the scan chains will simply be shifted out without gaining access to the internal design for testing.

Authentication stimuli: At N_{auth} cycles, authentication stimuli from the DB_{crp} database will begin to be sent by the tester and shifted in through the scan chains. According to the assumptions, the scan chain length is stored inside the PTAP internal registers. A counter in PTAP will start counting the number of stimuli bits that are being shifted into the scan flops. As a result, when the counter reaches the scan chain length, all the scan flops will contain the appropriate stimuli bits. Then, the PTAP will set '0' to the SE signal to enable the internal logic access.

Authentication response: At the next clock cycle, the PTAP will assert the SE signal again. A unique response to the stimuli caused by internal logic is to be captured by the scan cells and then shifted out through the scan chains.

Response check: The scanned-out responses are compared against the expected golden responses. The expected responses are determined during the die/chiplet level testing and stored in the shared database (DB_{crp}). If the responses do not match, then the authentication error flag is set. Further, to avoid the untrusted chiplets aware of the man-in-the-middle attack detection, several dummy patterns are sent before stopping the test.

3.4 Partial Encryption and Embedded Authentication Protocol

Once the chiplet has been authenticated, the actual structural test patterns can be applied for testing. However, the sniffing, spoofing, and tampering attacks are still threats to the structural test patterns. To reduce the overhead, a partial encryption procedure based on the ASCON algorithm [6] is proposed in this paper, and the comprehensive description is provided below:

Step 1: The PUF inside the target testing chiplet is activated by sending the PUF activation pattern according to the shared database DB_{crp}. At the same time, the challenges are also delivered to the PUF to generate the responses.

Step 2: The responses of the PUF based on the received challenges from the tester are utilized as the crypto-parameters (key, associated data, and nonce) for ASCON cipher. These parameters will be stored inside the internal registers of the ASCON cipher for future encryption-decryption processes. As for the tester, by accessing the DB_{crp} database shared by the chiplet designer, the same crypto parameters can be acquired as well. Hence, the test patterns can be securely transferred between the tester and the chiplet.

Step 3: A general structural test pattern consists of two parts, the test setup and scan chain pattern. At first, the test setup patterns are sent to set up the chiplet for testing (configuring the scan chains, accessing various IP blocks, etc.). Then, the scan chain patterns are sent to detect various faults such as stuck-at, transitional, coupling, and so on. So, to prevent unauthorized access to the internal logic of the target chiplet without causing significant complexity overhead, this protocol only asks the tester to encrypt the test setup patterns. On chiplet, these patterns are decrypted using the internally stored crypto parameters. Then the decrypted test setup patterns configure the target chiplet for the scan chain test.

Step 5: After the scan chain patterns are applied on the SiP, the test response is compared with the expected golden response. If the responses do not match, then embedded authentication stimuli as shown in table 1 is applied to differentiate whether the mismatch comes from the structural flaw inside the internal logic or the temper attack on test patterns. If there is no mismatch for the authentication pattern, it indicates that the previous scan chain pattern had failed due to structural defect. Then, the tester will return to applying rest of the scan chain patterns. However, mismatch in the authentication response signals a temper attack on the test patterns. Hence, the authentication error flag is raised, and the test execution is stopped after applying several dummy patterns.

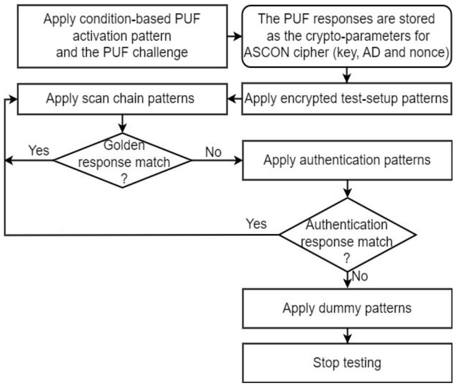
Apply condition-based PUF activation pattern and the PUF challenge
The PUF responses are stored as the crypto-parameters for ASCON cipher (key, AD and nonce)
Apply scan chain patterns
Apply encrypted test-setup patterns
Yes
Golden response match ?
No
Apply authentication patterns
Yes
Authentication response match ?
No
Apply dummy patterns
Stop testing

Figure 5: Partial Encryption and Embedded Authentication Protocol

4 SECURITY ANALYSIS

4.1 Man in the middle attack

The authentication flow has been introduced to ensure the resiliency of the test protocol against the man-in-the-middle attacks. To address the block/loopback of test data, structural test patterns are not transmitted unless the authentication is confirmed. If the authentication response to the authentication stimuli does not match with the golden response, then man-in-the-middle attack is detected. Further, dummy patterns are mixed with authentication stimuli so that the untrusted chiplets can not differentiate between authentication and dummy patterns. As for the impersonation threats, untrusted chiplets need to know the CRPs (challenge-respone pairs) generated by the on-chiplet PUF as well as the PUF activation conditions. Currently, in average, present chiplets containing more than 150 I/O pins have an average area of $200mm^2$ [9][12]. As a result, the attacker will have to go through more than 2^{150} combinations to gain access to the PUF. Consequently, the probability for the attacker to activate the PUF becomes extremely low. Thus, the proposed authentication protocol enables us to detect and prevent man-in-the-middle attacks.

4.2 Sniffing and spoofing attack

The test-setup patterns are the most crucial ones in terms of security as they contain sensitive information such as how to set up different blocks (memories, cores, etc.) of a chiplet to be tested or how to access the scan chains, the primary targets of sniffing and spoofing attacks. Scan chain patterns only contain the data to detect the stuck-at, transition, coupling, etc. faults. As they do not contain any sensitive information, there is no need for encrypting them. Thus, encryption of only test-setup patterns enables us to prevent sniffing and spoofing attacks while introducing a minimum timing overhead compared to full encryption.

4.3 Tamper attack

The protocol proposed in this paper cannot actively prevent tampering attack. However, the tampering performed by untrusted chiplets can be detected by the embedded authentication pattern within the structural test patterns. If the expected and generated responses of structural patterns do not match, then it indicates two possible scenarios. Either structural defects inside the target chiplet or malicious modification on the test stimuli/response led to the unexpected test response. So, if the mismatches occur, embedded authentication patterns are applied. As authentication patterns have passed previously, the authentication response should match the expected golden response. Hence, if there is no mismatch, then it indicates that the reason of failed test is the structural defects. But in the case of mismatches, it signifies that the correct test stimuli/responses are either not getting transmitted/received from the intended chiplet and thus, indicating tamper attacks. So, the authentication error flag is raised and the test execution is stopped.

5 RESULTS

Our proposed test protocol was implemented on the SiP architecture provided by Cadence to facilitate training on the 2.5D/3D test infrastructure. The original SiP architecture consists of 3 dies/chiplets in a 3D stack. The chiplets were synthesized using the GSCLIB045 standard cell library through Cadence's Genus tool. During synthesis, DFT architectures such as JTAG, TAP, WIR, and SIBs were added and scan chains were stitched up to ensure that the chiplets were compatible with the IEEE 1838 standard. However, for creating a realistic hostile testing environment such as figure 1 and enabling secure testing, only the top chiplet was customized with an ASCON cipher, a 16-bit arbiter PUF, and additional logic inside the PTAP and JTAG module. Although the GATE protocol was implemented in a 3D architecture, it is also applicable to 2.5D architectures as it is compatible with the IEEE 1838 standard.

Table 2: Area Comparison

Design	Cell count	Cell area (mm^2)	Net Area (mm^2)	Total Area (mm^2)	Area overhead (%)
Original chiplet	17817	0.75	0.024	0.774	-
Proposed chiplet architecture	35481	0.786	0.04	0.826	6.74

5.1 Area Overhead

The gate count, cell area, net area, and total area of both the original and proposed chiplet designs after synthesis have been presented in table 2. Our implementation incurs a mere 6.74% increase in area overhead while establishing a secure test environment. In comparison, the area overhead associated with TREEHOUSE [18] and POCA protocol [17] are <1% and 0.157% respectively. Although these protocols' area overhead is less than ours in comparison, they fail to protect the chiplets against tamper, sniffing, and spoofing attacks. Further, the POCA protocol assumes that the required components of the protocol such as TRNG, ECDH, and hash modules are integrated into the design by default and thus, not included in area overhead. The SST [5] and CSST [15] implementations demonstrate an area overhead ranging from 0.020-2.9% and 0.033-1.42%, considering designs consisting of 1-100 million gates. As chiplets are being adopted to facilitate lower gate count during fabrication, it is highly improbable that these methods will be embraced by chiplet designers due to high overhead of these implementation on smaller gate count designs. Moreover, the root-of-trust module proposed by Mohammed et al. can cause the area overhead to rise to 47.35% [14] and this implementation is only applicable to 2.5D architectures whereas our GATE protocol is compatible with both 2.5D and 3D implementation.

5.2 Timing Overhead

The initial dummy, authentication, encrypted test setup, and scan chain patterns were generated in STIL and Verilog format by Cadence's ATPG tool Modus. However, due to the lack of tool support, the authentication patterns embedded within the scan chain pattern had to be added manually by utilizing the statements of the IEEE 1450.1 standard [1]. Further, the Verilog testbench was modified accordingly to imitate the behavior of IEEE 1450.1 standard statements during simulation.

Table 3: Authentication protocol timing overhead

PUF Challenge	Authentication Cycle	Timing overhead (%)
16'h0000	0	0
16'h9E1A	751	0.17
16'hC147	38415	5.41
16'h309F	65536	11.67

5.2.1 Authentication Protocol. The time needed by the authentication protocol T_{auth} is equal to $(N_{dpi}+L_{scan}*2+N_{auth}+N_{cap})*T_{tck}$, where N_{dpi}, N_{auth}, N_{cap}, and T_{tck} represent the number of initial dummy test patterns, authentication cycle, capture cycle, and test clock period respectively. In our experiment, a 16-bit arbiter PUF is implemented for authentication. Based on the challenge, the PUF response can vary from 0 to 65536, causing a timing overhead ranging from 0-11.67% as shown in Table 3. Thus, an optimum challenge should be provided during the test to avoid significant timing overhead while ensuring secure authentication. For example, as shown in table 3, a challenge of 16'hC147 and 16'h9E1A generates an authentication cycle of 38415 and 751 respectively. The authentication cycle 38415 provide higher security in terms of confusing the untrusted chiplets compared to 751. However, in a real testing scenario, 751 clock cycles of dummpy patterns are sufficient to hide the active authentication patterns and difficult for attacker to block test information. Considering challenge 16'hC147 introduces higher timing overhead, the optimum solution would be to use the 16'h9E1A as the challenge since it provides moderate security with low (0.17%) overhead.

5.2.2 Parital Encryption Protocol. To measure the timing overhead and the efficiency brought by the partial encryption procedure, we applied the test patterns on an individual chiplet design through simulation with no encryption, full encryption and the proposed partial encryption configurations. Further, to see the results in a full SiP scenario, an SiP containing three chiplets was also tested using above configurations. The timing overhead associated with the above experiments are presented in Table 4. The actual time needed for the testing is dependent on the clock frequency set by the test engineer. The test clock frequency was set as 20MHz and the time shown in the Table 4 is the multiplication results of the clock period and the number of clock cycles required to run the test. According to the previous discussion, full encryption test encrypts all the test patterns, and partial encryption only encrypts test setup patterns. As the ratio of scan and test-setup pattern is larger for SiP than individual chiplets, our partial encryption procedure introduces even less timing overhead during testing of the SiP, which is 14.31%. Hence, with more chiplets integrated in an SiP, the timing overhead brought by our proposed partial encryption will be even less. Additionally, comparing the time needed by partial encryption with the full encryption, the test time is significantly reduced no matter under individual chiplet or SiP scenarios.

In comparison to the timing overhead (2.65ms) introduced by our GATE protocol, the implementations of TREEHOUSE [18] and POCA protocol [17] introduce timing overhead of 1284.5ps and 28700ns (1619 cycles) respectively. However, they fail to provide complete protection against all the attack vectors specified in the section 2.1. Additionally, the timing overhead of the SST [5] implementation can rise to 32.5%, reducing the likelihood of its utilization in post-integration testing.

Table 4: Partial encryption timing overhead

Design	No Encryption	Full Encryption		Partial Encryption	
	Test time (ms)	Test time (ms)	Overhead (%)	Test time (ms)	Overhead (%)
Chiplet	18.54	55.84	201.2	21.48	15.89
SiP	38.37	65.7	71.22	41.02	14.31

6 CONCLUSION

Our proposed GATE protocol provides a complete solution to the security threats associated with post-integration testing while incurring low area (6.74%) and timing overhead (chiplet - 15.89%, SiP - 14.31%). Further, it seamlessly compatible with the IEEE 1838 standard, enabling its use as an extension to the standard for secure post-integration 2.5D and 3D testing. In the future, our protocol will be upgraded to solve more complicated security issues.

7 ACKNOWLEDGMENTS

This work was supported by the Office of Naval Research through projects SVH (award number: P0261180) and SALEM (award number: P0326838).

REFERENCES

[1] 2005. IEEE Standard for Extensions to Standard Test Interface Language (STIL) (IEEE Std 1450-1999) for Semiconductor Design Environments. *IEEE Std 1450.1-2005* (2005), 1–123. https://doi.org/10.1109/IEEESTD.2005.97746

[2] 2013. IEEE Standard for Test Access Port and Boundary-Scan Architecture. *IEEE Std 1149.1-2013 (Revision of IEEE Std 1149.1-2001)* (2013), 1–444. https://doi.org/10.1109/IEEESTD.2013.6515989

[3] 2020. IEEE Standard for Test Access Architecture for Three-Dimensional Stacked Integrated Circuits. *IEEE Std 1838-2019* (2020), 1–73. https://doi.org/10.1109/IEEESTD.2020.9036129

[4] Soha Alhelaly, Jennifer Dworak, Theodore Manikas, Ping Gui, Kundan Nepal, and Alfred L. Crouch. 2017. Detecting a trojan die in 3D stacked integrated circuits. In *2017 IEEE North Atlantic Test Workshop (NATW)*. 1–6. https://doi.org/10.1109/NATW.2017.7938027

[5] Gustavo K. Contreras, Md. Tauhidur Rahman, and Mohammad Tehranipoor. 2013. Secure Split-Test for preventing IC piracy by untrusted foundry and assembly. In *2013 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFTS)*. 196–203. https://doi.org/10.1109/DFT.2013.6653606

[6] Eichlseder M. Mendel F. et al Dobraunig, C. 2021. Lightweight Authenticated Encryption and Hashing. *Journal of Cryptology* 34 (2021). https://doi.org/10.1007/s00145-021-09398-9

[7] Jaya Dofe, Chen Yan, Scott Kontak, Emre Salman, and Qiaoyan Yu. 2016. Transistor-level camouflaged logic locking method for monolithic 3D IC security. In *2016 IEEE Asian Hardware-Oriented Security and Trust (AsianHOST)*. 1–6. https://doi.org/10.1109/AsianHOST.2016.7835570

[8] Jaya Dofe, Qiaoyan Yu, Hailang Wang, and Emre Salman. 2016. Hardware security threats and potential countermeasures in emerging 3D ICs. In *2016 International Great Lakes Symposium on VLSI (GLSVLSI)*. 69–74. https://doi.org/10.1145/2902961.2903014

[9] Alexander Graening, Saptadeep Pal, and Pankaj Gupta. 2023. Chiplets: How Small is too Small? *2023 60th ACM/IEEE Design Automation Conference (DAC)* (2023), 1–6. https://api.semanticscholar.org/CorpusID:12106011

[10] Syed Rafay Hasan, Siraj Fulum Mossa, Omar Sayed Ahmed Elkeelany, and Falah Awwad. 2015. Tenacious hardware trojans due to high temperature in middle tiers of 3-D ICs. In *2015 IEEE 58th International Midwest Symposium on Circuits and Systems (MWSCAS)*. 1–4. https://doi.org/10.1109/MWSCAS.2015.7282148

[11] M. Shafkat M. Khan, Chengjie Xi, Md Saad Ul Haque, Mark M. Tehranipoor, and Navid Asadizanjani. 2023. Exploring Advanced Packaging Technologies for Reverse Engineering a System-in-Package (SiP). *IEEE Transactions on Components, Packaging and Manufacturing Technology* 13, 9 (2023), 1360–1370. https://doi.org/10.1109/TCPMT.2023.3311801

[12] Jinwoo Kim, Gauthaman Murali, Heechun Park, Eric Qin, Hyoukjun Kwon, Venkata Chaitanya Krishna Chekuri, Nihar Dasari, Arvind Singh, Minah Lee, Hakki Mert Torun, Madhavan Swaminathan, Madhavan Swaminathan, Saibal Mukhopadhyay, Tushar Krishna, and Sung Kyu Lim. 2019. Architecture, Chip, and Package Co-design Flow for 2.5D IC Design Enabling Heterogeneous IP Reuse. In *2019 56th ACM/IEEE Design Automation Conference (DAC)*. 1–6.

[13] Siraj Fulum Mossa, Syed Rafay Hasan, and Omar Elkeelany. 2017. Hardware trojans in 3-D ICs due to NBTI effects and countermeasure. *Integration* 59 (2017), 64–74. https://doi.org/10.1016/j.vlsi.2017.03.009

[14] M. Nabeel, M. Ashraf, S. Patnaik, V. Soteriou, O. Sinanoglu, and J. Knechtel. 2020. 2.5D Root of Trust: Secure System-Level Integration of Untrusted Chiplets. *IEEE Trans. Comput.* 69, 11 (nov 2020), 1611–1625. https://doi.org/10.1109/TC.2020.3020777

[15] Md. Tauhidur Rahman, Domenic Forte, Quihang Shi, Gustavo K. Contreras, and Mohammad Tehranipoor. 2014. CSST: An Efficient Secure Split-Test for Preventing IC Piracy. In *2014 IEEE 23rd North Atlantic Test Workshop*. 43–47. https://doi.org/10.1109/NATW.2014.17

[16] Jean Rolt, Giorgio Di Natale, Marie-Lise Flottes, and Bruno Rouzeyre. 2012. *On-Chip Comparison for Testing Secure ICs.*

[17] Md Sami Ul Islam Sami, Fahim Rahman, Adam Cron, Dale Donchin, Mike Borza, Farimah Farahmandi, and Mark Tehranipoor. 2021. POCA: First Power-on Chip Authentication in Untrusted Foundry and Assembly. In *2021 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. 124–135. https://doi.org/10.1109/HOST49136.2021.9702285

[18] Patanjali SLPSK, Sandip Ray, and Swarup Bhunia. 2023. TREEHOUSE: A Secure Asset Management Infrastructure for Protecting 3DIC Designs. *IEEE Trans. Comput.* 72, 8 (2023), 2306–2320. https://doi.org/10.1109/TC.2023.3248269

[19] G. Edward Suh and Srinivas Devadas. 2007. Physical Unclonable Functions for Device Authentication and Secret Key Generation. In *2007 44th ACM/IEEE Design Automation Conference*. 9–14.

[20] Jonathan Valamehr, Timothy Sherwood, Ryan Kastner, David Marangoni-Simonsen, Ted Huffmire, Cynthia Irvine, and Timothy Levin. 2013. A 3-D Split Manufacturing Approach to Trustworthy System Development. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 32, 4 (2013), 611–615. https://doi.org/10.1109/TCAD.2012.2227257

[21] Nidish Vashistha, Latifur Rahman, Md Saad Ul Haque, Azim Uddin, Md Sami Ul Islam Sami, Islam Sami, Amit Mazumder Shuvo, Paul Calzada, Farimah Farahmandi, Navid Asadizanjani, Fahim Rahman, and Mark Tehranipoor. 2022. ToSHI -Towards Secure Heterogeneous Integration: Security Risks, Threat Assessment, and Assurance. https://doi.org/10.13140/RG.2.2.25507.12329

[22] Zhiming Zhang and Qiaoyan Yu. 2019. Modeling Hardware Trojans in 3D ICs. In *2019 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*. 483–488. https://doi.org/10.1109/ISVLSI.2019.00093