Using pattern of on-off routers and links and router delays to protect network-on-chip intellectual property

https://doi.org/10.1145/3548680

Published in:
ACM Transactions on Computer Systems

Document Version:
Peer reviewed version

Queen's University Belfast - Research Portal:
Link to publication record in Queen's University Belfast Research Portal

Publisher rights
Copyright 2022, Association for Computing Machinery.
This work is made available online in accordance with the publisher’s policies. Please refer to any applicable terms of use of the publisher.

General rights
Copyright for the publications made accessible via the Queen's University Belfast Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The Research Portal is Queen's institutional repository that provides access to Queen's research output. Every effort has been made to ensure that content in the Research Portal does not infringe any person’s rights, or applicable UK laws. If you discover content in the Research Portal that you believe breaches copyright or violates any law, please contact openaccess@qub.ac.uk.

Open Access
This research has been made openly available by Queen's academics and its Open Research team. We would love to hear how access to this research benefits you. – Share your feedback with us: http://go.qub.ac.uk/oa-feedback
Using Pattern of On-off Routers and Links and Router Delays to Protect Network-on-Chip Intellectual Property

ARNAB KUMAR BISWAS, Centre for Secure Information Technologies, School of Electronics, Electrical Engineering and Computer Science, Queen’s University Belfast

Intellectual Property (IP) reuse is a well known practice in chip design processes. Nowadays, network-on-chips (NoCs) are increasingly used as IP and sold by various vendors to be integrated in a multiprocessor system-on-chip (MPSoC). However, IP reuse exposes the design to IP theft, and an attacker can launch IP stealing attacks against NoC IPs. With the growing adoption of MPSoC, such attacks can result in huge financial losses. In this paper, we propose four NoC IP protection techniques using fingerprint embedding: ON-OFF router-based fingerprinting (ORF), ON-OFF link-based fingerprinting (OLF), Router delay-based fingerprinting (RTDF), and Row delay-based fingerprinting (RWDF). ORF and OLF techniques use patterns of ON-OFF routers and links, respectively, while RTDF and RWDF techniques use router delays to embed fingerprints. We show that all of our proposed techniques require much less hardware overhead compared to an existing NoC IP security solution (square spiral routing) and also provide better security from removal and masking attacks. In particular, our proposed techniques require between 40.75% to 48.43% less router area compared to the existing solution. We also show that our solutions do not affect the normal packet latency and hence do not degrade the NoC performance.

CCS Concepts: • Networks → Network on chip; • Security and privacy → Malicious design modifications; • Applied computing → Computer-aided design.

Additional Key Words and Phrases: Intellectual property protection, NoC IP protection, Fingerprinting technique.

ACM Reference Format:

1 INTRODUCTION

In the last decade, multiprocessor system-on-chips (MPSoCs) have gained widespread use in various sectors from network processors to servers in high performance computing systems. Network-on-chips (NoCs) provide a scalable on-chip communication medium in these MPSoCs. NoCs provide notable improvements over point-to-point communication and bus-based communication in MPSoCs. Nowadays, NoCs are intellectual property (IP) and are part of the MPSoC design process. To enable widespread adoption of the technology without the risk of IP theft, there is a need for NoC IP security. A comprehensive NoC IP security framework can also provide security to an MPSoC chip that contains the IP.

Digital watermarking is a technique where the owner of an IP marks the IP to uniquely identify the owner. Digital fingerprinting is a method through which an owner or seller of an IP marks the IP differently for different buyers uniquely identifying the buyer. Watermarks can only identify the owner and protect the owner from IP theft, but fingerprints can identify both the buyer and the seller, i.e. they can protect the buyer from dishonest sellers and also sellers from IP theft. There are many hardware IP watermarking methods available in literature. Among them, only the study in [11] has specifically considered NoC IP security using a special routing algorithm called SSP. However, this work has several disadvantages which will be discussed in Section 2.4. In this paper,
we compare our proposed approach to the SSP method since it is specifically targeted to NoC IP at the router architecture level.

In this paper, we propose four novel NoC IP fingerprinting methods that rely on different fingerprint embedding mechanisms. The four methods are: ON-OFF router-based fingerprinting (ORF), ON-OFF link-based fingerprinting (OLF), Router delay-based fingerprinting (RTDF), and Row delay-based fingerprinting (RWDF). The pattern of OFF-routers and links in an NoC for ORF and OLF, respectively, is fixed after fingerprint embedding and can be detected easily. Both RTDF and RWDF techniques use router delay to embed fingerprints but the detection techniques are different. For RTDF, the fingerprint is detected from each router, and for RWDF it is detected from the entire row of NoC routers. Details of each technique are discussed in Section 4.

There exist IP security solutions in literature at the lower level of abstraction like at the gate level \[6\] and at the layout level \[7, 8\]. These lower level solutions are mutually exclusive to our architecture level solutions and can be used together if required. But there are some problems with these solutions which will be discussed in Section 2. Also, our solutions do not require the chip to be reverse engineered to extract the fingerprint. The fingerprint extraction happens as soon as the authentication packet exits the NoC. Further discussion on authentication packets and about our solutions are given in Section 4.

The main contributions of this work are as follows.

1. We propose four new NoC IP fingerprinting techniques: ORF, OLF, RTDF, and RWDF.
2. We demonstrate that our proposed techniques provide better security compared to existing work.
3. We also show that all the four methods require less router area compared to the existing SSP work.

The rest of the paper is organized as follows. We provide a description of related works in Section 2. Section 3 gives a preliminary description of the VLSI supply chain and also the threat model considered in this paper. Next, we describe our proposed techniques in Section 4. The security analysis, simulation, and synthesis results are shown in Section 5. The paper is concluded in Section 6.

2 RELATED WORKS

Here we discuss the different types of hardware IP watermarking techniques including the existing NoC IP protection method.

2.1 Constraint-based watermarking

Constraint-based watermarking was proposed in \[7, 8\]. A design is said to have a watermark embedded in it if the design satisfies a particular set of constraints. When we say that a design satisfies a particular set of constraints, it means that the design satisfies a set of conditions that also includes the conditions assigned specifically to identify the owner (watermark). There are couple of disadvantages to the constraint-based watermarking method. Firstly, the strength of the watermark depends on the fact that an adversary cannot obtain a watermarked design solution by chance. This is a problem in practice because an adversary can actually obtain a hardware design solution after regular optimization steps, which contains the watermark by chance. This is obtained by chance because the attacker does not have the owner’s watermark but the obtained design satisfies a set of conditions without attacker’s knowledge which actually indicate the owner’s watermark. Secondly, there is no guarantee that there exists any design solution (even after adding new constraints) that is not inferior to the un-watermarked design solution. This means that the procedure
of watermark embedding may move the final design solution point from the very small optimization space resulting in a trade-off between security and performance. Hence, the designer needs to do a thorough search in the optimization design solution space that requires a lot of processing. Authors in [7, 8] have proposed a post-processing flow in the placement procedure in the hardware design flow. They have encoded a signature by placing particular standard cells within a cell row with a specified parity (i.e., odd- or even-parity). They have also proposed a pre-processing flow in the routing procedure where a signature is encoded as an upper bound on the wrong-way wiring used for particular signal network routing.

2.2 Embedded watermark generation circuit
The author in [6] has proposed to embed a watermark generating circuit (WGC) and a test circuit in an IP core at the behavioral design level. Various system-on-chip design-for-test (DFT) strategies are used in the work. The main disadvantage of this technique is that the author has not considered the removal attack scenario because the WGC is a separate entity and can be disabled or removed by an adversary who owns the IP without any deterioration in the normal IP operation.

2.3 Embedding a test machine by recoding state variables
Authors in [4] have proposed a blind fingerprinting technique by embedding a test machine into a sequential circuit by recoding the state variables. The technique is called blind because it uses Chaum’s blind signature protocol based on RSA. One disadvantage of [4] is that each fingerprinting design solution embeds a new test machine into the sequential circuit. This is similar to developing a new finite state machine because of a change of state variables each time. Another disadvantage comes from the blind signature scheme using RSA. If an adversary can extract the embedded information from the IP, she can get the watermark of the IP designer using the public key. It is possible because encrypted data using the private key can be decrypted by the public key. The adversary can use the leaked watermark maliciously by using it in other IPs or by claiming ownership. Detailed descriptions of different hardware IP watermarking techniques can be found in [12–14].

2.4 Square spiral routing
Authors in [11] have proposed a square spiral routing algorithm in NoCs to embed watermarks. In this proposal, a square spiral path is formed using NoC routers and each router in the selected square spiral path (SSP) stores a part of the watermark. A trusted third party needs to check all possible SSPs in the NoC and gather information to check against any possible match with data from the designer. To reduce the detection time, the trusted third party considers partial matching [11], but note that this could lead to false authentication. Two different designers can have a certain level of similarity in their watermarks, and hence partially obtained information leads to uncertainty in the authentication process. Partial matching also allows an adversary to claim an IP ownership by random guessing because some bits may match with the original owner’s watermark, so the trusted third party needs to check all possible paths and compare against full watermarks from an owner to prove the claim of ownership. This requires a lot of searching and processing at the detection phase. The number of possible SSPs is directly dependent on the number of routers in the NoC. A trusted third party needs to send packets assuming each router as a source and each other router as a destination. This requires \( O(n(n - 1)) \) time for an NoC with \( n \) routers. The solution in [11] may result in unreasonable authentication time with the increase in the number of processors (and also routers) inside an MPSoC in the future [15].
2.5 Checking the power consumption waveform of hardware IP

In [5], a hardware watermarking method is given. The authors have used a watermark memory to store a watermark characterizing the IP and a watermark signal generator to generate watermark signal detectable on a power supply line of the electronic circuit. The main disadvantage of this technique is that the watermark cannot be successfully detected in the presence of noise at the power supply line. In [2], a similar method to authenticate an electronic circuit is given. Similarly to [5], this method also incorporates an authentication identification generating circuit in the IP. According to the method, the power consumption waveform of the IP under a predetermined condition is initially stored. Later the power waveform of a suspected IP under same condition is checked against the previously stored waveform. Matching of waveforms authenticates the IP. The main disadvantage of this technique is that the storage required to store all waveforms is huge. Another disadvantage is that the predetermined condition needs to be exactly the same and any noise at the power supply line has to be eliminated for exact waveform matching.

2.6 Physical Unclonable Function based IP protection

A hardware IP can also be identified by Physical Unclonable Function (PUF) [9]. It is a chip-dependent unclonable challenge-response function that can uniquely identify a specific integrated circuit. Physical features of the device are utilized to realize the mapping from an input signal to an unpredictable output. But PUF based solutions are more applicable for single IP design and cannot be used for a scalable network IP like NoC. For the latter, we need to use the inherent behavior of the NoC IP itself to fingerprint it.

Also an attacker can easily forge the characteristic of challenge-response pair by establishing mathematical model [9]. This is possible because existing PUF methods only provide few challenge-response pairs for a particular device due to limitation of physical characteristic of the device used for response. It cannot produce wide range of responses. Existing solutions also have high power overhead [9]. Another limitation of PUF based solutions is that they are only applicable to protect the owner and cannot be used by legal IP buyer. A legal IP buyer cannot authenticate and detect the ownership using PUF based solutions. This is not true for any fingerprinting method including our proposed methods. Fingerprint generation involves the buyer’s specific information and can also be easily used to determine the ownership by the buyer with the help of a trusted third party.

3 PRELIMINARIES AND THREAT MODEL

Here we discuss the different types of hardware IPs and the threat model considered in this paper.

3.1 Preliminaries

Mainly, two types of IPs are used in IC designs: soft IP and hard IP. Synthesizable RTL code and gate level netlists are used as soft IP which allows synthesis, placement, and routing design flows. IPs which are offered in layout format or GDSII format are known as hard IP. These IPs must obey the target foundry’s process design rules. Figure 1 shows different stages of IC design and fabrication stages and the use of different types of IPs. This discussion about IPs is applicable to NoC IP. Note that the whole chip design and fabrication flow is much more complicated and Figure 1 does not show the complete flow in full detail. Our main aim is to illustrate the main entities and how they share and use different IPs and highlight where our solution is applicable.

3.2 Threat model

In this paper, we consider an IP stealing attack against hard IPs or fabricated ICs. We target the outsider adversary who performs IP theft, counterfeiting, and cloning, and the foundry that performs
over-building and cloning [10]. Over-building and cloning means that the product is produced in excess number without owner’s permission and probably with different labels to hide any claim from owner. With our proposed solutions, the owner can easily prove the ownership of the IP from these overproduced or cloned products. Counterfeiting means that a similar looking but a fake product is made with the owner’s label which does not cost much to produce. This type of bad product causes loss of reputation for the owner in markets and hence loss of revenue due to a decrease of product sales. Using the proposed solutions, an owner can easily prove that the product is fake, that it was not created using the owner’s IP, and can claim damage from the adversary. Usually, rogue business entities are driven by profit only. They are mainly interested in IP theft or misuse over tampering or changing the IP design [10]. This means that our solutions will be useful against these types of adversaries.

4 PROPOSED FINGERPRINTING TECHNIQUES

In this section, we describe the fingerprint generation process and all four proposed methods to embed the fingerprint in the NoC.

4.1 Fingerprint generation

Figure 2 shows the fingerprint generation process which is same for all the four proposed techniques. A designer or owner has a set of watermarks and the aim of the fingerprint generation
process is to generate a fingerprint specific to a buyer using a randomly chosen watermark from this set. Here, we assume that the watermark is 128 bit long. When a buyer requests to buy an IP, the buyer’s public key is passed to the AES that generates 128 bit output. The output is obtained by performing encryption of the buyer’s public key considering it as message. Please note that AES is chosen because it provides enough security against an attacker who knows the message but does not know the secret key used for encryption [1]. So the attacker does not know the encrypted (modified) public key that is obtained after encryption. The Selection Function randomly selects one watermark from the set of watermarks. Then, a fingerprint is obtained by encrypting (using AES) this watermark using the modified public key. Next, the Extraction Module gives the final fingerprint after taking the target NoC size (number of rows and columns) and the generated fingerprint as inputs. Currently, this module selects the bits of the fingerprint from least significant bit (LSB) side according to the NoC size. The main aim of the process is to stop an attacker from obtaining the owner’s watermark if she knows the fingerprint from the NoC IP and buyer’s public key. Please note that the buyer’s public key is not hidden and that is why we assume that the attacker is also able to know this public key but not the modified public key after AES encryption.

In all of our proposed techniques, the fingerprint is embedded at the NoC design stage so that it can be extracted later from the IP and prove the ownership. After design and integration into an MPSoC, the IP has two working phases: normal working phase and authentication phase. During normal working phase, normal packets are used for communication through the NoC between different processing elements of the MPSoC according to the running applications. Figure 3 shows normal packet structure with a 1 bit type field. This bit is 0 for normal packets. If there is any doubt about ownership, authentication phase is started. During this phase, authentication packet(s) with 1 in the type field is (are) used for extraction of embedded fingerprint from the NoC IP. The authentication packet structure is different for different proposed techniques and will be discussed later.

4.2 ON-OFF router-based fingerprinting (ORF) and ON-OFF link-based fingerprinting (OLF) techniques

In the case of ORF and OLF techniques, the fingerprint is embedded by switching OFF routers and links, respectively in a specific pattern during authentication phase. One example of ORF and OLF technique is shown in Figure 4 and 5, respectively. In the case of ORF, the fingerprint is divided into groups based on the number of routers in a row apart from the first router of that row. According to the example shown in Figure 4, the group size is 3. Then, a bit “1” is added before each group. This is done to ensure that the first router is always ON to let the authentication packet enter the row. So, in a $4 \times 4$ NoC, a 12 bit fingerprint can be embedded. Next, checking is done to detect any isolated router due to the modified fingerprint. It may happen that an ON router cannot be reached because it is surrounded by OFF routers. In such case, a different watermark is used to get a new fingerprint and restart the whole procedure as shown in Figure 6. Otherwise, the fingerprint is embedded by setting the condition on the power supply of selected routers. During the authentication phase, these routers will be OFF but during normal operation they will work normally.

In case of OLF, the fingerprint is not modified. It is checked to see if a whole column of links are getting OFF after embedding. Currently, only the East output ports are considered. If all the links in a column gets OFF, the authentication packets cannot reach all routers in the NoC, so a
new watermark is selected to get a new fingerprint. Otherwise, the fingerprint is embedded by
setting the condition on the East output port of selected routers as shown in Figure 7. During
the authentication phase, these ports will be disabled, but during normal operation they will work
normally.

Please note that the proposed OLF method can be implemented targeting not only the East
output ports but in all other output ports too i.e., West, North and South. The selection of port only
depends on the direction of the authentication packet flow. In current case as shown in Figure 5, the
packets are flowing toward the right side of the NoC which means that they are flowing through
the East output ports of routers. That is why the East output ports are selected in current case.
If we consider the case when the authentication packets are flowing to downwards, we need to
select the South output ports.
During authentication phase, an authentication packet is sent through every row. In case of ORF, all the ON routers in a row will set a corresponding bit in the passing packet that is supposed to pass through that row as shown in Figure 8(a). It means that routers (0,2), (1,2), (2,2), and (3,2) in Figure 4 do not change the \textit{packet1} because it is not supposed to pass through that row. Please note that \textit{packet1} traverses through the path marked by 1 in Figure 4. In case of OLF, all the ON links in a row (the East output port in current case) will set a corresponding bit in the passing packet that is supposed to pass through that row as shown in Figure 8(b). An authentication packet has a field (with initial value 0) where each bit corresponds to a router or a link and can be set only by that router or link. Next, all the packets are collected and the embedded fingerprint is extracted. In case of ORF, the extra “1” bits before each group of bits are discarded. The collected fingerprint is then checked for validity. A different packet traversing in different rows reduces the total time to
detect the fingerprint. In Figures 4 and 5, 4 packets are sent through 4 rows. The packet traversal paths are shown with the packet numbers in each figure. The extracted fingerprint from Figure 4 and 5 are 010111011110 and 0101101101111101, respectively.

4.2.1 Router architecture and authentication packet routing. Figure 9 shows the router architecture used for ORF and OLF techniques. The router has five input and five output ports: North (N), South (S), East (E), West (W) and Eject (I). Only three input and two output ports are shown in the figure. The type checker module checks the input packet’s type field when it enters the input FIFO of an input port and transfers it to decision logic (for normal packets) or to authentication packet routing module (APRM) (for authentication packets). The decision logic routes normal packets using routing table (RT) based deterministic Y-X routing algorithm. The fingerprint embedder embeds the fingerprint bit in the authentication packet. The packet structure used for ORF and OLF techniques is shown in Figure 10. Currently, the fingerprint field in the packet structure is of 4 bit width because there are 4 routers in a row through which the authentication packet passes.

We first describe the authentication packet routing for the OLF case. As shown in Figure 5, packets are transmitted into the West input ports of the first column routers towards the destinations connected to the East out ports of the last column. If a link (the East output port in current case) is OFF, packet is diverted to the North or the South, but never to the West. This ensures that the packet will eventually reach the destination. The logic of this is that the exit route is searched first apart from the router from where the packet is coming. If an exit route is not present, then the packet will return back to the sending router. If the packet returns to the sending router, no packets are sent to the all link-OFF router after that. If authentication packet enters the North input port from another router’s South output port, Algorithm 1 is used to route the packet. Similarly if
Fig. 11. A $4 \times 4$ NoC with authentication packets traversing in every row.

the packet enters through the South input port, first the East, then the North, and at last the South output port is checked for readiness. Router checks the arrived packet’s destination which also helps to ensure that the packet gets the embedded fingerprint from the routers of the row through which it is supposed to pass. It eliminates the possibility of the packet getting information from the routers of other rows, where the packet is presented due to not-ready links of its corresponding row routers. If a router’s X-dimensional coordinate is $x$, the $(9+x)$th bit in a packet is set when the packet passes through the East output port and reaches the West input port of correct row.

Algorithm 1 Authentication packet routing algorithm at the North input port

1: if the East output port is ready then
2: send the packet to the East;
3: else if the East output port is not ready and the South output port is ready then
4: send the packet to the South;
5: else if the East output port is not ready and the South output port is not ready and the North output port is ready then
6: send the packet to the North;
7: end if

For the ORF case, apart from the West input port, the North and the South input ports also embed fingerprint if the packet is passing through the correct row. The writing position is same for all input ports of a router i.e., $9 + x$. So there is no problem of modifying multiple locations by mistake if the packet returns back from other router due to switched OFF router. There will be only overwriting to the same location. Authentication packet is routed following the same logic as OLF case.

4.3 Router delay-based fingerprinting (RTDF), and Row delay-based fingerprinting (RWDF) techniques

In the cases of RTDF and RWDF techniques, a fingerprint is generated by the same process as mentioned in Subsection 4.1. In these techniques, router delay is modulated to embed the fingerprint. Router delay is the time required to pass through the router by a packet after entering it at
Using Pattern of On-off Routers and Links and Router Delays to Protect Network-on-Chip Intellectual Property

Fig. 12. Flowchart of fingerprint decoding method from NoC routers.

Fig. 13. Flowchart of fingerprint decoding method from NoC rows.

the input port. During the authentication phase, an authentication packet is sent in every row as shown in Figure 11 and it is delayed inside routers according to the embedded fingerprint. Normal packets will not be delayed by the routers.

The fingerprint is divided according to the number of rows and each part is again divided among the routers in a row. The Delay control module inside each router is responsible for delaying the authentication packets. The router architecture will be described later.

There are two possible ways to extract the fingerprint from router delay, and that is how RTDF and RWDF techniques differ. In case of RTDF, an authentication packet is delayed in a router and the Delay Control module increments a field in the packet in each clock cycle. Every router’s Delay Control module increments its corresponding field in the packet and all the embedded information in a row are obtained after receiving the packet. Extracted information from all rows are appended to get the fingerprint. The flowchart of this method is given in Figure 12. In case of RWDF, an authentication packet is delayed in every router according to the embedded fingerprint but the Delay Control module does not modify the packet. After receiving the packet at the end of the row, the total delay (row latency) is measured. All the row delays are appended to get the embedded fingerprint. Flowchart of this method is given in Figure 13. The collected fingerprint (from RTDF or RWDF) is then checked for validity. A different packet traversing in different rows reduces the total time to detect the fingerprint. In the current case, 4 packets are sent through 4 rows as shown in Figure 11. The packet traversal paths are shown with the packet numbers in the figure.

4.3.1 Router architecture and authentication packet routing. Figure 14 shows the router architecture used for RTDF and RWDF techniques. The router architecture is similar to ORF and OLF techniques except the delay control module is presented in current cases. The packet structure used for RTDF technique is also shown in Figure 15. The APRM uses the Decision logic (connection not shown in the figure) to route authentication packets using routing table (RT) based deterministic Y-X routing algorithm.

In case of RTDF, only the West input port receives authentication packet which is then forwarded to the East output port, so the delay control module is presented at the West input port only. The delay control module contains a delay counter and a timer. Also, it calculates the delay caused by previous routers and updates the packet in the APRM. Each router gets the delay information beforehand. After receiving the packet, the delay control module starts incrementing the delay counter by one each clock cycle until it equals the set delay value. Then, it enables the
Fig. 14. Router architecture used for RTDF and RWDF techniques. Only two input and two output ports are shown among the five input and five output ports.

Fig. 15. Packet structure used for RTDF technique.

APRM to pass the packet to East output port. The delay counter is then reset to zero. The timer starts from the start of operation and increments by one in each clock cycle and helps to embed the delay information in the packet. Each packet has specific fields for each router. The timer value is stored in the correct location by storing the value at \([13 + 5 \times i : 9 + 5 \times i]\) (refer to the Fingerprint field of the packet structure as shown in Figure 15) where \(i\) indicates routers (0 to 3). Except for the first router, all routers calculate delay caused by the previous router and stores the value to that router’s corresponding location in the packet. It is obtained by deducting the timer value stored in that location from the current timer value. This calculation does not cost any extra clock cycle delay and the packet is passed as soon as the set delay value is reached by the delay counter.

The last router delay is calculated by the receiver (called the input-output controller) at the destination connected to the East output port. That means the receiver also has a similar timer which is started at the start of operation and can be used to calculate the delay of the last router. The input-output controller is used to send packets into the rows of NoC fabric and to receive the packets from it. This module is also inside the chip but outside the fabric. This is responsible for decoding the fingerprint and appending all information received from different packets to get the final fingerprint.

We can use multiple packets to extract the embedded fingerprint from routers of a row if the number of routers are more than the length of a single packet to handle. In this case, each packet will be distinguished by a packet id and routers will know beforehand to update a particular packet id only and not all authentication packets.

In case of RWDF, the delay control module is still there but it doesn’t contain the timer because it doesn’t calculate any delay. It has only the delay counter which delays the packet according to the set value and then enables the APRM to pass the packet to the East output port. The row delay is actually calculated by the I/O controller using its timer. In fact, if the timer is started at the sending time, the current value of timer is the row delay when the packet is received. The packet structure doesn’t matter because it doesn’t carry any information. Only the presence of the packet is needed to calculate the delay. Currently, the RTDF packet structure is used for RWDF also.

Please note that if the NoC IP needs to be authenticated, the authentication phase starts when the currently running operation is finished. So, in current situation, normal packets do not exist...
in router buffers when the authentication packets start to flow. This ensures that in case of RTDF and RWDF, there is no congestion and embedded fingerprint can be extracted correctly.

### 4.4 Total embedded fingerprint capacity comparison

In all the proposed techniques, the watermark size is fixed and currently it is 128 bits, but the embedded fingerprint capacity is different between different techniques. Embedded fingerprint capacity denotes the total number of bits embedded in the NoC IP using a particular technique. Here, we provide the models that give the fingerprint capacity for each case.

In case of ORF, \((N - 1)\) bits/packet transmission is possible for \(N\) routers in a row, so the total capacity is \(N(N - 1)\) bits for \(N\) packet transmission in \(N\) rows. In case of OLF, \(N\) bits/packet transmission is possible for \(N\) routers in a row, so the total capacity is \((N * N)\) bits for \(N\) packet transmission in \(N\) rows. Thus, for a \(4 \times 4\) NoC, ORF and OLF capacities are 12 bits and 16 bits, respectively.

In case of RTDF and RWDF, let us assume that \(R_{di}\) is the router delay of the \(i^{th}\) router. In the case of RTDF, total embedding capacity is \(N\{\log_2 R_{d0} + \log_2 R_{d1} + \cdots + \log_2 R_{d(N-1)}\}\) = \(N \sum_N \log_2 R_{di}\) bits for \(N\) rows each with \(N\) routers. In the case of RWDF, the total embedding capacity is \(N \log_2 (\sum_N R_{di})\) bits for \(N\) rows each with \(N\) routers. Currently, both for RTDF and RWDF techniques, the router delay is controlled by 2 bits and can cause a maximum of a 3 clock cycle delay in each router. Thus, for a \(4 \times 4\) NoC, RTDF and RWDF capacities are 32 bits and 16 bits, respectively.

### 5 SECURITY ANALYSIS, SIMULATION AND SYNTHESIS RESULTS

Here we discuss the security analysis of our proposed techniques and provide simulation and synthesis results.

#### 5.1 Security analysis

It should be noted that non-watermarked NoCs should not be detected as a watermarked one. The probability of occurrence (false detection) should be very low. This property is known as the uniqueness property. In case of ORF and OLF, the situation of getting different routers or links OFF in a non-fingerprinted NoC is possible only in a faulty NoC but in this work, we assume a fault-free NoC and hence the probability of false detection is very low for ORF and OLF cases. In case of RTDF and RWDF, the routers can only delay the packet transmission in an uncertain manner (which may resemble the modulated delay of RTDF or RWDF) if there is any congestion in the network. In our case, the probability of congestion in network is 0 because of the absence of any packet other than the authentication packet during authentication phase. So the probability of false detection is very low for RTDF and RWDF cases also.

Please note that in all of our proposed techniques, the fingerprint can be extracted easily from the NoC but the watermark which denotes the owner’s identity is hidden. The fingerprint generation process shown in Figure 2 provides cryptographic security against any attempt to obtain the watermark from fingerprint. This ensures that the attacker cannot obtain the watermark of the owner and cannot misuse this information. This is in contrast to the existing work like [11] where the owner’s watermark is not hidden or kept secret from the attacker. Even though they have used AES to encrypt but the path information is exactly same where the authentication packet will flow and successful extraction of embedded encrypted information is enough for an attacker to know the watermark of the owner. The attacker does not need to attack AES to know the path information (i.e., the routers through which the packet is routed) which is the watermark itself. It seems that the encryption is simply done to increase the number of fingerprints in [11] and they do not try to hide the watermark at all. This means that the use of AES does not provide any
cryptographic security at all in their case, which is different from our case. In our case, the path information is revealed to help fast authentication and prove the ownership by the owner but the watermark cannot be extracted by anybody else without knowing the secret keys. That means our solution provides cryptographic security but the work in [11] provides only security which is due to the effort required to find the path. This is much weaker than the security obtained due to AES in our case.

A masking and removal attack can be used to change the IP design to remove or hide the watermark or fingerprint without changing the functionality of the IP. This attack can be prevented by hiding the details of the IP (specifically the router architecture) using hardware obfuscation techniques. We assume that layout level obfuscation techniques such as mentioned in [3] are used to obfuscate the NoC IP. Note that we consider only hard IP or fabricated chips in this paper as mentioned in the threat model in Subsection 3.2. Please note that obfuscation method itself can also protect from reverse engineering and IP piracy but this is very costly. If an owner wants to prove his/her ownership, she has to remove each layer of the chip to extract the layout. In this paper, we are proposing much simpler solutions without requiring this costly procedure to prove the ownership. Our solutions only require access to the chip to check the ownership. Thus, our solution in addition to an obfuscation method can provide full protection, which is much simpler to prove ownership than gate level extraction and is also robust at the gate level because of the obfuscation method. Since the main target of the obfuscation in our case is to hide the details of the hardware implementation of each router, we think that layout level obfuscation methods and camouflaging will be very effective. Please note that the existing SSP method in [11] also uses obfuscation method as mentioned in that paper.

There is another aspect of the masking and removal attack. The embedded information bits the router supplies to the authentication packet may be changed or disabled by the attacker. This attack can happen only if the attacker is able to find the storage location inside the router that employs obfuscation techniques to hide the details. The probability of storage detection (\(P_{sd}\)) directly depends on the number of bits that are stored (irrespective of actual storage mechanism). Bigger storage means easier to detect it at the VLSI level.

The probability of removal attack and masking attack can be represented by

\[ P_{sd} = \frac{A_b}{A_T} \]  

Here \(A_b\) and \(A_T\) denote embedded information storage area and total router area, respectively. Figure 16 shows \(P_{sd}\) (from Equation 1) for different NoC sizes for different proposed mechanisms and also for an existing work (SSP) [11] for comparison purpose. Router synthesis for different techniques is done for different NoC sizes using Synopsis Design compiler tool using TSMC 65 nm technology libraries at 770 MHz clock frequency. It can be observed from the figure that the probability of storage detection is higher for existing SSP work because of large number of bits stored in each router. We consider 8 bits per router for all NoC sizes. The probabilities are low for ORF and OLF cases because of only 1 bit storage requirement and the probabilities for RWDF and RTDF cases are higher than ORF and OLF cases but they are still much lower than the existing SSP work because of much lower storage requirement (2 bit per router).

It is mentioned in [11] that the routers with SSP are different from other routers as per hardware implementation, which is not true in our proposed techniques. In that case, the security attributed by uncertain nature of SSP path gets nullified because an attacker can easily distinguish SSP routers from the non-SSP routers that are different from hardware implementation point of view. That means our proposed techniques provide better security from removal and masking attacks as shown in Figure 16.
Using Pattern of On-off Routers and Links and Router Delays to Protect Network-on-Chip Intellectual Property

Fig. 16. Probability of storage detection for different NoC sizes. All NoC sizes considered are square in shape like $4 \times 4$, $8 \times 8$ etc. but only one dimension is mentioned in the plot axis.

Fig. 17. Average latency values (in clock cycles) for different techniques during normal working phase for different NoC sizes. All NoC sizes considered are square in shape like $4 \times 4$, $8 \times 8$ etc. but only one dimension is mentioned in the plot axis.

5.2 Simulation and synthesis results
We implement the routers for proposed techniques and also for the router in [11] using Verilog and simulate the $4 \times 4$ mesh NoC of Figure 11 using vsim to check if the latency of normal packets gets modified due to addition of security measures in routers. We only change the routers for every simulation cases and use the packet structure shown in Figure 3. All our proposed routers are single cycle routers and the router in [11] is considered to be 2 cycle router where main router is single cycle with one additional output stage due to security additions as mentioned in [11]. From the
Table 1. Area and power results of different routers.

<table>
<thead>
<tr>
<th></th>
<th>Area (in $\mu m^2$)</th>
<th>Power (in mW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORF</td>
<td>9334.44</td>
<td>5.67</td>
</tr>
<tr>
<td>OLF</td>
<td>8981.64</td>
<td>5.66</td>
</tr>
<tr>
<td>RTDF</td>
<td>8512.20</td>
<td>5.32</td>
</tr>
<tr>
<td>RWDF</td>
<td>8124.48</td>
<td>5.16</td>
</tr>
<tr>
<td>Non secure</td>
<td>8105.76</td>
<td>5.13</td>
</tr>
</tbody>
</table>

architecture point of view, there should not be any effect on normal packets during normal working phase of the NoC because of the modifications. All the simulations are done for 15000 clock cycles. The warm-up period is 1000 clock cycles, the measurement period is from 1000 to 10000 clock cycles, and the remaining are the drainage period. We use artificial traffic generators and sinks for traffic generations and measurements. Figure 17 shows the average latency values in clock cycles for different cases for 50% packet injection rate. We also simulate during the authentication phase for different cases and the average latency values are exactly same as shown in Figure 17. This shows that the addition of security measures in routers does not cause any change in normal packet latency and hence does not affect the normal NoC performance. Please note that addition of security measures cause area and power overhead as shown in Table 1 that will be discussed later.

Please note that if we consider the change compared to non-secure router which does not contain the modifications due to security measures, our proposed techniques do not increase the packet latency. But the router architecture in [11] shows an additional output stage due to addition of security compared to non-secure router. That means in case of [11], latency increases 2 times compared to non-secure router both for normal packets and also for authentication packets.

We also test the NoC using RTDF and RWDF techniques on FPGA to check if the proposed methods of modulating packet delay works in a practical implementation on FPGA. In both cases, we target Xilinx Zynq-7000 SoC on Zedboard using Vivado 2014.4 tool and test the proposals at 100 MHz frequency of operation. We build an Input-output (I/O) controller module and test one row of the NoC in both RTDF and RWDF cases. This I/O controller module sends data valid signal for one clock cycle and the authentication packet to the West input port of router (0,3) (refer to the NoC in Figure 11). It also sends a ready signal to the East output port of router (3,3) and receives the packet back. We have found that we are able to extract the fingerprints in both cases without any error from the FPGA implementations. This is possible because we do not add any extra register stages in the critical path of the packet transmission through the routers and the delay in each router in number of clock cycles remains as expected.

Synopsis Design Compiler is used for ASIC synthesis purpose using 65 nm TSMC technology libraries. Area and power results of different routers are given in Table 1 at a frequency of 770 MHz which is the maximum frequency of operation. Please note that the power results indicate total power which is the summation of static and dynamic power. The router areas for ORF, OLF, RTDF, and RWDF are 40.75%, 43%, 46%, and 48.43%, respectively less compared to the router in [11]. The power results also show similar trends. This shows that our proposed techniques require less hardware overhead compared to existing work in [11]. Among different proposed routers, ORF area is larger than OLF because the fingerprint embedder is presented in every input port for ORF but only at West input port for OLF. The area of RWDF is less than RTDF because the timer is
Using Pattern of On-off Routers and Links and Router Delays to Protect Network-on-Chip Intellectual Property

Fig. 18. Router area values for different techniques for different NoC sizes. All NoC sizes considered are square in shape like $4 \times 4$, $8 \times 8$ etc. but only one dimension is mentioned in the plot axis.

Table 2. Fingerprint extraction time comparison for different solutions. $R_D$ denotes maximum router delay according to the implementation.

<table>
<thead>
<tr>
<th>Technique</th>
<th>Extraction time</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORF</td>
<td>$O(n(n-1))$</td>
</tr>
<tr>
<td>OLF</td>
<td>$O(n^2)$</td>
</tr>
<tr>
<td>RTDF</td>
<td>$O(R_D n)$</td>
</tr>
<tr>
<td>RWDF</td>
<td>$O(R_D n)$</td>
</tr>
<tr>
<td>SSP [11]</td>
<td>$O(n^2(n^2 - 1))$</td>
</tr>
</tbody>
</table>

absent in delay control module in RWDF. We have also mentioned the non-secure router area in Table 1 which lacks any IP protection mechanism in it. The router area increments of ORF, OLF, RTDF, RWDF, and the work in [11] compared to the non-secure router are 15.16%, 10.8%, 5.01%, 0.23%, and 94.34%, respectively. Please note that the security implementation in [11] requires an extra router output stage (area increment) which is absent in the non-secure router.

Figure 18 shows the router area for different techniques for different NoC sizes. From this figure, we can clearly understand the router area increment for different techniques if we increase the NoC size from $4 \times 4$ to $12 \times 12$. Note that the Table 1 gives the area values for a $4 \times 4$ NoC.

If we analyze the authentication time for different techniques, it is apparent that there are mainly two components in authentication time. The first part is the extraction time which is different for different solutions. The second part is the verification time which actually proves the ownership. This part takes similar amount of time for different techniques because the main operation (AES decryption) is same in all techniques. So if we want to compare the authentication time for different solutions, we need to compare the fingerprint extraction times from the NoC. The fingerprint extraction time depends on the NoC size and the particular fingerprint that is being extracted. Table 2 shows the worst case extraction time (in big O notation) for different solutions for an $n \times n$ NoC.

5.3 Discussion

In case of [11], when an SSP is chosen, the corresponding SSR algorithm is determined. This means that each individual NoC will need individual routing algorithm to be determined due to change of SSP. After determining it, they need to implement it also in hardware. So effort level increases.
In our case, the routing algorithm does not need to be re-determined each time. This is much faster even though we also need to make changes (set the embedded information in routers). But still the effort level is much less due to their requirement of re-determination and implementation of routing algorithm each time.

The problem of searching all possible SSPs in an NoC to find a match in the database is obviously a time consuming process which hinders the practicality of SSP solution. After finding a match, the obtained path information which is supposed to be the encrypted path information, is decrypted using the key in that database entry. If the decrypted path information is same as that database entry path information, then it proves the ownership. Compared to this process, our proposed techniques require much simpler authentication process. In our case, the fingerprint is extracted from NoC without great effort and then the owner also determines the fingerprint according to the buyer’s public key. If both fingerprints match that confirms the ownership. This shows that our solutions provide better NoC IP security solution with much less effort compared to existing solution.

5.3.1 Comparison among our proposed methods. Among our proposed techniques, RTDF and RWDF require less hardware overhead compared to ORF and OLF and also can embed large number of fingerprint bits. But the router delay during authentication phase of RTDF and RWDF are much higher than ORF and OLF techniques requiring longer authentication phase and the probability of storage detection for RTDF and RWDF are more than ORF and OLF techniques due to larger number of embedded information bits. So a designer needs to consider these trade-offs between different proposed techniques before choosing a particular security solution.

6 CONCLUSION

IP stealing attack against NoC IP is a serious security threat. Innovative NoC IP security solutions are required to protect from this threat. In this paper we have proposed four fingerprinting techniques for NoC IPs: ORF, OLF, RTDF, and RWDF. ORF and OLF techniques use pattern of ON-OFF routers and links, respectively. RTDF and RWDF techniques use router delays to embed fingerprint. We have shown that all of our proposed techniques require much less hardware overhead compared to an existing NoC IP security solution and also provide better security from removal and masking attacks. It is shown that our proposed techniques require between 40.75% to 48.43% less router area compared to the existing solution. We have also shown that our solutions do not affect the normal packet latency and hence do not degrade the NoC performance.

REFERENCES


