# Security-Enhanced 3D Communication Structure for Dynamic 3D-MPSoCs Protection

Johanna Sepúlveda[1,2], Guy Gogniat[2], Ricardo Pires[1], Wang Chau[1], Marius Strum[1]
[1]Microelectronics Laboratory LME, University of São Paulo, Brazil
[2]Information and Communication Science and Technology Laboratory Lab-STICC, Université Bretagne Sud, France
{jsepulveda, rpires, jcwang, strum}@lme.usp.br, guy.gogniat@univ-ubs.fr

***Abstract*—Three-dimension Multiprocessors System-on-Chip (3D-MPSoCs) hold promises to allow the development of compact and efficient devices. By means of such technology, multiple applications are supported on the same chip, which can be mapped dynamically during the execution time. This flexibility offered by the 3D technology, also represents vulnerability, turning the 3D-MPSoC security into a challenging task. 3D *communication structures* (3D-HoCs), which combine buses and network-on-chip can be used to efficiently overcome the present 3D-MPSoC vulnerabilities. 3D-HoCs can be used to implement different security services, monitor the data exchange and isolate dangerous IPs. In this paper, we implement Quality of Security Service (QoSS) in 3D-HoC to efficiently detect and prevent attacks by means of agile and dynamic security firewalls. Such a method takes advantage of the 3D-HoC wide system visibility and critical role in enabling system operation. We evaluate the effectiveness of our approach over several 3D-MPSoCs attack scenarios and estimate their impact on the overall performance. Results show that our architecture can perform a fast detection of a wide range of attacks and a fast configuration of the different security policies.**

***Keywords—3D-MPSoC, Network-on-chip, bus; quality-of-service; security; performance.***

## I. INTRODUCTION

Three-dimension Multiprocessors System-on-Chip (3D-MPSoCs) can be attacked via hardware/software. Software attacks are responsible for 80% of the embedded security incidents [2]. In order to protect the system against software attacks, security mechanisms can be implemented at the computation or **communication structure (CS)**. Three facts turns attractive the implementation of security at the 3D-CS: i) all software attacks start with an abnormal communication; ii) 3D-MPSoCs are foreseen as communication-centric systems [2,3]; and iii) the main role of the CS in the system operation can be used for security purposes. It can detect any attempt of attack by means of system monitoring and data exchange controlling.

CS of the 3D-MPSoCs can be divided into horizontal and vertical interconnection. *Horizontal interconnections* perform the data exchange among the IPs on the same die. *Vertical interconnections* communicate different dies by means of Through-Silicon-Vias (TSVs). TSVs are conductive nails that extend out the back-side of a thinned-down die [1]. Compared to traditional wire bonds, TSVs are high-density, short and low-capacity interconnection, able to operate at high frequencies. ***Hybrid-on-Chip communication structures*** (3D-HoCs) were proposed to implement vertical and horizontal data exchange within the 3D-MPSoCs. It combines buses and Networks-on-chip (NoCs) in order to take advantage of the best of the both structures. Buses are implemented by a set of TSVs to perform the *vertical interconnection*. NoCs are used as horizontal interconnection. A NoC is an integrated network that uses routers to allow the communication among the computation structure components. Bus/NoCs are connected to 3D-MPSoCs computation components by means of interfaces. These components implement the communication protocol of the system. The final 3D-CS configuration must fulfill the performance and security requirements of all the applications mapped on the 3D-MPSoC. In order to achieve this task, communication services can be provided. Quality-of-Security-Service (QoSS) appears from the junction of Quality-of-Service (QoS) and security concepts. It customizes the security communication service by means of different protection choices, called security levels. The adoption of QoSS for 3D-HoCs allows the implementation of efficient protection.

Previous works show that two dimension CS (2D-CS) can be an efficient and effective alternative to support different security services for 2D-MPSoCs. These works establish static security boundaries around 2D-MPSoC components. However, 3D-MPSoC are characterized by their dynamicity [1,3]. Statics boundaries are not suitable anymore [4]. In addition, 3D technology offers new protection opportunities. 3D-CS characteristics allow the passive monitoring of the system, the split of the security policy among the different layers of the chip and the isolation of dangerous IPs while guaranteeing the high bandwidth and low latency communication between layers.

In this work we propose a dynamic QoSS 3D-HoC architecture to protect the 3D-MPSoCs against software attacks. Two security services, *authentication* and *access control* are included to the 3D-HoC interfaces by means of agile and dynamic security firewalls. We evaluate the efficiency and efficacy of these implementations over different 3D-HoC topologies. To the best of our knowledge, this is the first attempt to discuss and implement security at the CS of the 3D-MPSoC. The experiments were performed using a SystemC-TLM timed simulation framework. It automatically carries out performance evaluations for a wide variety of MPSoC scenarios.

978-1-4799-1132-5/13/$31.00 ©2013 IEEE

The remaining text is organized as follows: Section II presents an overview of the previous works of CS-based security implementation for MPSoC protection. Section III presents the overview of 3D-MPSoCs. Section IV presents the security at the 3D-MPSoC. Section V and Section VI present the 3D-HoC and the security enhanced 3D-HoC architecture. Section VII presents the efficacy and efficiency evaluation. Finally we present our conclusions in Section VIII.

## II. Previous Works

Two dimensions security integration at the bus and NoCs CSs is addressed at the work of [4-9]. The work in [5] presents a bus-based security implementation. It uses two blocks: 1) Security enforcement module (SEM), to monitor and control data transfers; and 2) Security enforcement interface (SEI), to filter the data transfers. A secure kernel (SK) is responsible for the secure configuration of the SEM and SEIs. The works of [6,8] integrate a table at the network interface containing the access control rules of each IP. They rule the way that the system components access over the protected device. The packets that do not satisfy the access control rules are discarded. The purpose of [7] is to prevent attacks through the verification of the source and size of the packets. This work also integrates a secure network manager component to monitor the NoC behavior. The purpose is to prevent four common NoC attacks: denial-of-service (DoS), draining, extraction of secret information and modification. The work of [4] proposes a NoC architecture composed of three modules in order to avoid code injection attacks: 1) Stack Protection Unit (SPU), to track the transactions; 2) Instruction Trace Unit (ITU), to track the instructions; and 3) Local Security Manager (LSM), to react upon the reception of an alert signal of SPU or ITU. These previous works show the main role of the CS in the security of the system. It is a powerful structure for the surveillance and detection of the SoC attacks. However, the extension of this previous 2D approaches to the 3D technology presents two main limitations. 1) they implement a single NoC security level for the entire SoC; and 2) they implement static security policies. In this work we integrate security mechanisms at the 3D-HoC in order to protect 3D-MPSOCs. The use QoSS by means of dynamic firewalls allows the implementation of dynamic security boundaries while meeting the performance requirements.

## III. 3D-MPSOC

3D-MPSoCs integrate processing and storage IPs mapped on several layers of dies. Two or more dies are fabricated separately and then combined into a single stack. The way the IPs are mapped onto the system is called organization [10]. One of the most effective 3D-MPSoC organizations is the integration of a *storage layer* atop a *computation layer*, which is in turn, arranged as computation *islands* (clusters). Each island can integrate several processors and other IPs that exchange information through two dimensions CS (2D-CS). It can be a bus or a NoC. Fig 1 shows the architecture of an island.

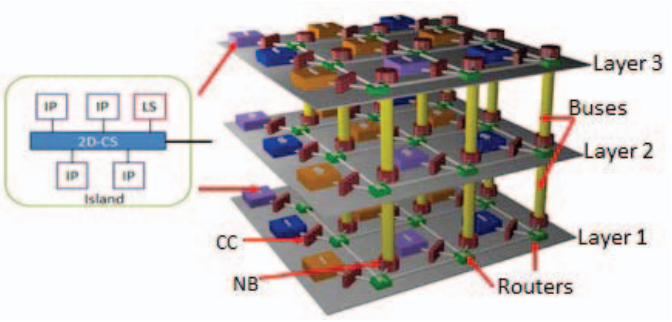


Figure 1. 3D-MPSoC architecture.

LS represent an IP that can implement the security policy that rules the interaction among the components of the island. This 2D-CS was addressed in [9] . Inter-island communication is performed by the NoC , and the island-storage communication is performed by the buses. The design trends that promote the security implementation are:

- ***Multi-application systems:*** Several applications can be mapped simultaneously in the different islands . Behavior and security requirements of the applications can be quite disparate. 3D-MPSoC should guarantee the protection of each application.
- ***Dynamicity:*** Applications mapped on the 3D-MPSoC may change during the execution time. The rate at which a new application is mapped on the system is called *dynamicity*. New applications may have different security requirements than the current one.
- ***Heterogeneity:*** Several heterogeneous components from different providers and characterized by different performances are integrated on the same chip. As a result, some islands can be weaker, from the security point of view.
- ***Observability:*** Power and thermal issues play a major role in the 3D-MPSoC design. In order to control the large and non-uniform heat propagation, monitors are embodied in the system [1]. They provide information about the workload at different points of the system, for example. Such need of system observability forces to track and keep information that can be used for an attacker to malfunction the system.

## IV. Security at 3D-MPSoCs

### A. Threat model

Attacks exploit different system vulnerabilities of the 3D-MPSoC in order to access or use the system resources without authorization [3]. There are two scenarios of infection of the 3D-MPSoCs: i) By interacting with other digital devices, a 3D-MPSoC may receive viruses (or other similar malicious pieces of code); ii) 3D-MPSoCs integrate several IP components generally provided by different suppliers. Some of these IPs may already be infected. Therefore, the protection of the 3D-MPSoC just in the I/O components is not enough. The objective of the attacks can be i) extraction of critical data

by means of unauthorized reading; ii) modification of critical data or system execution by means of unauthorized writing; and iii) denying of service by reading/writing operations whose aim to bring down the system performance. 3D-MPSoC security services implemented at the 3D-HoC can protect the system resources and data exchanges by means of communication management [8,7]. There are six security services [10]: 1) Confidentiality: ensures the data secrecy; 2) Integrity: assures that data is kept unchanged during any operation; 3) Authentication: validates the sender IP integrity; 4) Access Control: allows or denies the use of a particular resource; 5) Availability: ensures the use of the network resources; and 6) Non-repudiation: maintains evidence of 3D-HoC communication events.

A critical challenge is to extend existing 2D security countermeasures to enable the highly dynamic protection demanded by the 3D architectures. 3D-MPSoC's characteristics allow that the security can be split among its layers. distributed mechanisms can be developed and the security granularity can be increased. Each time a new application is mapped on the 3D-MPSoC, a new security policy must be implemented. Thus, according to the 3D-MPSoC organization, the security mechanisms at one or several layers must be upgraded.

### B. *Security model*

In this work we consider three attack scenarios that take advantage of the lack of the 3D-MPSoC security upgrading. However, our architecture can defend against a broader range of attacks. The security policy is defined by the security designer of the system (when the requirements of the systems are established) and a policy is associated to each application or system operation scenario. The security policy of the 3D-MPSoC can change as a consequence of three factors: i) ***New application*** is mapped on the 3D-MPSoC; ii) ***Current application is reallocated*** on the 3D-MPSoC (i.e. as a result of task migration); and iii) ***New 3D-MPSoC operation scenario*** could influence the security policy, that is, under certain conditions the security of the system may be reinforced or decreased. Security policies of the system can be loaded during: i) power up time, when all the applications that will be executed on the system are previously known; or ii) run time. At run time, the loaded security policy must come from a trusted third-party authority.

## V. Hybrid-On-Chip Communication (3D-HoC)

3D-HoCs use buses and NoCs to implement the vertical and horizontal interconnection of 3D-CS by means of buses and NoCs. Data flows through the 3D-HoCs as packets. Buses are composed by a set of wires and an arbiter, which controls the data exchange. NoCs are integrated networks that use routers and links to provide communication among the IPs. NoC router defines the path that the information must follow through the network from the initiator to the destination IP.

Electrical and geometrical concerns made the mesh-based 3D-HoCs the most popular choice in 3D-MPSoCs. ***Stacked*** uses a six-port router to integrate multiple layers, connected by mean of 2D mesh-based NoCs, through a bus spanning the entire vertical distance of the chip (Fig 1). The width of the bus can vary from 1 to $n$ wires, with $n$ typically up to 32[8]. *3D-HoCs* are notated as $L$ ($S_1$)/($S_2$|$n$), where $L$ is the number of layers, $S_1$ is the NoC size, and $S_2$ is the number of buses and $n$ is the bus width in bits. Fig 1. shows a 3D-HoC whose size is 3(3x3)/(9|32). It is used to connect 27 IPs.

## VI. Security Enhanced 3D-HoC

### A. *Quality-of-Security-Service (QoSS)*

QoSS (Quality-of-Security-Service) explores the tradeoff between the system trustfulness and its performance. The traffic of a single embedded application may integrate several flows, each of which characterized by different security requirements. QoSS concept allows differentiated treatment for the data exchange carried out through the 3D-HoC. The QoSS can be implemented by changing some local configurations of the 3D-HoC. In this work we propose the implementation of security in the CS by means of *firewalls*, which blocks or allows the completeness of a transaction according to the current security policy of the system. Security mechanisms are used to implement different levels of protection. Each *level of protection* appears from the combination of security mechanisms. A security choice represents a special configuration of the *security mechanism*. Higher security may imply in an increment of the system cost. The selection of a security level depends on the security requirements of the system, resources availability and cost. 3D-MPSoC designer may select a lower protection level in order to fulfill the performance requirements.

### B. *3D-HoC security mechanisms*

In this work we propose the implementation of two security services at the 3D-HoC: i) *authentication*, verifying the source integrity; and ii) *access control*, certifying the authorized use of the system resources. However, other services can be integrated. These security services are implemented as firewalls in the 3D-HoC interfaces. The firewalls allow or block a transaction according to the matching or mismatch between the content of the packet and the security policy. Firewalls store the security policy information in a *security table*.

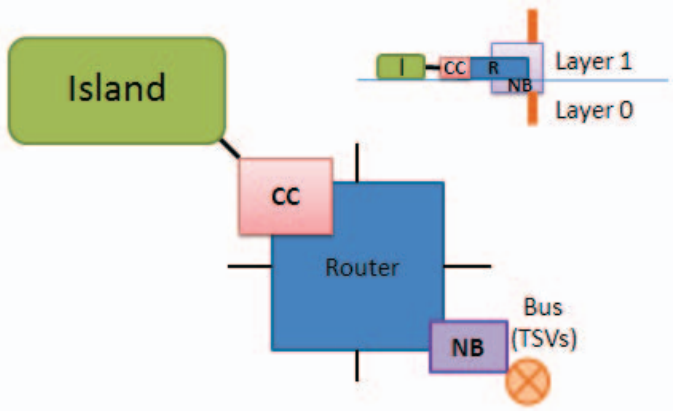


Figure 2. Firewalls of the 3D-HoC.

Valid transactions are allowed to be completed. In any other case, a notification is generated, informing the presence of a possible attack. 3D-HoCs integrates two types of interfaces: 1) Computation-Communication Interface (CCI), which links the islands/IPs and the routers; and 2) NoC-Bus (**NB**) interface between the routers and the bus. See Fig 1,2.

The security policy of the system is divided in these two types of interfaces. ***CCI*** implements the part of the security policy that rules the intra-layer communication. That is, among the components of the same layer of the 3D-MPSoC: i) interaction among the islands of the system, located in the computation layer; or ii) interaction among the storage components, located at the storage layer. Moreover, ***NB*** implements the part of the security policy that rules the inter-layer communication. That is, among the islands and the storage components. The intra-island communication can be protected by the mechanism shown in [9]. However, the focus of this paper is the security by the 3D-HoC.

Messages coming from the islands/IPs are translated by the interface into packets compliant to the protocol used within the 3D-HoC. The main requirement is that the packet must contain the information required to perform the security checking by the security mechanisms (See Section VI-*C*).
The adopted 3D-HoC packet format is composed of 11 fields. Some of them change during the transaction.

1) ***Source:*** Identifies the initiator of the communication.
2) ***Destination:*** Identifies the target of the communication.
3) ***Operation:*** Codes the transaction type. (i.e. a read, read-linked, read-exclusive, write and broadcast).
4) ***Type:*** Defines the information type that is being exchanged (i.e. data, instruction or signal types).
5) ***Role:*** Represents the role of the initiator component (i.e. user, root).
6) ***Priority:*** Expresses the priority of the packet.
7) ***Size:*** Defines the size of the payload of the packet (number of bytes).
8) ***Deadline:*** Establishes the interval in which the transaction must be performed (amount of clock cycles).
9) ***Path:*** Contains the signatures of the routers and bus arbiters used by the packet during the initiator-destination path (byte that is modified by each router of the NoC during the commutation).
10) ***ID:*** Contains the code known by an initiator/target pair (3 bits). It counts the number of transaction between an initiator/destination pair. When the ID reaches the maximum value, it restarts the count.
11) ***Payload:*** Contains the information generated by the master.

Our firewall differs from those proposed by [6,8] in the 7, 8, 9 and 10 fields. Such characteristics allow that our security mechanism avoids a wider set of attacks: i) DoS (Denial of service), by using the *deadline* field; ii) buffer overflow by the verification of the *address*, *type* and *size* fields.

### *C. 3D-HoC Firewalls*

Quality-of-Security-Service (QoSS) concept is adopted. It customizes the system protection by exploring the trade-off between the security and performance of the system. It proposes the existence of different levels of protection. Our approach defines four levels of protection (L0 to L3) for each security service: A*uthentication* and Access Control. They appear from the combination of three security mechanisms. See Table I. Note that many other levels can be defined. For higher security levels, more packet information is checked. The two security mechanisms are explained in the following paragraphs. They are implemented as firewalls embodied at the **CCI** and **NB** interfaces. The **CCI** firewall is activated when the destination of the packet injected by the island is located in the computation layer, that is, in the same layer that the initiator island. Otherwise, the packet destination is on the storage layer. In this case, **NB** will be responsible for the analysis of the packet.

**Access control - AC**

It guarantees that only authorized initiators inject authorized packets to the correct destinations. In order to perform this service, three different mechanisms were adopted. They verify a set of information contained in the packet: i) *destination*, that is, the target address; ii) *operation*, indicating the action that the initiator can perform over the destination; and iii) *deadline*, indicating the number of clock cycles in which the transaction is still valid. The correct information is stored in the *security table* of the *firewall* (**CCI** or **NB**). Access control is performed each time a packet is injected by an island (at the CC of the initiator island) or when a packet try to access to another layer of the 3D-MPSoC (at the NB which links the island to other layer). See Table I. **NB** also counts the consecutive accesses of an island to the same storage components. This is to avoid a common DoS attack whose purpose is to avoid other islands to access and modify a value in memory [8].

***Authentication - AU***

It verifies the initiator integrity. There are three security mechanisms that verify: i) *source*, that is the initiator address; ii) *path*, verifing the route of the packet inside the 3D-HoC; and iii) *ID code*, a number that identifies each transaction and that is just known by each initiator-destination pair.

### *D. Security enhanced 3D-HoC architecture*

Our proposed architecture is shown in Fig. 3. It integrates four hardware components: i) *policy keeper;* ii) *reconfiguration manager;* iii) *security mechanisms (*at the *CCI* and *NB* firewalls*);* and iv) *monitor*.

TABLE I. SECURITY MECHANISMS

| *Service* | *Mechanism* | *CCI* | *NB* | *L0* | *L1* | *L2* | *L3* |
|---|---|---|---|---|---|---|---|
| AC | Destination | Island | Memory | x | x | | x |
| | Operation | read, read-linked, write and broadcast | read, read-exclusive, write. | | x | x | x |
| | Size | No checking | Checking | | x | x | x |
| | Deadline/ role | cycles/root-user | cycles/root -user | | | x | x |
| AU | Source | Island | Island | x | x | | x |
| | Path | No checking | Checking | | x | x | x |
| | ID Code | Checking | Checking | | | x | x |

Our work supposes that the task allocation of the different applications has been previously defined.

***Policy keeper:*** It stores the information of the 3D-MPSoC task mapping and the security policy of the system (rights) of each task being executed on the 3D-MPSoC over the system resources. The security policy set the protection level (from L0 to L3) of each service (AC, AU). Each application has a security policy capable of being described by the fields of the Table I. The size of the table stored by the policy keeper component depends on the number of applications, tasks and IPs integrated at the 3D-MPSoC.

***Reconfiguration manager:*** Coordinates the upgrading of the security table of all the firewalls in order to guarantee the protection of the system.

***Security mechanisms:*** Defends the 3D-MPSoC against possible attacks. The security mechanism uses the information embodied in the packets that flow through the 3D-HoC to enforce the different security policies. See Section VI-C.

***Monitor:*** Audits the communication behavior of the MPSoC. It detects the 3D-HoC activity in order to determine the completion of the transaction among the different initiator/destination pairs. They are embodied at the routers of the 3D-HoC.

### *E. Response to attacks*

When an attack or an abnormal behavior is detected, the reconfiguration manager modifies all *security tables* inside the firewalls in order to perform two actions: i) increase the security level; and ii) isolate the target of the attack from its attacker. When the threat disappears, the system can resume to normal operation.

### *F. Operation of the system*

When the security policy of the 3D-MPSoC must be upgraded, the *reconfiguration manager* starts three procedures: *analysis of security policy*, *mechanism configuration* and *recovery*. At the *analysis of security policy procedure* the reconfiguration manager downloads the properly security policy, stored in the *policy keeper* component. It uses the 3D-MPSoC tasks mapping information to identify the **CCIs** and **NBs** (*target interfaces*) whose *security table* must be modified and its new configuration parameters. The *security mechanism* stores them at the *security tables*. In order to complete the *mechanism configuration procedure*, the *reconfiguration manager* blocks the injection of packets whose final destination is the processing component linked to the security interface that is going to be reconfigured (*target interfaces*). Such packets are stored onto the interface linked to the initiator island. The reconfiguration manager also starts a QoS (Quality-of-Service) mechanism that rises up the priority of the communication at the 3D-HoC of the packets whose final destination is the component linked to the *target interface*. That is, modify the *Priority* packet field. The QoS mechanism modifies the arbitration of the NoC routers and bus arbiters, so that, the communication of the packets with higher priorities is performed first. Once the communication of all the packets flowing to the target interface is finished, the reconfiguration of the *target interface* can be started.

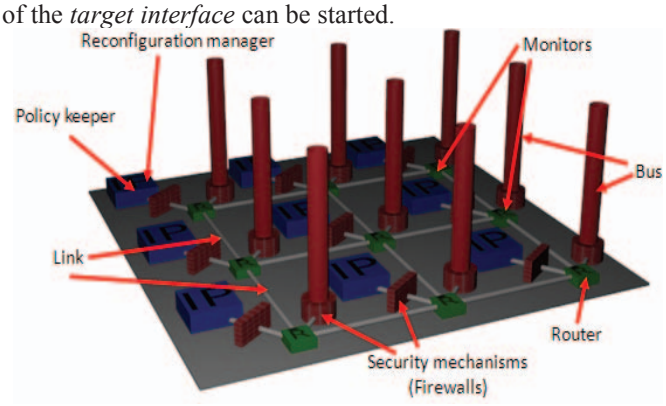


Figure 3. Security implementation at the 3D-CS.

The reconfiguration consists in the upgrading of the security tables of the **CCI** and **NB**. Note that the communication is not interrupted at the 3D-HoC during the reconfiguration. The 3D-HoC routers and buses continue communicating the remaining packets through the network. This characteristic of our architecture can reduce the latency penalties due to the reconfiguration. When the reconfiguration is finished, the final *recovery procedure* is started. The *configuration manager* frees the injection of the packets that were being blocked during the reconfiguration. The normal 3D-HoC operation is achieved when all the target interfaces are reconfigured.

## VII. RESULTS

### *A. Experimental setup*

We have developed a SystemC-TLM cycle-accurate model of the security implementation at the router and at the 3D-HoC and a 2D (*mesh*) NoC topology. We implemented our security mechanisms in both structures. For the 2D, only the **CCI** was integrated (2D-CCI). However, the *security table* integrates the values of CCI and NB. Both implementations are able to be upgraded during the execution time. The CS was employed to connect a 75-IP cores MPSoC. For comparison reasons, we employ a 3(5x5)/(25|32) 3D-Hoc and a 15x5 2D NoC. They are characterized by a XY(Z) routing scheme [4], round-robin(RR)/QoS arbiter and FIFO memory organization. The evaluation was performed by the SystemC-TLM framework that includes a set of traffic and attack generators, monitors to annotate the communication events and analysis tools that quantify a set of metrics. Power and latency models of [10] were employed. The proposed solution has been verified under five attack scenarios whose purpose is to modify or extract data or avoid the utilization of the system. It employs both traffics: i) synthetic, by mean of hot-spot, transpose, uniform traffic topologies with Poisson and LRD natures; and ii) real application, provided by MiBench benchmark suite [9]. We selected three applications: auto/industrial (*A1*), consumer electronics (*A2*) and telecommunication (*A3*). Each application is characterized by different security policies. We also explore the effect of the communication ratio (CR) on the efficiency of our architecture. CR is defined by the relation between the horizontal and vertical traffic inside the 3D-MPSoC.

### B. Results of security efficacy

Table II shows the identical security efficacy as the percentage of attacks detected by the security architecture in the 2D and 3D structuers. It was expected because the values of the security values at both alternatives were the same. The difference is that in 3D the security tables are spread along the system. The 97% security efficacy means that the security designer should increase the protection level of the access control service in order to achieve 100% of protection.

### C. Results of security efficiency

Table III presents the latency, power and area penalties for different level of protection of the 3D-HoC under uniform (U) traffic. Results show the existence of the trade-off among different levels of protection and performance. However, it is important to notice that the area penalty remains minimal. This is expected because of the 3D technology properties, which reduce area by stacking components. Fig 4 shows the comparison among the security enhanced 2DNoC and 3D-HoC architectures for different percentages of dynamicity (see Section III). Results show that 3D-HoC achieves a better performance when compared to 2DNoC. Moreover, it shows to be less sensitive to the dynamicity of the system. Such results probe that spreading the security countermeasures in the system is a better solution. Such results arise from: i) 3D technology characteristics (smaller initiator/destination paths); and ii) at the reconfiguration phase of the system (when the security tables must be upgraded), in the 3D-HoC only some small areas where blocked, so the system was able to continue its operation in the remaining areas. At the 2D mostly all the interfaces must be upgraded and therefore, blocked. Fig. 5 shows the performance results for the different protection levels of the 3D-HoC and a centralized 3D-HoC (C3D-HoC). The security checking is performed only in CC, but this security table stores the information of the CC and NB.

TABLE II. SECURITY EFFICACY

| *Attack scenario* | *2D-NoC* | *3D-HoC* |
|---|---|---|
| Write critical data | 97% | 97% |
| Read critical data | 100% | 100% |
| Malicious task migration | 100% | 100% |
| Nonexisting target /Repeated data | 89% | 89% |
| Communication target = source | 100% | 100% |

TABLE III. PENALTIES COMPARED TO 3D-HOC WITHOUT SECURITY U TRAFFIC.

| **3D-CS** | *Latency* | *Power* | *Area* |
|---|---|---|---|
| 3D-HoC L0 | 3.2% | 2.5% | 0.2% |
| 3D-HoC L1 | 4.5% | 6.4% | 0.5% |
| 3D-HoC L2 | 6.6% | 9.3% | 0.8% |
| 3D-HoC L3 | 8.3% | 10.4% | 1.2% |

TABLE IV. PENALTIES COMPARED TO 3D-HOC WITHOUT SECURITY FOR DIFFERENT TRAFFIC PATERNS AND DYNAMICITY PERCETAGES

| **Traffic characteristics** | **Dynamicity** | | | | | |
|---|---|---|---|---|---|---|
| | **30%** | | **50%** | | **70%** | |
| | *Latency* | *Power* | *Latency* | *Power* | *Latency* | *Power* |
| Hot-Spot | 9.6% | 11.3% | 10.9% | 12.1% | 11.6% | 13.6% |
| Transpose | 8.8% | 10.8% | 9.2% | 11.7% | 9.6% | 13.4% |
| Uniform | 8.3% | 10.4% | 8.8% | 11.2% | 9.0% | 12.1% |
| MiBench | 7.5% | 9.6% | 8.2% | 9.3% | 8.7% | 9.9% |

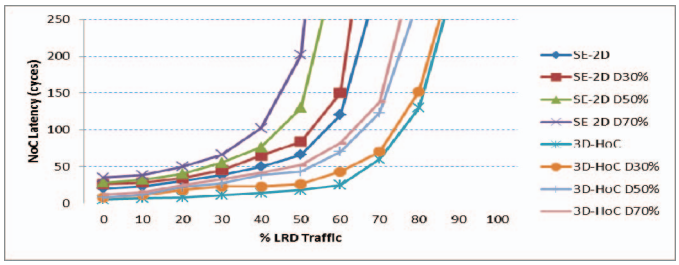


Figure 4. Latency results for CS L3 AC and AU security level and diferrent percentages of dynamicity.

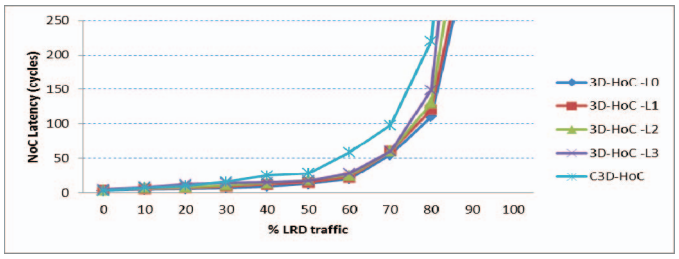


Figure 5. 3D-HoC latency results for different levels of protection.

They show that our secure 3D-HoC can perform a fast detection of a wide range of attacks and a fast configuration of different security policies when compared to the C3D-HoC. Table IV summarizes the penalties of 3D-HoC for different traffic patterns.

## VIII. CONCLUSIONS

In this work we propose a dynamic security enhanced 3D-HoC for 3D MPSoC protection. We show that 3D-HoC can be an efficient structure to guarantee the protection in the system. 3D technology not only presents new challenges, but new opportunities to achieve a secure and efficient system. Three techniques are employed in order to achieve an efficient configuration: 1- Only some firewalls are upgraded, so the communication in the remaining of the system is not interrupted; 2- Security customization; and 3) Intrinsic low latency of the 3D technology. We compare our distributed architecture with a centralized one. As dynamicity increases, the distributed alternative becomes more efficient. As future work we plan to implement integrity and confidentiality security services.

## REFERENCES


[1] Healy, M.B. et al, "Design and analysis of 3D-MAPS: A many-core 3D processor with stacked memory". In: Proc. (CICC), 2010.

[2] Lukovic S. et al, “Enhancing NoC components to support security”In Proc WESS 2010

[3] Kocher P., Lee R., McGraw G., Raghunathan A., Ravi S.: Security as a New Dimension in Embedded System Design. (DAC2004). .

[4] Lukovic S., Christianos N., Enhancing Network-on-Chip components to support Security of Processing Elements. (WESS 2010).

[5] Coburn J., Ravi S., Raghunathan A., Chakradhar S.: SECA: Security-Enhanced Communication Architecture. ( CASES 2005).

[6] Fiorin L., Silvano C., Sami M.: Security Aspects in Networks-on-Chips: Overview and Proposals for Secure Implementations. (EUROMICRO 2007).

[7] Evain S., Diguet J.: From NoC security analysis to design solutions. (DATE 2006).

[8] Fiorin L., Lukovic S., Palermo G.: Implementation of a Reconfigurable Data Protection Module for NoC-based MPSoCs. (PDP 2008).

[9] Sepulveda M.J, Gogniat G., Pires R., Cahu W., Strum M., “Dynamic NoC-based MPSoC Security Implemetation”. (SBCCI 2011).

[10] Sheibanyrad, A. et al, “3D integration for NoC-based SoC Architectures”. Integrated Circuits and Systems. Springer. 2011.