IEICE TRANS. ELECTRON., VOL.E104–C, NO.7 JULY 2021

INVITED PAPER *Special Section on Solid-State Circuit Design — Architecture, Circuit, Device and Design Methodology*

Secure Cryptographic Unit as Root-of-Trust for IoT Era

Tsutomu MATSUMOTO[†a)], **Makoto IKEDA**[††], **Makoto NAGATA**[†††], ***and*** **Yasuyoshi UEMURA**[††††,†††††], ***Members***

SUMMARY The Internet of Things (IoT) implicates an infrastructure that creates new value by connecting everything with communication networks, and its construction is rapidly progressing in anticipation of its great potential. Enhancing the security of IoT is an essential requirement for supporting IoT. For ensuring IoT security, it is desirable to create a situation that even a terminal component device with many restrictions in computing power and energy capacity can easily verify other devices and data and communicate securely by the use of public key cryptography. To concretely achieve the big goal of penetrating public key cryptographic technology to most IoT end devices, we elaborated the secure cryptographic unit (SCU) built in a low-end microcontroller chip. The SCU comprises a hardware cryptographic engine and a built-in access controlling functionality consisting of a software gate and hardware gate. This paper describes the outline of our SCU construction technology's research and development and prospects.

key words: *IoT, security IP, public-key cryptography, root of trust*

1. Introduction

Enabling things to communicate freely with each other will lead to the creation of new values. The international standard ISO/IEC 20924:2018 defines IoT as an infrastructure of interconnected entities, people, systems and information resources together with services which processes and reacts to information from the physical world and virtual world [1]. Figure 1 shows a model of the IoT from a device perspective, classifying end nodes such as sensors and actuators, intermediate nodes which are network devices, and upper nodes where large amounts of data are stored and processed.

Let us consider the future development of the IoT architecture. Looking at the many IoT systems that have been built to date, we can see that they are vertically integrated and managed from the upper nodes to the end nodes by domain or corporate group, and data exchange between these IoT verticals is done loosely via the cloud (the upper nodes). In the future, however, there will be a meshing of data distribution among the various layers of IoT, regardless of domain or business owner, a multiplication of layers of services, and virtualization, and the potential for multiple stakeholders to connect in diverse ways.

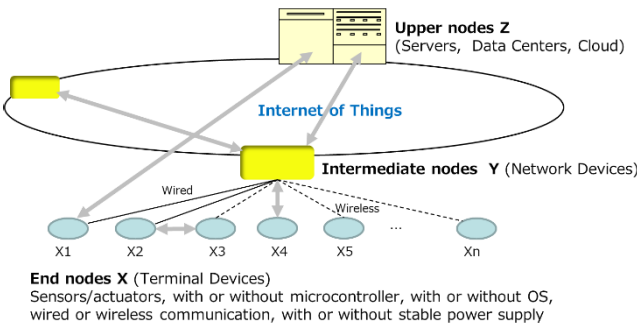

Fig. 1 A model of Internet of Things.

This paper proposes a secure cryptographic unit (SCU), which is a root of trust providing a secure public key cryptographic capability that can be embedded in a microcontroller chip or systems on a chip (SoC), that constitutes the end node of such IoT. The core technology of SCU was developed by Cross-ministerial Strategic Innovation Promotion Program (SIP) 1st Phase, "Cybersecurity for Critical Infrastructure." The details of the program with respect to SCU are complemented by reference [2].

The rest of this paper is organized as follows. Section 2 describes what must be done to raise the level of IoT security. The concept and idea of the SCU are then presented in Sect. 3; the cryptographic engine part of the SCU is described in Sect. 4, and the tamper-resistant technology of the SCU is detailed in Sect. 5; the utilization of the SCU is described in Sect. 6. The security assurance and interoperability for practical use of the SCU are discussed in Sect. 7, followed by conclusions of this paper in Sect. 8.

2. How to Establish IoT Security

2.1 Threat Analysis and Security Goal

Let's think about how security should be in such an IoT. We analyzed the threats in the IoT based on the fact that threats interact in both the physical and cyber worlds, and that there are attacks that attempt to shake the certainty of the correspondence between the physical and cyber worlds [3]. As a result, we decided to take as a fundamental necessary technical goal the realization of mutual authentication by cryptography and the confidentiality and integrity of data, includ-

Manuscript received December 21, 2020.
Manuscript revised January 11, 2021.
Manuscript publicized January 28, 2021.
[†]The author is with Yokohama National University, Yokohama-shi, 240–8501 Japan.
[††]The author is with the University of Tokyo, Tokyo, 113–0032 Japan.
[†††]The author is with Kobe University, Kobe-shi, 657–8501 Japan.
[††††]The author is with ECSEC TRA, Tokyo, 101–0054 Japan.
[†††††]The author is with National Institute of Advanced Industrial Science and Technology (AIST), Tokyo, 135–0064 Japan.
a) E-mail: tsutomu@ynu.ac.jp
DOI: 10.1587/transele.2020CDI0001

ing programs, among all components in the scope of the IoT.

2.2 Cryptography

There are two main types of cryptography: symmetric key cryptography and public key cryptography. In symmetric key cryptography, the sender and receiver use the same secret key, and the computational complexity of the cryptographic process is small, but the cost of key management is high. Public key cryptography, on the other hand, is a method in which the sender and receiver use different keys, one of which can be made public, and the cost of key management is relatively low but the computational complexity of the cryptographic process tends to be high.

The first widespread public-key cryptosystem was RSA, but it tends to increase the key length to maintain sufficient security in the future, which limits its applicability to IoT end nodes. For this reason, the adoption of elliptic curve cryptography is reasonable. Elliptic curve cryptography is a public-key cryptosystem whose security is based on the fact that an elliptic curve consisting of points (x, y) satisfying some equation y^2 = (cubic polynomial in x) over a finite field and the point at infinity form a finite group with respect to the addition of points, and that on this group or on its subgroup, a problem called the discrete logarithm problem is computationally intractable.

2.3 Create a Situation in Which Public Key Cryptography Can Be Used Freely by IoT End Nodes

Of course, nodes with rich computing power could be equipped with security chips such as trusted platform modules (TPMs) [4], so there was no problem in using public key cryptography. However, end nodes of the IoT with extremely limited computing power and/or energy capacity have often no cryptographic functions, or even if they have cryptographic functions, they are limited to symmetric key cryptography. In order to achieve the above goal, i.e., end-to-end security, the introduction of public key cryptography is essential. For IoT devices, public key cryptography must be made available as a matter of course without any special effort, even to the end nodes, which have many limitations.

As a trump card to achieve the challenging goal of penetration of public key cryptography to end nodes, we propose the concept of Secure Cryptographic Unit (SCU) to realize secure cryptographic functionality embedded in IC chips (Fig. 2).

3. SCU: Secure Cryptographic Unit

3.1 The Concept of SCU

IoT devices need to be equipped with appropriate security functions at the hardware, software, and logical levels. For example, hardware level security includes cryptographic key management mechanisms and cryptographic co-processors,

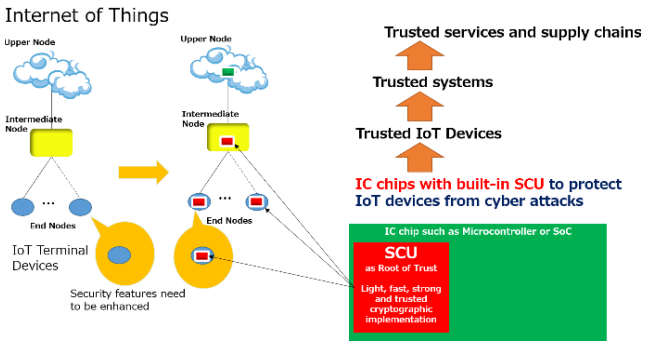

Fig. 2 Securing every part of the IoT.

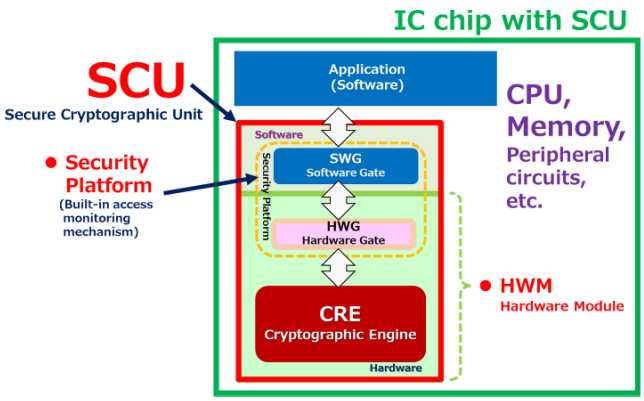

Fig. 3 Architecture of secure cryptographic unit.

while software level security functions include secure software update and remote attestation technologies.

The SCU is represented as a security intellectual property (IP) that realizes these hardware-level security functions. The security functions of the SCU include the installation of secure parameters, secure management of cryptographic keys, generation of physical true random numbers, cryptographic arithmetic functions, and an access monitoring mechanism for the SCU hardware. The access monitoring mechanism is a function that can detect tampering of the software from the hardware side of the SCU. This is embodied with a specially designed software that have a privilege to access the part of SCU hardware in performing SCU's cryptographic operations.

This mechanism enables the SCU to achieve a higher level of security in comparison to traditional hardware cryptographic modules. The installation of the SCU into a microcontroller or relevant SoCs is the way to realize secure IoT devices. We note that the concept of SCU was inspired by the target of evaluation (TOE) in the protection profile (PP) of a secure IC chip for embedded devices [5].

3.2 The Architecture of SCU

As shown in Fig. 3, SCU is a cryptographic IP consisting of a software gate (SWG), a hardware gate (HWG), and a cryptographic engine (CRE). The hardware gate and the cryptographic engine are collectively called the hardware module (HWM) of the SCU, defined as hardware IP. The user who

develops or purchases the hardware IP of the SCU (the user here is assumed to be the chip manufacturer) implements the IP of the SCU in a microcontroller or SoC and uses it.

The software gate and the hardware gate form a security platform. The security platform exists for the access monitoring function described below. The software gate is dedicated software to execute cryptographic operations on the hardware gate, and is called by application programs through a dedicated application programming interface (API). The software gate can issue commands to the hardware gate to operate them.

The hardware gate is a circuit that is executed from the software gate and performs operations using the cryptographic engine. The hardware gate has a function to detect whether the software gate has been tampered with, and stops operation when it detects tampering of the software gate.

The cryptographic engine contains hardware implementations of cryptographic functions including elliptic curve cryptography, symmetric key cryptography, message authentication codes (MACs), cryptographic hash functions, and a random number generator.

3.3 Key Management Mechanism of the SCU

The SCU has a function to securely store user cryptographic keys. The SCU's key management function ensures that cryptographic keys never leave hardware gate in plaintext to the outside world such as random access memory (RAM), non-volatile memory (NVM), central processing unit (CPU) and others. Whenever a key is retrieved from the SCU for storage in the SCU's external NVM, it shall be encrypted with the key held in the hardware gate, and shall be retrieved with the MAC attached.

3.4 Cryptographic Functions of the SCU

The SCU is equipped with cryptographic engine (CRE), and it can be used by user programs. Cryptographic operations using the CRE can only be executed from the user program by calling software gates. The SCU is equipped with elliptic curve cryptography, symmetric key cryptography, message authentication codes (MACs), cryptographic hash functions, and random number generators.

3.5 Access Monitoring Mechanism of SCU

The access monitoring mechanism of the SCU is realized by the Security Platform, which consists of the pair of hardware gate (Fig. 4) and software gate (Fig. 5). In the software gate, a value calculated based on the hardware behavior when the software gate is executed (hereafter referred to as the expected value) is encrypted with the hardware gate key. At software gate execution time, the hardware gate recalculates the expected value based on its own behavior. If the recalculated expected value and the expected value embedded in the software gate match, the hardware gate (Fig. 5) determines that it is a legitimate execution of the software gate

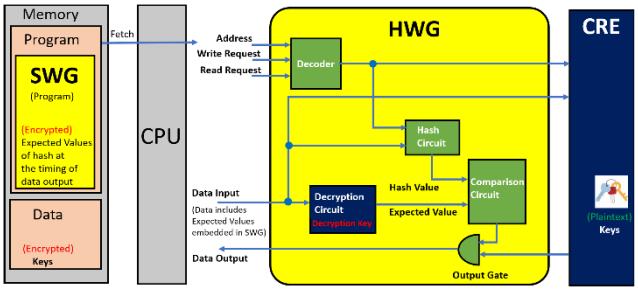

Fig. 4 Internal structure of hardware gate.

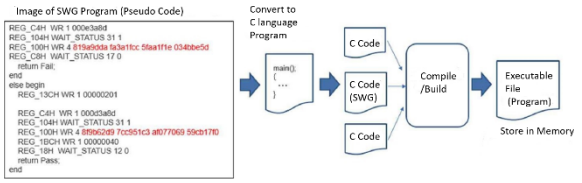

Fig. 5 How to create a software gate.

and outputs the calculation result. On the other hand, if the expected values do not match, the hardware gate determines that an illegal software gate has been executed, does not output the calculation results, and stops operation. In order to resume operation, a reset of the chip is required.

3.6 Type of SCUs

We develop two families of SCU hardware modules. One is the KM10 series including KM14, KM15, and KM16 to be described in Sect. 4. The KM10 series, as a family of SCU hardware modules developed with performance-oriented design techniques targeting small size, low power consumption, and low latency. The KM10 cores can be integrated into low-end microcontrollers and manufactured in general CMOS processes including very advanced technology nodes. The other is the KM20 series. This is a family of SCU hardware modules in throughput-oriented highly scalable design methodology, adopting multi-chip system-in-package (SiP) integration and advanced secure packaging technologies toward high tamper resistance.

4. Cryptographic Engine

4.1 Overview of CRE

Figure 6 shows the overall architecture of the developed SCU consisting of cryptographic engine and security platform. The hardware cryptographic engine involves crypto cores of AES-128, Chacha20, Poly1305, SHA256 and RNG, along with elliptic-curve cryptography (ECC). As our target SCU design is to popularize public-key cryptography into all kinds of IoT devices, we focus on ECC core design hereafter.

In addition, data and parameters are exchanged between the MCU and the elliptic-curve cryptographic (ECC)

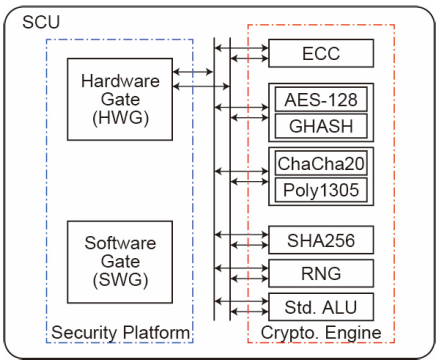

Fig. 6 Overall architecture of cryptographic engine (CRE) ECC covers elliptic curve digital signature algorithm (ECDSA).

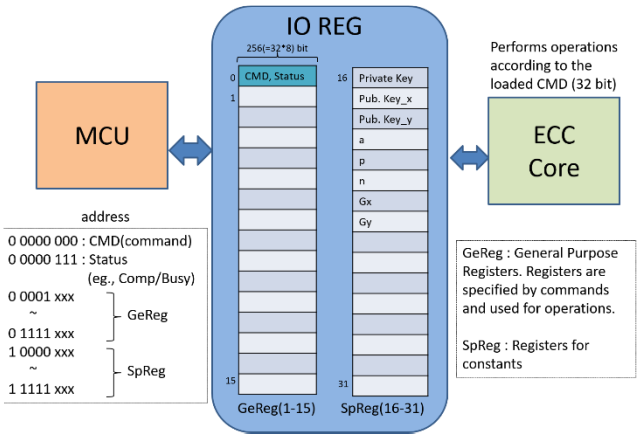

Fig. 7 IO mapped interface between MCU and ECC core.

core through the hardware gate in input-output mapped register shown in Fig. 7. This design enables the elliptic-curve cryptographic engine to be replaced easily according to the required performance. We have carried out design space exploration, as shown in Fig. 8, in terms of hardware costs and execution time, changing radix of the arithmetic unit as one of parameters. Here, we assumed a short Weierstrass curve, namely a curve defined by equation of the form $y^2 = x^3 + ax + b$, and the architecture based on the Montgomery ladder method in the Montgomery region, using the Jacobian coordinates. We have designed and optimized the arithmetic unit from 8-bit to 256-bit radices, for P-256 curve, and obtained the hardware cost (area) and execution time, which is the product of critical path delay and number of clock cycles.

All the results are obtained by Synopsys Design Compiler. Based on the results shown in Fig. 8, we will describe the design and optimization of KM14, which is targeted to realize the world's smallest ever reported elliptic-curve cryptographic engine design, KM15, the fastest ever reported elliptic-curve cryptographic engine design, along with KM16, the optimal design for application specific performance, in the following sections.

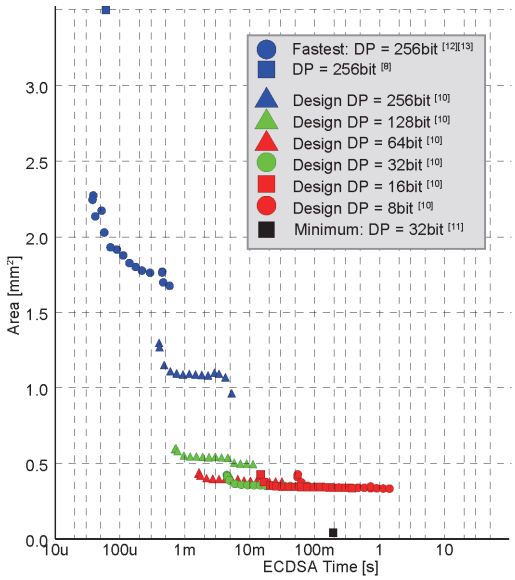

Fig. 8 Design space exploration for p-256 scaler multiplication on short Weierstrass curve.

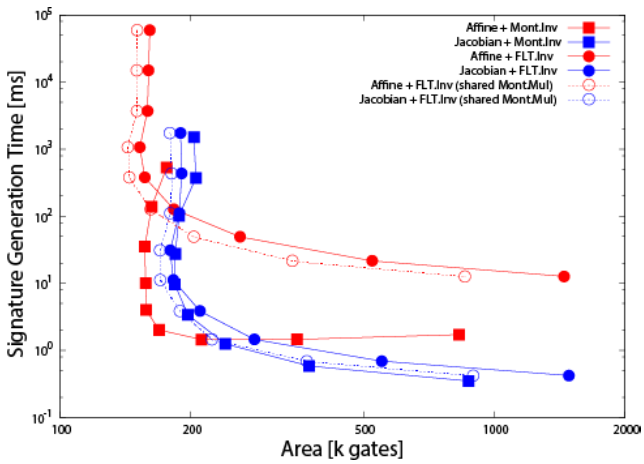

Fig. 9 Selection of coordinate for various radix of functional units. The Jacobian coordinate system shows smaller area for higher-radix systems, but the Affine coordinate systems results in smaller for lower-radix cases.

4.2 KM14: Smallest Ever Reported ECDSA Engine

The target of KM14 is to apply public-key cryptography to all IoT devices even in very tiny, and hardware resource limited MPUs like 32-bit and 16-bit MPUs. Therefore, as shown in Fig. 9, we have optimized and shared the arithmetic units as much as possible, eliminated intermediate registers, optimized the state-machines, and employed SRAM instead of using FFs as for the resister files. According to Fig. 9, we employed the Affine coordinate system for KM14, although it is usually regarded that the Jacobian coordinate system results in the smaller hardware.

Figure 10 shows the layout of the KM14 elliptic curve cryptographic core design (Fig. 10 (a)) and the entire KM14 design (Fig. 10 (b)), both in SOTB 65 nm CMOS process.

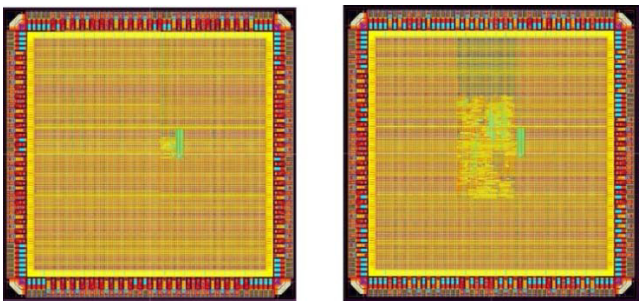

Fig. 10 Layout of (a) KM14 elliptic curve cryptographic core, and (b) entire KM14 CRE.

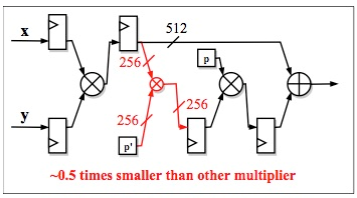

Fig. 11 p-256 Fp Montgomery multiplier with 7-stage pipeline.

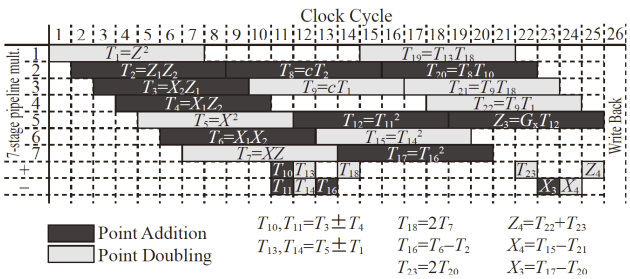

Fig. 12 Scheduling results of the Montgomery ladder stage using the 7-stage pipelined Montgomery multiplier shown in Fig. 11.

Number of logic gates of KM14 core is 13kG normalized by 2-input NAND gate and total number of clock cycles required for signature generation is 19.4M. Measurement results show signature generation time is 250 msec, and the power consumption is 0.16 mW. KM14 core occupies 160 um × 240 um in 3 mm square die, and KM14 occupies 680 um × 1,120 um.

4.3 KM15: Fastest Ever Reported ECDSA Engine

The KM15 is intended to be included in high-performance and hardware rich MCUs, and to be applied to applications that require high-speed signature generation and verification such as vehicle-to-vehicle communication. We have designed and optimized the Montgomery multiplier with 7-stage pipeline structure, shown in Fig. 11, and carried out scheduling optimization for Montgomery ladder stage, results in 27 clock cycles. We have designed using SOTB 65 nm CMOS process, with 1,580kG in normalized by 2-input NAND gate. The number of clock cycles required for signature generation is 7,500 clocks, and measurement results show signature generation time of 31 us with power consumption of 1.2 W. Effective area of KM15 is 2.35 mm × 2.40 mm in 3 mm square die.

Fig. 13 Die photo of KM15.

Table 1 Performance comparisons of KM14, KM15 and KM16.

	Plat.	#Gate [kG]	Area [mm²]	#Clk	Vdd [V]	Freq [MHz]	Tsg [ms]	Pow. [mW]	E [uJ]	Enc/ kG	Enc/ uJ
KM14 [11]	65nm	13	0.03	19.4M	0.75	77	250	**0.16**	100	0.31	0.04
KM15 [12] [13]	65nm	1,580	5.64	**7.5k**	0.45	35.7	0.21	15.6	3.28	3.01	**1,452**
					0.75	98.0	0.076	123	9.32	8.33	**1,412**
					1.4	238	**0.031**	1,227	38.7	**20.4**	834
KM16	65nm	~300		1.0M			~10	~100			
[6]	Stratix II (90nm)	9,177A LM+96 DSP	--	107k	--	157	0.32	--	--	--	--
[7]*	90nm	540	2.72	22.3k	--	131	0.17	--	--	10.9	--
[8]*	65nm	2,500	--	15k	--	236	0.06	--	--	6.67	--
[9]	65nm	1,370	1.92	34.7k	0.25		11	**0.15**	**1.68**	0.07	54.1
					0.3		2.3	0.69	**1.68**	0.32	259
					1.1		0.33	42.9	13.9	2.21	218

*: Synthesis results

4.4 KM16: Application Specific Optimized ECDSA Engine

KM16 shows and example design of CRE, which realizes a speed that meets the required specifications with the affordable hardware costs, rather than the extremely small realization like KM14 and extremely high speeds like KM15.

Table 1 shows performance comparisons of KM14, KM15 and KM16 with the state-of-the art design of the elliptic curve cryptographic engines.

5. Tamper Resistance Implementation

5.1 Tamper Resistance of SCU Chip

SCU needs to be tamper resistant when it is integrated on a security IC chip. The attacks of primary importance in the usage scenarios of SCU include local electromagnetic analysis (LEMA) and laser fault injection (LFI). SCU has been designed for an SC attack resistance in multiple physical layers at the cryptographic logic level and IC chip physical level, as detailed in Sects. 5.2 and 5.3, respectively, with

the trade-off considerations among operation speed, power consumption, Si area and tamper tolerance. Prototype IC chips and systems have been developed and demonstrated.

5.2 Logic-Level LEMA Resistance

SCU is implemented with CMOS logic gates physically on the frontside of an IC chip.

The data path of ECDSA in SCU is equipped with arithmetic units of modular multiplication (EC point doubling) and modular addition (EC point addition) to implement scalar multiplication based on the Montgomery ladder method. Its logical operation has compacted operation sequences as outlined in Fig. 14, which are elaborately designed for minimizing computation delay by applying Jacobian projective coordinate system. The consecutive sub-computation units from A1 to A17 and also from D1 to D11 are broken down for executing EC point addition and doubling, respectively.

The LEMA on the ECDSA signature generation revealed SC leakage even with the Montgomery ladder, powerfully by the similarity evaluation among consecutive EM waveforms of multiple hundreds of clock cycles, for specific logical sub-computation units as depicted in the bottom part of Fig. 14 (a). The units of A1, A2 and D2 are specially processed for the same arithmetic operation ($Z \times Z$) on the operands of either Z_1 or Z_2, according to the conditional branch with the polarity of k_i which is the i_{th} bit of a nonce, k. The EM waveform either in A1 or in A2 exhibits the larger similarity to D2 and unavoidably distinguishes the binary value of k_i. Once a full-length EM wave is provided, the binary values of k are successively derived by taking similarity analysis over every bit of k. This is the root source of SC information leakage. It is known that a full secret key can be regenerated from the nonce if a signature (r, s) and a hash value (e) are visible to an adversary.

The aforementioned LEMA vulnerability was first found in this project for ECDSA [14], and successfully prevented in [15]. By exchanging the processing order of operands at A1 according to k_i, as depicted in Fig. 14 (b). The largest similarity always stays in the position of A1 regardless of its polarity, which effectively eliminates the potentiality of key guesses.

5.3 IC-Chip Level Resistance on Direct Probing and LFI

SCU is implemented with CMOS logic gates physically on the frontside of an IC chip. Conventionally, the backside of CMOS circuits is formed by a p-type Si substrate with its thickness of 350 um or even thinner, and then attached to a supporting flame of a package or bonded to a plastic substrate of an interposer or even to a printed circuit board. It is often used in a low-cost semiconductor product that an IC chip is assembled by face-up packaging with bonding wires on its periphery. An adversary scans the frontside of a cryptographic engine with a micro-sized probe or a tiny magnetic probe for potential SC leakage through direct probing.

On the contrary, a high-performance very large-scale IC chip prefers to be packaged in a flip chip structure since huge number of input/output (I/O) pads can be evenly connected to an interposer with u-bumps arrayed on post-CMOS process redistributed layer (RDL) metal patterns on its frontside. The frontside cryptographic circuits become effectively away from an adversary, however, there is a known vulnerability to the substrate SC leakage where the whole backside of a Si chip is exposed to an open space. A variety of backside attack scenarios are depicted in Fig. 15. In order to protect the exposed backside of a flip-chip structure, the backside buried metal (BBM) technology is applied.

A backside direct probing provides an attacker with the easiest and straightforward access to the power consumption current of a cryptographic engine, I_{CC}, during its active operation. The I_{CC} produces voltage variation in the whole p-type Si substrate through p^+ contacts on its return paths (V_{SS}) and leads to Si-substrate noise SC leakage.

Infrared (IR) laser pulses penetrate into the Si substrate and interact with pn junctions of metal-oxide-semiconductor (MOS) transistors. The flip-chip IC packaging facilitates LFI attacks, again, since the whole area of IC chip backside is exposed to an adversary. Resin coating on the backside can be physically as well as chemically removed if an IC

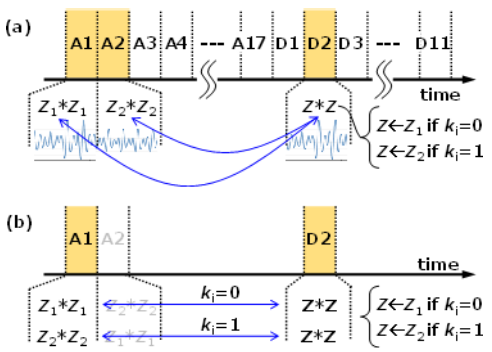

Fig. 14 Logic-level LEMA on SCU with computation sequences of scalar multiplication in (a) conventional and (b) proposed algorithm of ECDSA.

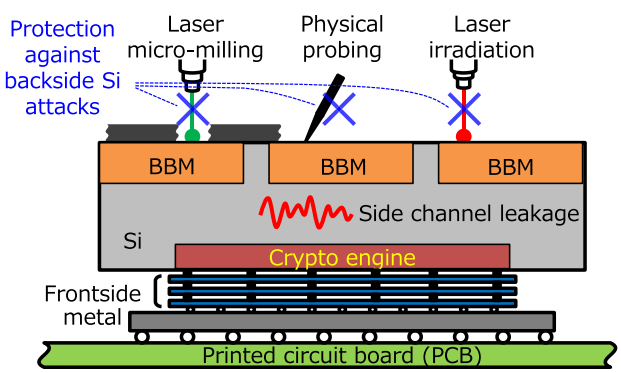

Fig. 15 Secure packaging with BBM technology against backside attack scenarios on Si IC chip.

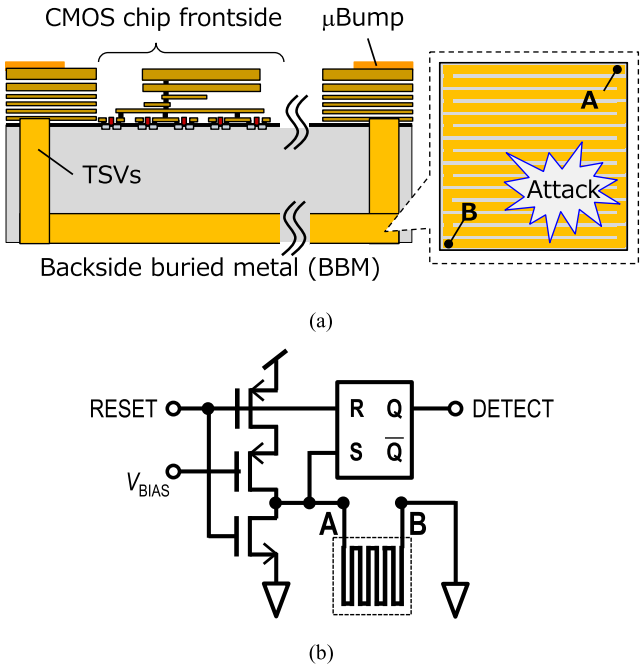

Fig. 16 BBM technology for backside protection of secure IC chip. (a) Cross sectional sketch with backside meander and (b) frontside circuit for detecting disconnection of backside meander.

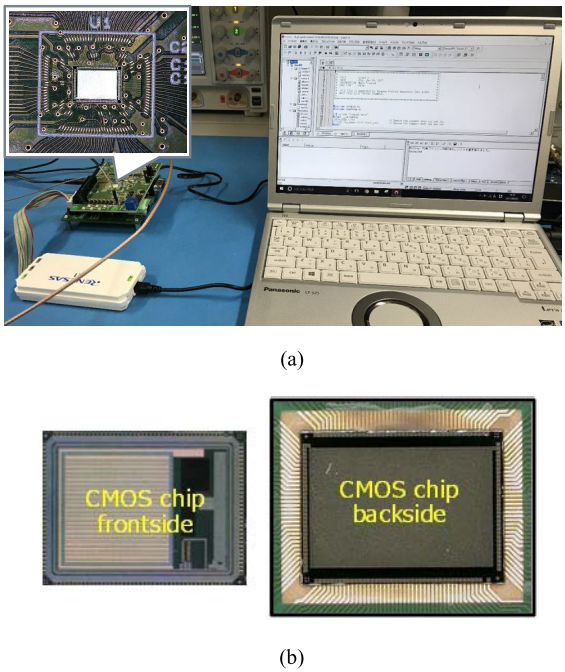

Fig. 17 (a) System level demonstrator of IC chip in secure packaging. (b) Die photos on frontside and backside of 130 nm CMOS IC chip (KM20) embedding ECDSA engine.

chip is molded.

The BBM, in a practical implementation of Fig. 16 (a), is patterned on the backside of a Si substrate and connected to the frontside CMOS circuits with through Si vias (TSVs). BBM and TSV are seamlessly integrated through a post-CMOS via-last wafer level processing flow, which was reported for the first time from this research project [16].

The BBM stripes, which is shaped in a meander pattern and additionally biased in an isolated voltage domain, provide the functionality of backside electronic protection and intrusion detection.

The BBM processing technology has been successfully established and demonstrated over 130 nm CMOS wafers. The system-level demonstrator of Fig. 17 confirms that the performance of frontside CMOS circuits, including ECDSA engines as a part of SCU (KM20) is unchanged after the BBM processing.

The BBM demonstrator realizes a backside meander with the width and space of 15 um and 10 um, respectively, which are narrow enough to repel probing needles with the tip diameter of typically larger than 50 um. The thickness of BBM are 10 um and buried to the thinned Si substrate of approximately 40 um. The needles are then forced to sense the bias voltage of meander stripes, and hidden from Si substrate voltage variation. The suppression of backside SC leakage was demonstrated for 25 dB and larger [17]. The BBM stripes effectively protect a Si substrate from the direct probing of SC leakage, which can be applicable even to the finely scaled device technology used in the frontside circuits.

The BBM meander again protects crypto engines on the frontside from the backside illumination of IR laser. The core part of crypto engines, e.g. data register, can be effectively hidden by the BBM stripe. In addition, the disconnection or even partial removal of BBM stripes, by intentionally illuminating focused green laser (e.g. 532 nm wavelength), is detectable with the circuit of Fig. 16 (b) [17]. Once the BBM of an IC chip is laser-cut or mechanically milled, the detector circuitry on the CMOS frontside immediately detects the backside disconnection.

This electronic anomaly recognition triggers on-chip security protocols. The higher level of attack avoidance is then achieved with the BBM barriers against physical intrusions.

6. SCU Usage

6.1 Implementing SCU into System LSI

Since SCU is an IP, a part of system LSI, there are two ways to implement SCU on the system LSI.

The first way is for chip vendors to purchase SCU's as IP and implement them on system LSI chips in their products. We already found such developer at the end of SIP 1st (Cross-ministerial Strategic Innovation Promotion Program, Phase 1, 2015–2019). That vendor is currently developing a "SCU-implemented system LSI chip" using the SCU as IP, the result of the national project. They have developed many types of embedded devices and will expand the use of the chip if they are successful in their first public use.

On the other hand, application users generally expect embedded devices to implement security chip rather than IP.

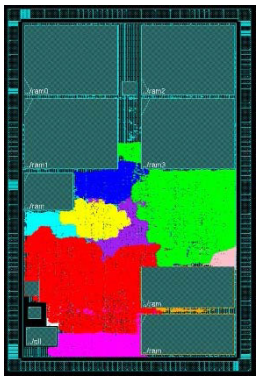

Fig. 18 Design of "SCU implemented system LSI chip" SC01.

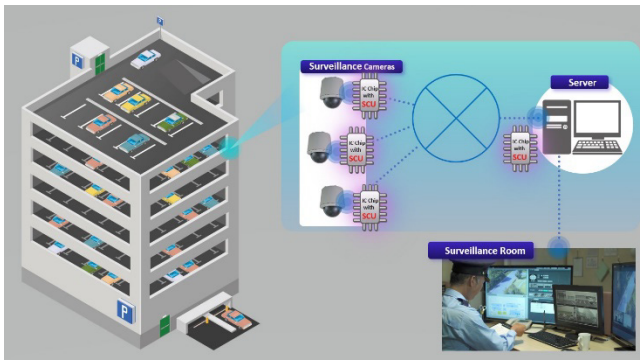

Fig. 19 Surveillance camera system using SCU (image).

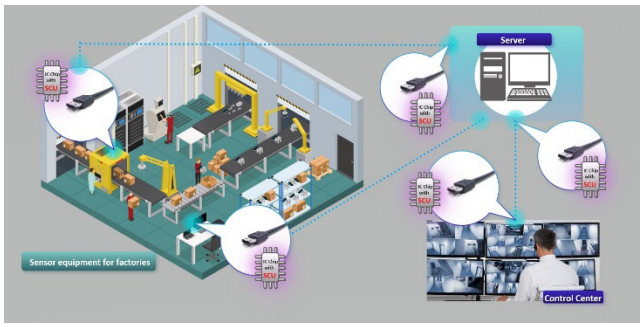

Fig. 20 Connector system using SCU (image).

To this end, we are developing our own "SCU-implemented system LSI chip" in SIP 2nd (Cross-ministerial Strategic Innovation Promotion Program, Phase 2, 2019–2023).We are focused on developing very small chips for embedded devices with fewer resources such as sensor nodes and actuators. Figure 18 shows the design of our first proprietary "SCU-implemented system LSI chip", named SC01.

After the development of such chips in SIP 2nd, ECSEC Technology Research Association, a member of the research team, will transform its corporate status to the private company and become a core member of the chip business consortium. The new company will perform "turnkey business" tasks, selling and maintaining the "SCU-implemented system LSI chip" and related IoT systems.

6.2 Advantage

Over the long term of two national projects, SIP 1st and 2nd, we have investigated the benefits and capabilities of "SCU-implemented system LSI chip" in the security market.

We know that there are many prior cores, kernels, or chips that is designed to be used to secure the system. At the end of this research, we have come to the conclusion that chip size will be the advantage of SCU.

"SCU-implemented system LSI chip" will also be integrated into embedded devices with fewer resources, such as sensor nodes and actuators. This opens up the future of the use of public key cryptography for low-resource terminal devices in IoT systems. (See Sect. 4 for numerical evidences.)

6.3 Implementation to Embedded Systems

To demonstrate the reality of the social implementation of SCU, two model systems are being developed during SIP 2nd.

One is a surveillance camera system using SCU. In 2019, we demonstrated the use of SCU on the VMnex® [18] monitoring camera system from Hitachi Kokusai Electric Inc. (Fig. 19). At the time, SCUs were not implemented on system LSI chips. An independent SCU was on the board and was activated by the CPU of another system LSI chip connected to the board. After successful demonstration testing in 2019, we are attempting to use the "SCU-implemented system LSI chip" to be developed by the chip vendor on the VMnex® system in 2021.

The other one is the connector system. The connector system is expected to be used in a very wide range of fields. "SCU-implemented system LSI chip" can be embedded in the terminals of Ethernet, USB and other connectors. The function of SCU will be device-to-device authentication or data authentication with digital signage and verification. This type of connector system could also be used in factory control systems, office LAN systems or AI-controlled machines such as robot (Fig. 20). In 2021, our research team will implement SC01 in the connector system to drive technology development in the lab.

7. SCU, Practical Applications

7.1 Security Assurance

It is important to establish a security assurance scheme for SCU and the "SCU-implemented system LSI chip" because some predecessors already have well-organized security assurance scheme in place. We respect and follow the Common Criteria (ISO/IEC15408) concept. However, there are several problems with the direct integration of CC into the security assurance of SCU.

The first interference is "SCU is IP''. Current CC has no way to evaluate hardware IP as TOE since it is difficult

to perform vulnerability assessment to the IP itself. Second, while there are already properly managed vulnerability assessment methods in place for using smart cards, there are several different security situations for small chips in embedded devices. The chip may be too small to implement the usual tamper-resistance mechanisms on a large chip.

We are carefully studying to find reasonable and secure assurance techniques for limited resource system LSI chips. We then construct our own security assurance scheme for SCU and "SCU-implemented system LSI chips". The original scheme relies almost entirely on the concept of CC, but we add new ideas to adjust the current CC to hardware IP and less resource intensive chips.

At the end of SIP 1st Phase, we already prepared the Security Target template for SCU and disclosed on NEDO website (in Japanese language) [2]. During SIP 2nd Phase, we are going to prepare the Protection Profile for "SCU-implemented system LSI chip" and disclose it (in Japanese language). The attack methods and supporting documents are also described in preparation for the original security assurance scheme for SCU and "SCU-implemented system LSI chip".

7.2 Interoperability

Interoperability means how the market identifies "this is SCU". Technical issue is how to share the application interfaces, "APIs", between "SCU-implemented system LSI chips" manufactured by different vendors. In order to maintain interoperability, we are currently researching to identify enough information to share SCU APIs for different vendors.

In addition, the API information needs to be shared by the consortium of all SCU stakeholders. We are also researching how to build appropriate "SCU consortium" and how to fairly manage the SCU community. On the other hand, "SCU consortium" needs to share the same security requirements and security assurances in order to maintain the security of the system by SCUs. For that reason, we are currently attempting to establish its own security assurance scheme for SCU and "SCU-implemented system LSI chip". In this scheme, SCU consortium will take the initiative to certify the result of the security evaluation.

8. Conclusion

SCU can enable public key cryptography to be used in small sensors and actuators where public key cryptography has not been possible before, and can be a very powerful root of trust to help bring security to the end nodes of the IoT. Please look forward to the future social implementation of SCU, a technology that could become the foundation for building our prosperous and secure future [19].

Acknowledgments

This work was supported by the Cabinet Office (CAO), Cross-ministerial Strategic Innovation Promotion Program (SIP) 1st Phase, "Cybersecurity for Critical Infrastructure," and 2nd Phase,"Cyber Physical Security for IoT Society", JPNP18015 (funding agency: NEDO). The authors of this paper would like to thank all those involved in these programs.

References

[1] ISO/IEC 20924:2018 Information technology — Internet of Things (IoT) — Vocabulary
[2] NEDO Achievement Report Database, Report Control Number 20200000000133, Project Number P15011, June 2020.
[3] T. Matsumoto, "Perspectives on the field of hardware security research," Journal of IEICE, vol.103, no.1, pp.34–39, Jan. 2020 (in Japanese).
[4] Trusted Computing Group, "Trusted platform module (TPM) 2.0: A brief introduction," https://www.trustedcomputinggroup.org/wp-content/uploads/TPM-2.0-A-Brief-Introduction.pdf
[5] ECSEC.TRA, Protection Profile of Secure IC Chip for Embedded Devices, c0427, IPA, 2014.
[6] N. Guillermin, "A high speed coprocessor for elliptic curve scalar multiplications over $\mathbb{F}_p$," Cryptographic Hardware and Embedded Systems (CHES 2010), pp.48–64, 2010.
[7] S.-C. Chung, J.-W. Lee, H.-C. Chang, and C.-Y. Lee, "A high performance elliptic curve cryptographic processor over GF(p) with SPA resistance," 2012 IEEE International Symposium on Circuits and Systems (ISCAS), pp.1456–1459, May 2012.
[8] M. Tamura and M. Ikeda, "Montgomery multiplier design for ECDSA signature generation processor," IEICE Trans. Fundamentals, vol.E99-A, no.12, pp.2444–2452, Dec. 2016.
[9] M. Tamura and M. Ikeda, "1.68μJ/signature-generation 256-bit ECDSA over GF(p) signature generator for IoT devices," 2016 IEEE Asian Solid-State Circuits Conference (A-SSCC), pp.341–344, Nov. 2016.
[10] T. Ikeda and M. Ikeda, "Scalable processor design for elliptic curve cryptographic processor over GF(p)," Proc. 2016 IEICE Society Conference, A-7-6, Sept. 2016 (In Japanese).
[11] R. Saito, H. Awano, and M. Ikeda, "Implementation of small-footprint co-processor for elliptic-curve digital-signature-algorithm on 256-bit prime fields," Technical Report of Hardware Security of IEICE, HWS, March 2018 (In Japanese).
[12] S. Sugiyama, H. Awano, and M. Ikeda, "31.3 μs/signature-generation 256-bit $\mathbb{F}$p ECDSA cryptoprocessor," 2018 IEEE Asian Solid-State Circuits Conference (A-SSCC), pp.153–156, Nov. 2018.
[13] S. Sugiyama, H. Awano, and M. Ikeda, "Low latency 256-bit $\mathbb{F}_p$ ECDSA signature generation crypto processor," IEICE Trans. Fundamentals, vol.E101-A, no.12, pp.2290–2296, Dec. 2018.
[14] K. Koiwa, D. Fujimoto, Y. Hayashi, M. Nagata, M. Ikeda, T. Matsumoto, and N. Homma, "EM security analysis of compact ECDSA hardware," Proc. 2018 IEEE International Symposium on Electromagnetic Compatibility and 2018 IEEE Asia-Pacific Symposium on Electromagnetic Compatibility (EMC/APEMC 2018), abstract reviewed, May 2018.
[15] K. Koiwa, R. Ueno, D. Fujimoto, Y. Hayashi, M. Nagata, M. Ikeda, T. Matsumoto, and N. Homma, "Collision-based EM analysis on ECDSA hardware and a countermeasure," Proc. Joint International Symposium on Electromagnetic Compatibility and Asia-Pacific International Symposium on Electromagnetic Compatibility (Joint EMC & APEMC 2019), pp.793–796, Sapporo, Japan, June 2019.
[16] Y. Araga, M. Nagata, H. Ikeda, T. Miki, N. Miura, N. Watanabe, H. Shimamoto, and K. Kikuchi, "A thick Cu layer buried in Si interposer backside for global power routing," IEEE Trans. Compon. Packag. Manuf. Technol., vol.9, no.3, pp.502–510, March 2019. DOI: 10.1109/TCPMT.2018.2877211

[17] T. Miki, M. Nagata, H. Sonoda, N. Miura, T. Okidono, Y. Araga, N. Watanabe, H. Shimamoto, and K. Kikuchi, "Si-backside protection circuits against physical security attacks on flip-chip devices," IEEE J. Solid-State Circuits, vol.55, no.10, pp.2747–2755, Oct. 2020. DOI: 10.1109/JSSC.2020.3005779
[18] https://www.hitachi-kokusai.co.jp/products/camera/isnex/vmnex/index.html
[19] ECSEC.TRA, A Short Video Introduction to Theme A1 of the SIP 2nd Phase "Cyber physical security for IoT society" https://www.youtube.com/watch?v=ll6ZRcm6t-8&feature=youtu.be

Tsutomu Matsumoto is a professor of the Faculty of Environment and Information Sciences, Yokohama National University. He also serves as the Director of the Cyber Physical Security Research Center (CPSEC) at the National Institute of Advanced Industrial Science and Technology (AIST). Starting from Cryptography in the early '80s, Prof. Matsumoto has opened up the field of security measuring for logical and physical security mechanisms. He received a Doctor of Engineering degree from the University of Tokyo in 1986. He serves as the chair of the Japanese National Body for ISO/TC68 (Financial Services) and the Cryptography Research and Evaluation Committees (CRYPTREC) and as an associate member of the Science Council of Japan (SCJ). He received the IEICE Achievement Award, the DoCoMo Mobile Science Award, the Culture of Information Security Award, the MEXT Prize for Science and Technology, and the Fuji Sankei Business Eye Award.

Makoto Ikeda received the B.S., M.S., and Ph.D. degrees in electronic engineering from the University of Tokyo, Tokyo, Japan, in 1991, 1993, and 1996, respectively. He joined the Electronic Engineering Department, University of Tokyo, as a Faculty Member in 1996, and he is currently a full Professor with the Systems Design Lab, at the University of Tokyo. His interests include the hardware security, including cryptographic engine design, asynchronous system design and smart image sensor designs. He is a senior member of IEEE, IEICE Japan, and a member of IPSJ and ACM.

Makoto Nagata received the B.S. and M.S. degrees in physics from Gakushuin University, Tokyo, in 1991 and 1993, respectively, and a Ph.D. in electronics engineering from Hiroshima University, Hiroshima, in 2001. He is currently a professor of the graduate school of science, technology and innovation, Kobe University, Kobe, Japan. Dr. Nagata is chairing the Technology Directions subcommittee for International Solid-State Circuits Conference since 2018. He was a technical program chair, a symposium chair and an executive committee member for the Symposium on VLSI circuits. He is currently an AdCom member to the IEEE Solid-State Circuits Society and also serves as a distinguished lecturer (DL) in the society, both since 2020. He is an associate editor for IEEE Transactions on VLSI Systems.

Yasuyoshi Uemura received BAs in both Political Science and Philosophy from Keio University in 1975 and 1977. He established Electronic Commerce Security Technology Research Association (ECSEC.TRA) in 2000. He was Vice President of ECSEC.TRA (2000), Executive Vice President (2008), has been President since 2014 until now. Mr. Uemura also serves as a Research Advisor to the CPSEC at the National Institute of Advanced Industrial Science and Technology (AIST). His research work's main scope is Information Security of Hardware, especially on Smartcard, Embedded device, and System LSI. He was one of several persons who integrated Common Criteria (ISO/IEC15408) into Japanese society in its early stage.