On IC Traceability via Blockchain

Md Nazmul Islam, Vinay C Patil, Sandip Kundu
University of Massachusetts Amherst, MA, USA
{mislam, vcpatil, kundu}@ecs.umass.edu

***Abstract*—Traceability of ICs is important for verifying provenance. We present a novel IC traceability scheme based on blockchain. A blockchain is an immutable public record that maintains a continuously-growing list of data records secured from tampering and revision. In the proposed scheme, IC ownership transfer information is recorded and archived in a blockchain. This safe, verifiable method prevents any party from altering or challenging the legitimacy of the information being exchanged. However, we also need to establish correspondence between a record in a public database and the physical device, in an unclonable way. We propose an embedded physically unclonable function (PUF) to establish this correspondence The blockchain ensures the legitimacy of an IC's current owner, while the PUF safeguards against IC counterfeiting and tampering. Our proposed solution combines hardware and software protocols to let supply chain participants authenticate, track, trace, analyze, and provision chips during their entire life cycle.**

***Index Terms*—Blockchain, Traceability, Supply chain, Physically Unclonable Function, Ownership transfer.**

I. Introduction

The ubiquitous integration of silicon devices into applications from varied fields and the globalization of semiconductor manufacturing exacerbate the difficulty of ensuring the authenticity of integrated circuits (ICs) and a secure, reliable supply chain. Malicious actors have more points of opportunity to counterfeit, tamper with or re-package ICs and introduce compromised ICs into a supply chain. According to current estimates [1], around 1% of semiconductor sales are estimated to be those of counterfeited units. Most of the counterfeit electronic components are penetrating the market through recycling [2], [3]. This affects the legitimate suppliers financially [4] and also, poses a security risk when tampered or aged parts are sold as new for use in safety-critical applications, such as automotive systems, avionics and cyber-physical infrastructure [5]. Hence, the emphasis on authenticity, integrity, safety of a device have placed new demands for the development of a *traceable* supply chain.

Until now, using a centralized system with a governing third party has been the only conceivable way to empower a traceability system to ensure data and transaction transparency along supply chains and over a product's lifetime. The governing third party is commissioned with the task of creating a centralized data storage to enable a flow of trusted information [6]. However, relying on a third party (or a small group of cooperating parties) creates an inherent bias, fraud, and single point of weakness in the system. Hence, we need to come up with a more robust approach for developing IC traceability.

In this paper, we propose a novel approach of enabling IC traceability system via the use of blockchain. Blockchain is an *immutable*, *encrypted* ledger that keeps records of digital transactions [7]. Each block contains a list of transactions, timestamp, nonce, and a link to the previous block, forming a chronological chain. This consensus-based, shared, and immutable ledger can enable identification and traceability of an IC throughout the supply chain and its deployment lifetime.

To ensure the fact that at any point along the supply chain, any malicious party cannot tamper with a legitimate IC or introduce a counterfeit IC, we utilize physical unclonable functions (PUFs) as the hardware roots of trust and in particular, Strong PUFs [8]. PUFs are innovative, lightweight circuit primitives which extract secrets from intrinsic physical properties of ICs. They can enable secure, low-cost authentication using Challenge-Response Pairs (CRP) [9]. The primary focus of this paper is specifically on establishing traceability of IC supply chain, although the protocol can be applied to almost any electronic component supply chain.

The major contributions of this paper are:

- Proposing a traceability protocol - that IC suppliers can use to *track* and detect any counterfeits in supply chain.
- Enhancing the traceability protocol - that customers can use to *trace* and verify IC's provenance.

The paper is organized as follows: Section II describes the related works on IC supply chain traceability. In Section III, we present our proposed methodology to establish IC supply chain traceability using blockchain and embedded PUF. Section V outlines a demonstration of the protocol and discusses future directions. Section VI concludes the paper.

II. Related Works

For many years, RFID-based counterfeit detection methods have been integrated into the supply chain to enable IC traceability and detect counterfeits [10], [11]. Schuster *et al.* propsed an RFID-based track and trace solution using EPC (Electronic Product Code) Network infrastructure [10]. Elkhiyaoui *et al.* proposed a new protocol, CHECKER for counterfeit detection in RFID-based supply chains through onsite checking [11]. However, RFID-based technologies cannot provide truly secure solutions for supply chain traceability because an adversary can easily copy the unique identifier of one RFID tag to another tag.

Several active metering approaches have been proposed in the literature to uniquely identify and prevent counterfeit ICs in the supply chain [12]. Logic encryption techniques are also used to hardwire designs with built-in secret encryption keys to prevent counterfeit ICs in the supply chain [13]. However, these hardware metering approaches increase the design cost, marketing cost and the cost of distributing keys. There have been very few works which can also track the entire supply chain. Our work provides a novel approach for detecting counterfeit ICs by tracking entire supply chain.

978-1-5386-4260-3/18/$31.00 ©2018 IEEE

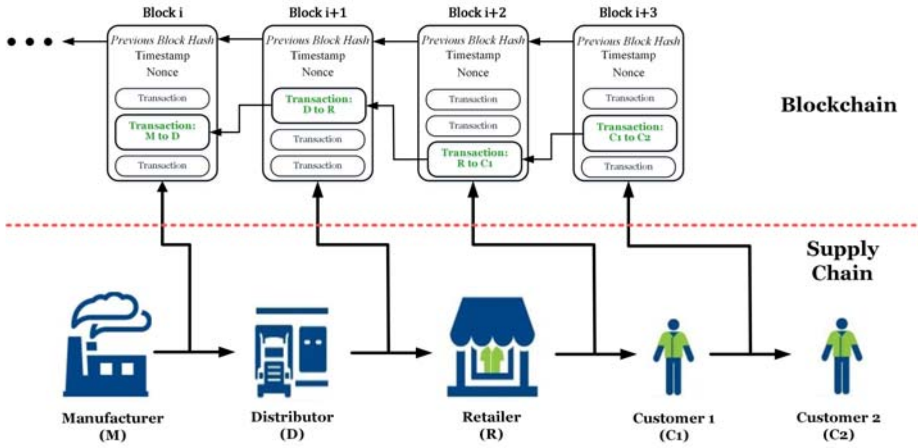

Fig. 1. Proposed approach for IC traceability from manufacturer to the end-user.

III. IC Traceability Protocol

In this section, we present our proposed protocol for IC traceability via blockchain. First, we describe our approach and the key system requirements of the protocol. Next, we present the methodology of creating the blockchain. Finally, we explore the details of ownership transfer which is the key part of our overall blockchain protocol.

A. Approach

The blockchain will contain a record of the relevant IC ownership transfer information, termed as a *transaction*, and PUF data, used for authentication, for each point of transfer over the device's lifetime. This also enables proof of ownership without an explicit need for trusted intermediary. Furthermore, authorized parties can utilize the blockchain to authenticate, track, analyze, and provision chips. Fig. 1 presents our proposed approach using blockchain.

Before detailing the protocol, we enumerate the key requirements for creating the blockchain and explain their necessity for enabling reliable tracing of ICs.

1) Only the legitimate IP owners registered with a designated consortium can claim the initial ownership of an IC and write the relevant PUF data in the blockchain.
2) Only the current owner of the IC is able to create a new transaction for transferring the ownership to a new owner.

The first requirement prevents unauthorized parties, like counterfeiters, from falsely claiming ownership of an IC. This can be ensured by verifying that the *first* transaction in the blockchain for ownership transfer, *i.e genesis transaction*, was created only by the registered IP owner. This verification is done by consortium blockchain in our protocol. The consortium can be formed of multiple semiconductor organizations, each of which operates a node in blockchain. Validation of a transaction requires a set number of nodes to sign-off on it.

The second requirement prevents an unauthorized party from hijacking the ownership of an IC by creating a fraudulent transaction. Our protocol grants only the current owner the ability to create a new transaction. The current owner's identity is established using the information in the last transaction recorded in the blockchain and also, by utilizing PUF authentication to verify physical possession of the IC.

B. Protocol for Blockchain Creation

1) Ownership addresses and keys: All the potential owners of an IC are assigned their own addresses, used during ownership transfer. An address is generated from ECDSA (Elliptical Curve Digital Signing Algorithm) public/private key pair. The major advantage of ECDSA over other Public Key Cryptography (PKC), such as RSA is that ECDSA uses much smaller keys and signatures to achieve the same level of security. First, a random 256-bit private key is generated. Each transaction is supplemented with a digital signature created using the private key. The signature uniquely identifies the current owner as the *seller*.

$$K_{priv} \in \{0,1\}^{256}$$

Next, a 512-bit public key is generated from the private key using ECDSA algorithm. The public key is used for verification of the transaction signature. The public key is not revealed until a transaction is signed, unlike most systems where the public key is made public.

$$K_{pub} = ECDSA_{512}(K_{priv})$$

Since the 512-bit public key is large, it is converted to a smaller address that can be shared with others and utilized as part of the blockchain transaction. The 512-bit public key is double hashed using SHA-256, which is further hashed using RIPEMD-160 to generate the 160-bit address.

$$\mathrm{K}_{\mathrm{address}} = \mathrm{RIPEMD}_{160}(\mathrm{SHA}_{256}((\mathrm{K}_{pub})))$$

The consortium keeps a database that binds the identity information of an associated party with transaction address.

2) Transaction implementation: The general structure of a transaction in our protocol is shown in TABLE I. We denote the current and the new owners as the *seller* and *buyer*, respectively. A transaction input contains the reference to the previous transaction, seller's signature, seller's public address, and IC information. A transaction output contains the buyer's address and the transaction value. The salient features of a transaction are described as follows:

TABLE I
FORMAT OF A TRANSACTION FOR OWNERSHIP TRANSFER

	Field	Description	Size (Bytes)
	inputCount	Number of ICs to be sold	1
Input(s)	prevTxID	The previous transaction reference	32
	challenge	Challenge to the PUF	variable
	hashResponse	Hash of the Responses from PUF	variable
	serialNumber	Identifier of an IC	variable
	icInfo	Other necessary information related to IC	variable
	signature	Seller's signature	71
	publicKey	Seller's public key for verifying the signature	64
Output(s)	value	Transaction Value	1
	publicKeyHash	Buyer's Address	20

- Our proposed protocol facilitates the seller to sell multiple ICs in the same transaction. Seller just needs to specify the the number of ICs to be sold in the $inputCount$ field of the transaction.
- For each IC, the corresponding previous transaction hash is referenced in the $prevTxID$ field. For a *genesis transaction*, this is set to 0. Each IC's PUF challenge-response data is also included in $challenge$ and $hashResponse$ field, respectively. We discuss the PUF data formatting in greater detail later. Other important IC information is included in the $icInfo$ field.
- The $serialNumber$ in the format serves as an identifier for an IC. This identifier (e.g. Electronic Product Code, or EPC) can be used to enable the verifier to look up the correct transaction for the IC being queried among a collection of ICs. Here, the serial number is being used for the purpose to *identify*, and not as the primary means to *authenticate*.
- A transaction includes the signature and the public key of the seller in $signature$ and $publicKey$ field, respectively. The seller generates the public key from private key. This key must match the hash given in the previous transaction output ($publicKeyHash$). The public key is also used to verify the seller's signature. The signature is an ECDSA signature over a hash of a raw version of the transaction. The signature, combined with the public key, proves that the transaction was created by the real owner.
- The transaction output nominates buyer's address in $publicKeyHash$ field. Any future transaction by the buyer will require the relevant public key and signature.

3) Incorporating a transaction to a block and creating a blockchain: After signing a transaction, the seller sends it into the consortium blockchain network, where the nodes pick up the transaction and verify the signature cryptographically. After verification, the blockchain node ensures that the referenced transaction has not been spent in a different transaction to prevent *double spending*. Finally, all the verified transactions that are considered to have happened at the same time, are placed in groups called blocks. TABLE II presents the structure of a block.

Each block has a reference to the previous block, and this is what places one block after another forming a chronological chain. A consensus mechanism ensures that the creation and modification of data are agreed by all the nodes or a majority of the nodes. Each transaction in a block is tied to a unique identifier ($TxID$), which is a double SHA256 hash of the verified, signed transaction. A $TxID$ is used to look up a transaction in the blockchain and referenced for future transaction.

TABLE II
STRUCTURE OF A BLOCK

	Field	Description	Size (Bytes)
Header	prevBlockHash	The hash value of the previous block used as a pointer	32
	timeStamp	A Unix timestamp recording when this block was created	4
	nonce	The block-specific nonce to allow variations of the header and compute different hashes	4
Transactions	txnCount	No. of transactions in block	1
	transaction	List of verified transactions	variable

C. Protocol for Ownership Transfer

Fig. 2 illustrates the detailed system model of the proposed ownership transfer protocol. The protocol consists of several phases which we describe as follows:

1) Sending seller's public key to the buyer: In order to allow the buyer to trace the IC supply chain information and authenticate it, the seller sends him the previous transaction ID ($TxID$) in the blockchain. The latest transaction information of an IC contains the current owner's address ($publicKeyHash$). In order to prove the genuineness of the seller, the potential buyer needs the corresponding seller's public key. So, in first step of any ownership transfer, the seller sends his public key (K_S^+) and a signature ($K_S^-(K_S^+)$) to the buyer. This signature and the public key proves that seller is the same person with the predetermined address.

2) Verification of the IC by the buyer: With the received public key (K_S^+), the buyer can query the previous transaction ($TxID$) information in the blockchain. Buyer can verify the genuineness of the ownership and authenticate the IC from the blockchain information. The occurs in the following two steps:

2a) Verifying the ownership information: Buyer applies hash to the public key which he has received from the seller. If the hash matches the referenced previous transaction, buyer can verify the genuineness of the current seller. Also, using $TxID$, buyer can obtain the transaction information which has reference to its previous transaction, which again has reference to its previous information and so on. In this way, buyer can trace back all the way along supply chain to IC's provenance.

2b) Authenticating the IC: Buyer can retrieve challenges from the blockchain and apply them to the IC. The PUF embedded in the IC will give the corresponding responses. If the calculated hash of the responses matches with the hash recorded in the blockchain, the IC is authenticated.

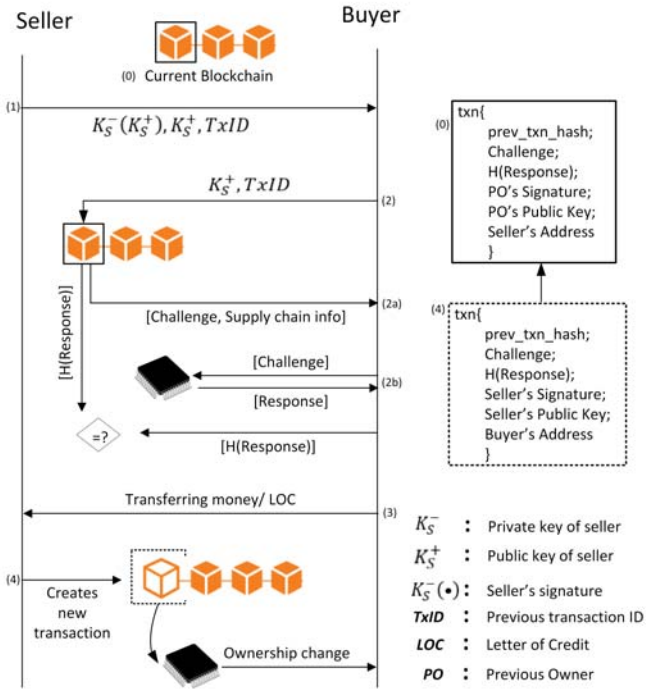
Seller
Buyer
(0) Current Blockchain
(1)
$K_S^-(K_S^+), K_S^+, TxID$
$K_S^+, TxID$
(2)
[H(Response)]
(2a)
[Challenge, Supply chain info]
[Challenge]
(2b)
[Response]
=?
[H(Response)]
Transferring money/ LOC
(3)
(4)
Creates new transaction
Ownership change
(0) txn{ prev_txn_hash; Challenge; H(Response); PO's Signature; PO's Public Key; Seller's Address }
(4) txn{ prev_txn_hash; Challenge; H(Response); Seller's Signature; Seller's Public Key; Buyer's Address }
K_S^- : Private key of seller
K_S^+ : Public key of seller
$K_S^-(\bullet)$: Seller's signature
TxID : Previous transaction ID
LOC : Letter of Credit
PO : Previous Owner

Fig. 2. Detailed diagram of the ownership transfer protocol.

3) Making payment: Once the IC is authenticated and the seller's IC ownership is verified, the buyer makes a payment to the seller. The payment can be made in terms of Letter of Credit (LOC) and only after the seller completes a transaction later, the money is transferred. This ensures that if a seller does not write the transaction, he will not get payment.

4) Creating a transaction: After receiving the confirmation of the payment, the seller is now ready to write the transaction for the ownership transfer. In the similar way described in section III-B2, seller creates a transaction. Seller puts the buyer's address in the output field of the transaction. If the transaction is successfully verified by the consortium and added to the blockchain, the IC is owned by the buyer.

IV. Security Analysis of the Protocol

The genesis block is assumed to have enough PUF data to disambiguate the IC from millions of other devices [14]. To increase the number of authentication events, the PUF can be re-mined during each ownership transfer event and the new data can be included in $challenge$ and $hashResponse$ fields of the transaction format. The hashing of PUF responses prevents their exposure in the blockchain and ensures only the physical owners of the IC can re-generate the hash. The re-mining process can also check the reliability of the PUF which can be affected by aging over the product lifetime and update the blockchain accordingly.

V. Protocol Demonstration and Discussion

We have implemented the proposed protocol in Python programming language and the code is made available online [15]. We emulate the blockchain as a Text File. In our emulated blockchain, a genesis transaction can be created by only selected private-public keys. Other transactions can be created by any private-public key. Transaction created with valid private-public key (whose corresponding address is the recipient of the referenced previous transaction) is verified and incorporated to a block.

The number of PUF challenges required to distinguish 1 trillion ICs is calculated to be $\sim$1024 for a Hamming distance threshold of 10% [14]. For practical implementation, using e.g. a 128-bit arbiter PUF, a set of 128 challenges can be taken and MD5-128 hash can be applied to the corresponding 128-bit responses. This adds to total 2176 bytes of data in the $challenge$, $hashResponse$ fields of a transaction.

VI. Conclusion

In this paper, we outline a novel methodology to establish IC traceability using blockchain technology and PUFs. The blockchain allows legitimate parties to track an IC over its entire lifetime. Embedded PUFs are used to provide a simple, secure and robust authentication process at every point-of-sale. The underlying technology guarantees the integrity of the system even in the face of dishonesty or idleness.

References

[1] U. Guin, K. Huang, D. DiMase, J. M. Carulli, M. Tehranipoor, and Y. Makris, "Counterfeit integrated circuits: A rising threat in the global semiconductor supply chain," *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1207–1228, 2014.

[2] L. W. Kessler and T. Sharpe, "Faked parts detection," *URL: http://www.circuitsassembly.com/cms/component/content/article/159/9937-smt.*, 2010.

[3] M. N. Islam and S. Kundu, "Modeling residual lifetime of an ic considering spatial and inter-temporal temperature variations," in *Asian Test Symposium (ATS), 2016 IEEE 25th*. IEEE, 2016, pp. 240–245.

[4] M. Pecht and S. Tiku, "Bogus: electronic manufacturing and consumers confront a rising tide of counterfeit electronics," *IEEE spectrum*, vol. 43, no. 5, pp. 37–46, 2006.

[5] M. N. Islam and S. Kundu, "An analytical model for predicting the residual life of an ic and design of residual-life meter," in *VLSI Test Symposium (VTS), 2017 IEEE 35th*. IEEE, 2017, pp. 1–6.

[6] P. Helo and B. Szekely, "Logistics information systems: an analysis of software solutions for supply chain co-ordination," *Industrial Management & Data Systems*, vol. 105, no. 1, pp. 5–18, 2005.

[7] R. Wattenhofer, *The science of the blockchain*. CreateSpace Independent Publishing Platform, 2016.

[8] P. Tuyls and B. Škorić, "Strong authentication with physical unclonable functions," in *Security, Privacy, and Trust in Modern Data Management*. Springer, 2007, pp. 133–148.

[9] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *Proceedings of the 44th annual design automation conference*. ACM, 2007, pp. 9–14.

[10] R. Koh, E. W. Schuster, I. Chackrabarti, and A. Bellman, "Securing the pharmaceutical supply chain," *White Paper, Auto-ID Labs, Massachusetts Institute of Technology*, pp. 1–19, 2003.

[11] K. Elkhiyaoui, E.-O. Blass, and R. Molva, "CHECKER: On-site checking in RFID-based supply chains," in *Proceedings of the fifth ACM conference on security and privacy in wireless and mobile networks*. ACM, 2012, pp. 173–184.

[12] J. A. Roy, F. Koushanfar, and I. L. Markov, "Ending piracy of integrated circuits," *Computer*, vol. 43, no. 10, pp. 30–38, 2010.

[13] J. Rajendran, Y. Pino, O. Sinanoglu, and R. Karri, "Logic encryption: A fault analysis perspective," in *Proceedings of the Conference on Design, Automation and Test in Europe*. EDA Consortium, 2012, pp. 953–958.

[14] P. Ramesh, V. C. Patil, and S. Kundu, "Peer pressure on identity: On requirements for disambiguating PUFs in noisy environment," in *2017 IEEE North Atlantic Test Workshop (NATW)*, May 2017, pp. 1–4.

[15] Ownership transfer blockchain protocol. [Online]. Available: https://github.com/nazmulislam025/Blockchain