

(12) 특허협력조약에 의하여 공개된 국제출원

(19) 세계지식재산권기구
국제사무국

(43) 국제공개일
2017년 8월 17일 (17.08.2017)



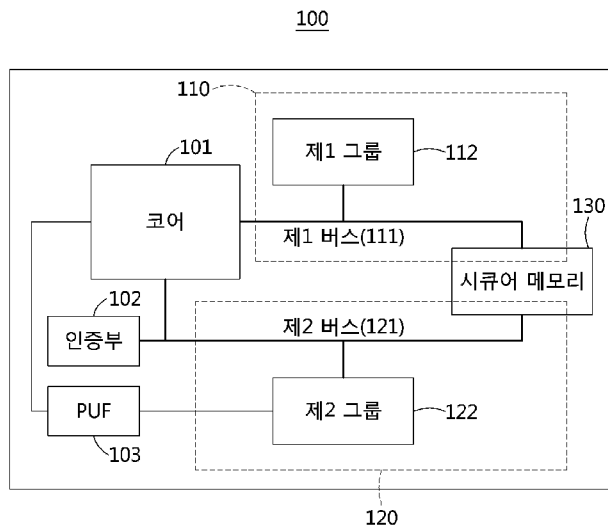
(10) 국제공개번호
WO 2017/138799 A1

- (51) 국제특허분류:
G06F 21/57 (2013.01) *G06F 15/78* (2006.01)
G06F 21/55 (2013.01)
- (21) 국제출원번호: PCT/KR2017/001560
- (22) 국제출원일: 2017년 2월 13일 (13.02.2017)
- (25) 출원언어: 한국어
- (26) 공개언어: 한국어
- (30) 우선권정보:
10-2016-0016587 2016년 2월 12일 (12.02.2016) KR
10-2017-0019497 2017년 2월 13일 (13.02.2017) KR
- (71) 출원인: 한양대학교 산학협력단 (INDUSTRY-UNIVERSITY COOPERATION FOUNDATION HANYANG UNIVERSITY) [KR/KR]; 15588 서울시 성동구 왕십리로 222, Seoul (KR).
- (72) 발명자: 김동규 (KIM, Dong Kyue); 05502 서울시 송파구 올림픽로 135, 226 동 1201 호, Seoul (KR). 김지훈 (KIM, Ji-Hoon); 13560 경기도 성남시 분당구 정자로 2, 2204 호, Gyeonggi-do (KR).
- (74) 대리인: 민영준 (MIN, Young Joon); 06296 서울시 강남구 남부순환로 2706, 3층 (도곡동, 차우빌딩), Seoul (KR).
- (81) 지정국 (별도의 표시가 없는 한, 가능한 모든 종류의 국내 권리의 보호를 위하여): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KH, KN, KP, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) 지정국 (별도의 표시가 없는 한, 가능한 모든 종류의 역내 권리의 보호를 위하여): ARIPO (BW, GH, GM,

[다음 쪽 계속]

(54) Title: HARDWARE DEVICE AND AUTHORIZATION METHOD THEREFOR

(54) 발명의 명칭: 하드웨어 디바이스 및 그 인증 방법



101 ... Core
102 ... Authorization unit
111 ... First bus
112 ... First group
121 ... Second bus
122 ... Second group
130 ... Secure memory

(57) Abstract: A security semiconductor chip is presented. The semiconductor chip is, for example, a system-on-chip. Processor cores included in the system-on-chip are connected to normal IPs through a system bus. A security bus, which is a hidden bus physically separated from the system bus, is provided separately. The security bus is connected to security IPs which perform security functions or handle security data. The security semiconductor chip may perform necessary authentication by switching a normal mode to a security mode.

(57) 요약서: 보안 반도체 칩이 제시된다. 반도체 칩은 이를 테면 시스템 온 칩이다. 시스템 온 칩에 포함되는 프로세서 코어에는 시스템 버스를 통해 통상의 IP들이 연결되어 동작한다. 상기 시스템 버스와 물리적으로 구분되는 히든 버스인 보안 버스가 별도로 제공된다. 보안 버스에는 보안 기능을 수행하거나 보안 데이터를 취급하는 보안 IP들이 연결된다. 보안 반도체 칩은 일반 모드와 보안 모드를 바꾸어 가며 필요한 인증을 수행할 수 있다.



KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ,
UG, ZM, ZW), 유라시아 (AM, AZ, BY, KG, KZ, RU,
TJ, TM), 유럽 (AL, AT, BE, BG, CH, CY, CZ, DE, DK,
EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU,
LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK,

SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ,
GW, KM, ML, MR, NE, SN, TD, TG).

공개:

— 국제조사보고서와 함께 (조약 제 21 조(3))

명세서

발명의 명칭: 하드웨어 디바이스 및 그 인증 방법

기술분야

- [1] 하드웨어 및 그 인증 방법에 연관되며, 보다 특정하게는 반도체 칩이나 전자 장비 등의 하드웨어 디바이스가 정품인지, 변조되거나 손상되지 않았는지 등을 인증하는 방법에 연관된다.

[2]

배경기술

- [3] 소프트웨어 기반 보안 기술에서는 OS(Operation System) 등의 취약점을 이용한 공격이 새로 나올 때마다 이에 대해 지속적으로 보안 업데이트를 해야 했다. 따라서 보안 공격을 방지하기 위해서 하드웨어 모듈 차원의 보안 공격 방지 설계와 운용이 제시되었다. 이러한 하드웨어 모듈 차원의 보안 방법으로, ARM사의 TrustZone 기술이 사용되고 있다.
- [4] TrustZone 기술과 같은 하드웨어 기반 보안 기술에서는 부팅 단계에서 OS 이미지의 무결성을 검증 후 읽어 들이기 때문에 플랫폼의 무결성을 시스템 시작 단계에서부터 스텝 바이 스텝으로 보장할 수 있게 된다. 이는 하드웨어가 OS를 비롯한 소프트웨어의 무결성을 검증하는 방식이다. 그러나, 소프트웨어의 시점에서 하드웨어의 무결성을 검증하는 것은 아니다.
- [5] 미국 등록 특허공보 US8,468,342호 (공고일 2013년6월18일)에는 시스템 부팅 시 OS 무결성 체크를 수행하여 안전한 부팅을 수행하는 컴퓨팅 장치가 제시되었다.
- [6] 기술문헌 "ARM Security Technology: Building a Secure System using TrustZone® Technology" (ARM Limited, White-Paper PRD29-GENC-009492C, April 2009)은 ARM사의 TrustZone 하드웨어 구조, 그리고 그 운용에 있어서의 Secure booting 등을 제시한다.

[7]

발명의 상세한 설명

기술적 과제

- [8] 본 발명은 하드웨어 디바이스의 인증을 위한 하드웨어 디바이스 및 그 인증 방법을 제공하기 위한 것이다.

[9]

과제 해결 수단

- [10] 일측에 따르면, 반도체 칩에 있어서, 프로세서 코어; 및 외부 신뢰 기관과 상호 인증을 수행하여 상기 반도체 칩 및 상기 외부 신뢰 기관 사이의 인증을 수행하는 인증부를 포함하는 반도체 칩이 제공된다.
- [11] 일실시예에 따르면 반도체 칩은, 상기 프로세서 코어에 제1 버스를 통해

연결되는 엘리먼트들을 포함하는 제1 그룹; 및 상기 프로세서 코어에 제2 버스를 통해 연결되며 보안 모드에서 동작하는 엘리먼트들을 포함하는 제2 그룹을 더 포함한다. 이 경우, 상기 인증부가 상기 인증에 성공한 경우에만 상기 제2 그룹을 사용 가능하도록 활성화 한다. 예시적으로, 그러나 한정되지 않게, 상기 제2 그룹은 Cryptographic IP, secure SRAM, secure DMA 및 boot ROM 중 적어도 하나를 포함한다.

- [12] 일실시예에 따르면 반도체 칩은, 상기 인증부가 상기 외부 신뢰 기관과 상기 상호 인증을 수행하는 데에 이용되는 키를 제공하는 PUF를 더 포함할 수 있다. 일실시예에 따르면 상기 제2 그룹은 상기 제1 그룹과 상이한 메모리 어드레스를 이용하여 데이터를 처리한다.

- [13] 예시적으로, 그러나 한정되지 않게, 상기 프로세서 코어는 32 비트 RISC(Reduced Instruction Set Computing) 타입의 임베디드 프로세서인 Core-A 프로세서를 포함한다. 이 실시예에서, 상기 제2 버스는 상기 Core-A 프로세서의 ASR (Application Specific Register)을 상기 제2 그룹에 포함되는 엘리먼트를 제어하는 주소 공간으로 사용하여 구현되는 히든 버스(hidden bus)를 포함한다. 이 경우, 상기 제2 그룹은 상기 보안 모드에서 상기 ASR에 대한 데이터 처리 명령인 MTA(Move to ASR) 명령과 MFA(Move from ASR) 명령어를 보안 명령어(secure instruction) 처리에 이용할 수 있다. 이는 ASR 인터페이스를 이용하여 상기 제2 그룹에 포함되는 보안(secure) IP들을 위한 히든 버스를 구현하는 것으로 이해될 수 있다. 다른 일실시예에 따르면 상기 프로세서 코어는 RISC V ISA 적용 CPU core일 수도 있다.

- [14] 일실시예에 따르면 상기 인증부는 하드웨어 로직이며, 기존의 컴퓨팅 언어와 상이한 독자 언어로 구동되어 상기 인증 절차를 수행한다. 또한, 상기 인증부는 기존의 컴퓨팅 언어와 상이한 독자 언어로 구성되는 인증 절차를 수행할 수 있다. 일실시예에 따르면 상기 인증부는 ROM이나 상기 반도체 칩 내의 IP와 다르게 일반 셀로 구성되어 PnR(Place and Route) 시 외부에 노출되거나 보안적으로 공격받지 않는다.

- [15] 다른 일측에 따르면, 소프트웨어를 임베드한 하드웨어 장치에 있어서, 상기 소프트웨어를 구동하는 프로세서 코어; 및 상기 하드웨어 장치를 통해 상기 소프트웨어를 구동하기 위해 외부 신뢰 기관과 상호 인증을 수행하여 상기 반도체 칩 및 상기 외부 신뢰 기관 사이의 인증을 수행하는 인증부를 포함하는 장치가 제공된다. 일실시예에 따르면 상기 장치는 상기 프로세서 코어에 노말 모드에서 동작하는 엘리먼트들을 포함하는 제1 그룹을 연결하는 제1 버스; 및 상기 프로세서 코어에 보안 모드에서 동작하는 엘리먼트들을 포함하는 제2 그룹을 연결하는 제2 버스를 더 포함할 수 있다. 일실시예에 따르면 상기 인증부가 상기 인증에 성공한 경우에만 상기 제2 그룹을 사용 가능하도록 제2 버스로의 접근이 활성화된다.

- [16] 일실시예에 따르면 상기 장치는 상기 인증부가 상기 외부 신뢰 기관과 상기

- 상호 인증을 수행하는 데에 이용되는 키를 제공하는 PUF를 더 포함할 수 있다.
- [17] 또 다른 일측에 따르면, 반도체 칩의 동작 방법에 있어서, 인증부가 외부 신뢰 기관과 상호 인증을 수행하는 단계; 및 상기 인증이 성공하는 경우에 상기 반도체 칩의 보안 모드를 활성화하는 단계를 포함하는 방법이 제공된다. 여기서 상기 반도체 칩은 상기 프로세서 코어에 노말 모드에서 동작하는 엘리먼트들을 포함하는 제1 그룹을 연결하는 제1 버스; 및 상기 프로세서 코어에 보안 모드에서 동작하는 엘리먼트들을 포함하는 제2 그룹을 연결하는 제2 버스를 포함하는 것일 수 있다. 이 경우, 상기 방법은 상기 인증이 성공하는 경우에 상기 제2 그룹 및 상기 제2 버스를 활성화 하고, 상기 인증이 실패하는 경우에 상기 제2 그룹 및 상기 제2 버스를 비활성화 하고 상기 제1 그룹 및 상기 제1 버스를 활성화 할 수 있다. 일실시예에 따르면 상기 제2 그룹은 Cryptographic IP, secure SRAM, secure DMA 및 boot ROM 중 적어도 하나를 포함한다.

[18]

발명의 효과

- [19] 본 발명에 따르면, 하드웨어 디바이스를 인증할 수 있다.

[20]

도면의 간단한 설명

- [21] 도 1은 일실시예에 따른 SoC 장치의 블록도이다.
- [22] 도 2는 일실시예에 따른 SoC를 구현한 구조를 도시한다.
- [23] 도 3a 내지 3f는 일실시예에 따라 Pre-authentication이 수행되는 과정과 프로토콜을 설명하기 위한 도면이다.
- [24] 도 4는 일실시예에 따르면 보안 부팅(secure booting)의 흐름을 설명하기 위한 흐름도이다.
- [25] 도 5는 일실시예에 따른 PUF의 구현을 설명하기 위한 개요도이다.
- [26] 도 6은 일실시예에 따라 물리적 공격으로부터 근본적으로 보호되는 보안 SoC 플랫폼을 도시한다.
- [27] 도 7 내지 도 8은 일실시예에 따라 시큐어 부팅을 수행하는 흐름도이다.

[28]

발명의 실시를 위한 형태

- [29] 이하에서, 실시예들을 첨부된 도면을 참조하여 상세하게 설명한다. 그러나, 권리범위는 이러한 실시예들에 의해 제한되거나 한정되는 것은 아니다. 각 도면에 제시된 동일한 참조 부호는 동일한 부재를 나타낸다.
- [30] 아래 설명에서 사용되는 용어는, 연관되는 기술 분야에서 일반적이고 보편적인 것으로 선택되었으나, 기술의 발달 및/또는 변화, 관례, 기술자의 선호 등에 따라 다른 용어가 있을 수 있다. 따라서, 아래 설명에서 사용되는 용어는 기술적 사상을 한정하는 것으로 이해되어서는 안 되며, 실시예들을 설명하기 위한 예시적 용어로 이해되어야 한다.

[31] 또한 특정한 경우는 출원인이 임의로 선정한 용어도 있으며, 이 경우 해당되는 설명 부분에서 상세한 그 의미를 기재할 것이다. 따라서 아래 설명에서 사용되는 용어는 단순한 용어의 명칭이 아닌 그 용어가 가지는 의미와 명세서 전반에 걸친 내용을 토대로 이해되어야 한다.

[32]

[33] 하드웨어와 소프트웨어의 상호 인증

[34] 위에서 언급한 바와 같이 하드웨어 기반의 보안 기술 중 secure booting과 같은 절차로 하드웨어가 소프트웨어의 무결성을 검증하는 기술이 보편적이다. 만약 스마트폰이나 핸드헬드 디바이스의 OS가 무결한 이미지가 아니라 손상되거나 해킹 등으로 보안 기능이 변조된 것이 발견되는 경우, 하드웨어가 이를 부팅 단계에서 발견하거나, 및/또는 특정 어플리케이션의 시행 단계에서 발견하여 보안 위험을 막는 것 등이다. 해커에 의해 변조된 OS가 설치된 스마트폰에서 모바일 뱅킹이나 페이먼트 어플리케이션을 실행할 때 해킹된 OS가 아닌지 검증하여 절차 진행이 중지되는 것과 같은 일은 이제 흔하다.

[35] 그런데 반대로 OS나 어플리케이션 소프트웨어가 하드웨어를 인증하는 것은 제시된 바 없다. 요즈음과 같이 스마트폰의 모바일 OS부터, 전자 장비의 펌웨어, 임베디드 소프트웨어 및 각종 어플리케이션이 온-에어 배포(distribution on air)되는 경우, 개발자 또는 매뉴팩처 입장에서 이들이 설치되어 운용될 장비가 정품이고 변조/손상되지 않은 무결한 것임을 보장받는 것은 충분히 필요한 기술이다. 이하에서 제시되는 여러 실시예들에서는, 불법 모조 하드웨어, 계약에 의한 사용 기간이 만료된 하드웨어, 원래의 기기에서는 보안에 의해 막혀 있는 기능이나 정보에 접근하기 위해 변조된 하드웨어 등을 방지하기 위해 소프트웨어 스스로 하드웨어를 인증하는 기술이 제시된다. 물론 하드웨어가 소프트웨어를 인증하는 내용을 함께 포함하므로, 이러한 면에서는 하드웨어와 소프트웨어의 상호 인증을 통한 안전성이 보장되는 시스템으로 이해되어도 좋다.

[36] 이하의 실시예에서는 시스템 온 칩 레벨의 장치에서 하드웨어와 소프트웨어의 상호 인증 기술이 예시적으로 설명되나, 반드시 이러한 예에 국한되는 것은 아니다. 따라서 핸드 헬드 장비의 하드웨어와 그 OS 또는 펌웨어의 상호 인증, 나아가 자동차나 항공기와 그 운영 소프트웨어의 상호 인증 등 발명의 사상을 벗어나지 않는 범위에서의 유사한 응용은 얼마든지 있을 수 있다.

[37]

[38] 시스템 온 칩의 소프트웨어에 대한 보안 공격

[39] SoC에 대한 공격은 물리적 공격과 소프트웨어 공격을 포함한다. 물리적 공격이 무력화되고 소프트웨어 시스템이 보안 SoC 플랫폼에 의해 보호된다면, 개발자들은 보안 이슈들에 대하여 신경 쓸 필요 없이 자신의 일에만 집중할 수 있다. 여기서 제시되는 실시예들은 소프트웨어에 대한 보안 공격을 방지하여 안전한 환경에서 운영되는 SoC 플랫폼을 제공한다.

- [40] 소프트웨어 공격은 소프트웨어 모듈만으로는 방어하기가 어려운 경우가 있어서, 하드웨어 모듈 차원에서 소프트웨어 보안 공격을 방어하는 것이 바람직하다. 예시적으로 ARM 사의 TrustZone과 같은 환경이 될 수 있다. TrustZone을 이용하기 위해서는 TEE(Trusted Execution Environment) 표준에 의해 정의된 보안 OS(Operation System)나 보안 라이브러리가 필요하다. 그러나 TEE의 성능 요구사항은 작은 CPU 코어와 실시간 OS(Real Time OS: RTOS)가 사용되는 IoT(Internet of Things)의 말단 SoC들이 충족하기 어려운 수준일 수 있다. 실시예들에 따르면 큰 오버헤드를 야기하는 secure OS나 보안 라이브러리 대신, 직접 TEE를 지원할 수 있는 보안 ISA(Instruction Set Architecture)를 갖춘 SoC가 제시된다. 그리고 이러한 보안 ISA를 수행할 수 있는 CPU 코어 및 그 동작 방법의 실시예들이 제시된다. 도면을 참고하여 다양한 실시예들을 설명하기로 한다.
- [41]
- [42] 시스템의 구성
- [43] 도 1은 일실시예에 따른 반도체 시스템 온 칩(100)의 블록도이다. 칩은 프로세서 코어(101)를 포함한다. 코어(101)에는 제1 버스(111)와 제2 버스(121)이 별도로 연결된다. 제1 버스(111)는 일반 모드(normal mode)에서 동작하는 범용 컴퓨팅 그룹인 제1 그룹(112)에 제공된다. '그룹'은 개별 컴퓨팅 자산(intellectual property: IP)나 메모리, 캐시 등의 집합으로 이해될 수 있다. 이하에서 별다른 언급이 없더라도 엘리먼트, IP, 기능 요소, 컴퓨팅 자산 등은 혼용되어 사용될 수 있다.
- [44] 제2 버스(121)은 보안 모드(secure mode)에서만 동작하는 엘리먼트들을 포함하는 제2 그룹(122)에 제공된다. 제2 버스(121)는 제1 버스(111)와는 물리적으로 완전히 분리되어 있다. 이러한 물리적 고립이 normal world(110)와 secure world(120)의 구분을 만들 수 있으며, 이 구분이 소프트웨어에 대한 보안 공격에 강인한 SoC를 제공한다. 물리적인 고립은 normal world(110)의 제1 그룹(112)이 secure world(120)의 제2 그룹(122)과는 완전히 다른 메모리 어드레스 체계를 가지고 데이터를 처리한다는 것으로도 이해될 수 있다.
- [45] SoC(100)에는 시스템 파워-온 시 및/또는 필요에 의해 보안 모드가 실행될 필요가 있을 때 secure world(120)에 대해 외부 인증 기관(미도시)과 상호 인증을 수행하는 인증부(102)가 포함될 수 있다. 인증부(102)는 외부 인증 기관에 대해 이 SoC(100), 제2 그룹(122) 및 그 개별 IP들, 나아가 이에 임베드된 소프트웨어가 정당한 것이며 무결한 것임을 입증하여 인증을 받을 수 있다. 또한 인증부(102)는 현재 트랜잭션의 대상이 되는 외부 터미널이 신뢰할 수 있는 상기 외부 인증 기관이 맞다는 점을 상호 인증하기도 한다. 인증부(102)가 Pre-authentication 외부 인증 기관으로부터 공개키를 받아 등록하고 인증서를 발급받는 발급 프로토콜과 필요에 따라 기존의 공개키를 새 공개키로 교체하는 등 소프트웨어의 하드웨어 인증에 대해서는 도 3을 참조하여 상세히 후술한다.

[46]

[47] 일반 모드와 보안 모드의 전환

[48] 일실시예에 따른 SoC(100)는 일반 모드(normal mode)와 보안 모드(secure mode)를 가질 수 있다. 상술한 인증부(102)의 인증이 성공하는 경우에만 제2 그룹(122)이 부팅되고 신뢰 수행환경(Trusted Execution Environment, TEE)에 연관되는 secure world(120)가 활성화된다. 물론 이 경우에는 제1 그룹(112)도 부팅되고 normal world(110)가 활성화 된다. 그러나 상기 인증이 안 되는 경우에는, 일반 수행 환경(Rich Execution Environment, REE)에 연관되는 제1 그룹(112)만 부팅되어 normal world(110)는 활성화되지만 secure world(120)가 활성화되지 않는다.

[49] secure world(120)가 활성화된 경우라도 제2 그룹(122)이 항상 동작하는 것은 아닐 수 있으며 대기상태에 있을 수 있다. REE 환경의 컴퓨팅, 이를테면 안드로이드 운영체제와 어플리케이션이 실행되는 중에, 보안 위반 이벤트(event of secure violation)가 발생하거나 또는 금융 결제가 필요한 등의 이유로 상승된 보안 취급이 처리되어야 하는 호출이 있는 경우 상기 보안 모드가 수행될 수 있다. 다시 말해, Normal world에서는 모바일 폰의 안드로이드와 같은 REE(Rich Execution Environment)가 일반적인 하드웨어 IP들과 함께 ARM 기반의 SoC에서 동작한다. 그러다가 보안 자원, 이를테면 Cryptographic IP나 secure SRAM이 필요할 때에는 보안 OS와 보안 하드웨어 IP로 이루어진 TEE가 동작하여 REE에게 결과를 전송한다. ARM 사의 TrustZone은 SoC 플랫폼의 보안 IP에 접근하기 위한 제어 신호로, AMBA AXI에 1 비트 버스 신호를 추가로 제공한다. 그러나 보통의 작은 CPU 코어와 RTOS(Real Time OS)가 사용되는 IoT(Internet of Things) 말단 SoC에서 TEE에서의 성능 요구사항들을 충족시키기는 어려울 수 있는데 실시예들에 따르면 이것이 가능하다.

[50]

[51] 인증의 수단

[52] 일실시예에 따르면 반도체 SoC(100)는 인증부(102)가 상기 외부 신뢰 기관과 상호 인증을 수행하는 데에 이용되는 근원 키(root key)를 제공하는 PUF(Physical unclonable function)(103)를 더 포함할 수 있다. 이러한 근원 키를 이용한 RSA, AES 등의 다양한 인증 알고리즘 중 적어도 하나가 상기 인증에 사용될 수 있다. 예시적으로, 그러나 한정되지 않게, PUF는 반도체 제조 공정상의 공정 편차를 이용하여 상기 반도체 칩 내에 내재적으로(intrinsic) 포함될 수 있다. 하나의 예시적인 구현에서, 상기 PUF는 반도체의 전도성 레이어 간에 레이아웃 된 비아(via)나 인터-레이어 콘택이 공정에서 정상적으로 패터닝 되어 상기 전도성 레이어 사이를 단락하는지에 따라 디지털 값을 제공하는 것일 수 있다. PUF의 구현과 역할에 대해서는 도 5와 6을 참조하여 보다 상세히 후술한다.

[53]

[54] 시큐어 메모리

- [55] 일실시에에 따르면 시큐어 메모리(130)는 일반 모드에서 활용하는 영역과 보안 모드에서 활용하는 영역을 포함한다. 시큐어 메모리(130)는 예시적으로 비휘발성 메모리(NVM)을 포함한다. 보안 모드에서 활용하는 영역은 데이터를 그대로 저장하지 않고 PUF(103)이 제공하는 근원 키를 이용하여 암호화 해서 저장할 수 있다. 따라서 물리적으로 시큐어 메모리(130)의 데이터가 추출되더라도 PUF(103)과 직접 연결(direct wired)되어 복호화되지 않으면 이는 의미 없다.
- [56]
- [57] 예시적 시스템 구현 - Core A 플랫폼의 이용
- [58] 도 2는 일실시에에 따른 SoC를 Core-A 환경에서 구현한 구조를 도시한다. 예시적으로, 그러나 한정되지 않게, 상기 프로세서 코어는 32 비트 RISC(Reduced Instruction Set Computing) 타입의 임베디드 프로세서인 Core-A 프로세서(201)를 포함한다. 다른 일실시에에 따르면 상기 프로세서 코어는 RISC V ISA를 적용한 CPU core일 수도 있다. 이하에서 '프로세서', '프로세서 코어', '시큐어 코어', '시큐어 프로세서' 등은 경우에 따라 상호간에 바뀌어 쓰일 수 있는 용어들로 이해되어야 한다. 예시적인 구현으로 'Core-A'를 이용한 구현에 대해 주로 설명하지만 RISC V ISA를 적용한 CPU 코어 등 다른 타입의 구현이 얼마든지 가능함은 이 분야의 기술자에게 자명하다.
- [59] 일반 버스(210)에 REE 환경의 제1 그룹 엘리먼트들인 SRAM(211), Peripheral IP들(212), 디버깅 포트(213)가 연결되어 있다. 그리고 보안 버스(220)에 TEE 환경에서 동작해야 하는 Cryptographic IP들(221), boot ROM(222), secure SRAM(223) 및 secure DMA(224)가 연결되어 있다. 도 1을 참조하여 설명한 바와 같이, 인증부(202) 및 적어도 하나의 PUF들(203)이 존재하여 인증을 수행할 수 있다. PUF들(203)은 인증을 위해 코어(201)에 직접 연결되어 있으며, 또한 Cryptographic IP 중 AES/SEED IP, ECC/RSA IP에 직접 연결되어 있을 수 있다. 나아가 시큐어 NVM(230)도 PUF(203)에 직접 연결되어 있어 데이터를 암호화하여 안전하게 저장하는 것이 가능하다.
- [60] 일실시에에 따른 보안 버스(220)는 상기 Core-A 프로세서의 ASR (Application Specific Register)을 상기 secure world의 제2 그룹 엘리먼트들의 제어를 위한 주소 공간으로 사용하여 구현되는 히든 버스일 수 있다. 이 응용에서, 상기 제2 그룹은 상기 보안 모드에서 상기 ASR에 대한 데이터 처리 명령인 MTA(Move to ASR) 명령과 MFA(Move from ASR) 명령어를 보안 명령어(secure instruction) 처리에 이용할 수 있다. 이는 ASR 인터페이스를 이용하여 상기 제2 그룹에 포함되는 보안(secure) IP들을 위한 히든 버스를 구현하는 것으로도 이해될 수 있다.
- [61] 이러한 구현에 따르면, 큰 오버헤드를 갖는 secure OS나 라이브러리 대신 직접 TEE를 지원할 수 있는 보안 ISA(secure Instruction Set Architecture)가 구현된다. 따라서 ARM의 TrustZone을 이용하는 것에 비해 작은 CPU를 갖는 디바이스들에 적합하다. 실시예들에 따르면 일반 모드와 구분되는 보안 모드 관리 및

하드웨어와 소프트웨어 상호 인증을 위해 인증부인 pre-authentication 하드웨어가 제공된다. 또한 신뢰성, 안정성, 랜덤성이 이미 검증된 VIA-PUF(Physical Unclonable Function)을 이용한 상호 인증 하드웨어 IP들이 제공되므로, 강화된 secure world를 갖는 '보안 SoC 플랫폼'이 제공된다. 제시된 실시예에 따르면, 보안 ISA가 제시된다. 일반적인 ISA와 달리 별도로 구현되는 secure world 버전의 ISA이다. 기존의 ARM사 TrustZone 기술에 따른 방법으로 secure world 내 자원을 접근하기 위해서는 보안 OS나 복잡한 라이브러리가 필요하다. 그러나 제안하는 실시예에서는 보안 ISA에 기반하여 secure world 내 자원을 명령어(instruction) 수준에서 바로 접근할 수 있다.

- [62] 또한, Pre-authentication IP를 사용하여 보안 ISA에 대한 보안 모드 관리를 함으로써 안전한 상호 인증을 수행하고 비인가된 사용자의 내부 접근, 이를테면 소프트웨어 공격을 막는다. 실시예에 따라 제안되는 보안 코어는 secure world에 대한 숨겨진 통로를 제공하며, 이는 주 메모리 인터페이스와 물리적으로 분리되어 있다. 또한 소프트웨어를 읽어들이는 때 secure DMA를 이용하여 하드웨어에 의해 자동으로 무결성이 확인되도록 하며, 인증을 위해서 내부 PUF가 보안 저장 공간이나 보안 IP들을 위한 키를 제공한다. 각각을 보다 상세히 설명한다.

[63]

[64] 보안 ISA(Instruction Set Architecture)

- [65] 기존의 방법에 따라 secure world 내에서 자원을 사용하거나 보안 함수를 호출하기 위해서는 보안 OS나 라이브러리가 범용 CPU core에서 동작하기 위해 필요하다. 그러나 실시예에 따르면 secure world에서의 요구사항과 secure OS 및 라이브러리의 특징적 기능을 제공하면서도 오버헤드가 없는 보안 ISA가 제시되며 이는 종전의 Core-A를 기능적으로 확장한다. 이는 보안 시스템이 보안 OS나 라이브러리의 도움 또는 어플리케이션 개발의 어려움 없이 TrustZone을 직접 접근할 수 있도록 해주며 관리를 위한 노력도 줄어 들 수 있다. 실시예에 따른 보안 ISA의 예시적 특징은 다음과 같다.

- [66] - 제어 흐름 무결성: 효율을 위해 하나의 명령어로 제어 흐름의 무결성을 보호하는 코드 대체
- [67] - 키 관리: 공개키 암호를 위한 랜덤 수 및 키 쌍 생성 포함
- [68] - 메모리 보안 레벨 관리: 각 페이지와 세그먼트별로 다양한 보안 레벨 제공
- [69] - 보안 엔진: 보안 하드웨어 엔진과 밀접하게 동작하는 ISA 확장
- [70] - Secure Booting / 인증: 보안 디버깅과 부팅을 위한 인증 지원
- [71] - 보안 모드 관리: 보안 모드를 위한 마이크로구조의 관리 지원
- [72] - 레지스터 내용 보호: VIA-PUF로 보안 레지스터의 내용 암호화

[73]

[74] Pre-authentication 수행

- [75] 도 3a 내지 3f는 일 실시예에 따라 Pre-authentication이 수행되는 과정과

프로토콜을 설명하기 위한 도면이다. 도 3a를 참조하면 하드웨어 SoC(300) 내에는 인증을 처리하고 보안 모드에서 동작하는 부분(310)이 존재한다. 보안 ISA가 사용될 수 있기 위해서는 pre-authentication IP(301)가 전원이 켜졌을 때나 리셋된 이후 말단간 상호인증을 통해 외부 인증서 서버, 이를테면 상업 네트워크에서 터미널의 보안을 관리하는 TSM (Terminal Security Management) System 서버(301)와 통신할 수 있다. 도시된 바와 같이 예시적으로, 그러나 한정되지 않게 이러한 상호 인증은 PUF(303)가 만들어 내는 근원 키를 이용한 공개키 암호화 방식에 의할 수 있다.

- [76] 일실시에에 따르면 상기 상호 인증에 사용되는 pre-authentication IP(301) 내 인증 프로토콜은 일반적인 언어(예를 들어, C 등의 고급 언어나, 특정 Core용 어셈블리어)가 아니다. Pre-authentication IP(301) 내에서는 인증 프로토콜의 수행만을 위해 설계한 전용 언어로 구현되어 있기 때문에 분석이 어렵고 실질적으로 분석이 불가능하다. 또한 pre-authentication IP(301)가 하드웨어 로직으로 구현됨으로써 수정이 불가능(무결성 보장)할 뿐만 아니라 ROM 등의 물리적으로 분리된 IP 형태가 아닌 기본 셀들의 조합으로 구현함으로써 PnR(Place and Route) 과정에서 다른 셀들과 섞이게 하는 것이 가능해 역공학 등의 물리적 공격으로부터도 강인하다.
- [77] 서버(301)와 제안한 플랫폼(300)은 각자의 개인키로 서명을 생성하고 인증서 내에는 자신의 공개키가 포함되어 있다. 상호 인증은 인증서와 서명을 교환함으로써 수행된다. Pre-authentication IP(302)는 VIA-PUF(303)로부터 개인키를 얻고 이를 이용해 생성한 서명을 인증서 서버에게 전달한다. 그리고 저장하고 있던 외부 인증서 서버의 공개키로 수신한 서명을 검증한다. 이 때 서명 알고리즘은 공개키 암호 알고리즘으로 ECC나 RSA일 수 있다. 성공적인 상호인증으로 시스템이 보안 모드에 진입하면 보안 코어가 secure booting을 제어한다. 또한 디버깅 포트는 오직 보안 모드일 때에만 열려, 인증된 사용자만 디버깅을 수행할 수 있게 된다. 인증이 성공하는 경우에는 보안 모드가 허가될 수 있으며, secure world의 IP 자원들(311 및 312 등)이 동작할 수 있다.
- [78] 한편, 상호 인증에서 교환했던 공유 키는 향후 보안 목적으로 데이터를 주고 받을 때 암호화 키나 MAC(Message Authentication Code)용 키, 그리고 암호화 또는 MAC에 사용되는 IV(Initial Vector)로 사용하여 통신 시 기밀성 및 무결성 확보에 사용될 수 있다. 이하에서는 상호 인증을 위한 자세한 프로토콜을 설명한다.

[79]

[80] Pre-authentication 프로토콜

[81] Pre-authentication 프로토콜은 사용에 앞서 공개키를 등록하고 인증서를 발급받는 '발급 프로토콜'과 필요에 따라 기존의 공개키를 새 공개키로 교체하고 새 공개키에 대한 인증서를 재발급할 수 있는 '재발급 및 갱신 프로토콜'을 포함하고 있다. 발급 이후 칩은 TSM (Terminal Security Management) System 에게

사용을 요청하는 것으로 시작하며, 유효 기간이 남은 경우 그대로 사용 프로토콜이, 유효 기간이 만료된 경우 갱신 프로토콜이 온라인상에서 수행된다. 만약 유효 기간이 만료된지 오랜 시간 지났거나 해당 칩의 기존 키 쌍이 공격 등으로 인해 더 이상 유효하지 않다고 TSM에서 판단된 경우 온라인상에서의 사용 프로토콜과 갱신 프로토콜은 사용할 수 없으며, 발급 기관에서 발급 프로토콜과 비슷한 방식으로 재발급을 받아야 한다. 이 때 재발급 프로토콜은 ID 발급 부분만 생략되므로 이를 제외하면 발급 프로토콜과 거의 동일하다. 그리고 안전하다고 가정된 발급기관에서 키와 인증서를 변경할 뿐만 아니라 기존 키는 더 이상 유효하지 않으므로 공개키 전달 시 서명이 필요 없다. 다시 말해 갱신 프로토콜에선 칩에서 새로 생성한 공개키를 TSM에게 전달할 때 변경 전 개인키로 서명하여 전달한다. 각 단계의 프로토콜을 도면을 참조하면서 상세히 설명한다.

[82]

[83] 발급 프로토콜

[84] 도 3b에서 제시된 흐름은 발급 프로토콜을 제시한다. 기기에서 생성한 공개키 쌍 중 공개키를 등록하고 공개키에 대한 인증서를 발급 받는 과정이다. 공개키 쌍 생성 및 전달, 인증서 발급 부분은 재발급 프로토콜에서도 동일하게 수행될 것이다.

[85] 먼저 칩 및 일련번호 목록 전달과정이 수행된다. 유효한 기기들에 대한 일련번호 목록(SN-list)은 제조자로부터 TSM에게로 전달되어 있다(3202).

[86] ㉑ 칩 삽입(3201) : 발급기기에 인증 칩 또는 인증 칩이 삽입된 기기가 삽입된다. 이 때 발급자(예를 들어 은행 또는 신용카드회사)가 구동하는 발급기기는 신뢰되는 것으로 간주한다.

[87] ㉒ SN 요청 : 그러면 발급기기는 칩에게 SN을 요청하여 전달받는다.

[88] ㉓ ID 요청 : 발급기기는 TSM에게 해당 칩에 대한 ID를 요청하면서 칩으로부터 받은 SN을 전달한다. TSM은 ID를 생성(3203)하고, 전달 받은 SN 확인(3204) 후 자신의 공개키(Q_{TSM})를 ID와 함께 발급기기에 전달한다.

[89] * 오류 처리(유효하지 않은 SN으로 판단되는 경우) : TSM에게 유효하지 않은 SN이 전달된 경우라면, 상기 단계에서 ID를 전달하는 대신 오류 메시지(ERR_ID)로 응답하고 폐기(Disuse)하고 종료할 수 있다(3205). 선택적으로 발급기기는 칩에게 유효한 SN을 달라고 다시 요청할 수 있고 ㉒ 과정부터가 반복될 수 있다.

[90] ㉔ 공개키 요청 : 오류가 없다면 발급기기는 칩에게 공개키를 요청하면서 TSM으로부터 받은 ID, TSM의 공개키를 전달한다.

[91] * 오류 처리(CRC 오류) : 발급기기가 TSM으로 받은 메시지에 CRC 오류가 있을 경우 ID 요청(REQ_ID, ㉓)을 다시 하여 ID, TSM의 공개키를 다시 받는다.

[92] ㉕ 공개키 생성 준비 : 칩은 CRC 검사(3206), TSM의 공개키의 유효성 검사(3207)를 수행한다. 그리고 공개키 생성에 사용될 갱신 정보(UpNumECC)를

초기화한다(3208).

- [93] * 오류 처리(CRC 오류) : 칩이 발급기기로부터 받은 메시지에 CRC 오류가 있을 경우 오류 메시지(ERR_PUK)를 송신하여 ID, TSM의 공개키를 다시 받는다.
- [94] * 오류 처리(공개키 오류) : 칩이 TSM으로부터 받은 TSM의 공개키(Q_{TSM})가 유효하지 않을 경우 오류 메시지(ERR_PUK)로 응답하여 TSM의 공개키를 재요청한다. 발급기기는 이를 수신하여 TSM에게 ID를 요청(REQ_ID, ©)을 다시하여 ID, TSM의 공개키를 다시 받는다.
- [95] 상기 초기화(3208) 후 단계(3209)에서 도 3c가 수행된다. 도 3c를 참조하여 설명될 ㉔ 내지 ㉖는 발급과 재발급에 공통되는 부분으로, 발급에 관한 도 3b에서는 단계(3209)로, 재발급에 관한 도 3e에서는 단계(3508)로 지칭된다. 후술할 도 3e의 단계(3508) 과정에서 여기의 ㉔ 내지 ㉖가 다시 참조된다.
- [96] ㉔ 공개키 생성 및 전달(3301) : ID, PUF, 갱신 정보를 바탕으로 공개키 쌍을 생성하여 그 중 공개키는 발급기기으로 전달하고 발급기기는 이를 TSM으로 전달한다. 이 구간은 신뢰되는 것으로 간주하며 통신상의 오류를 배제할 수 있도록 CRC 코드를 덧붙인다.
- [97] * 오류 처리(CRC 오류) : 발급기기가 칩으로부터 받은 메시지에서 CRC 오류 발생 시(3302) 오류 메시지(ERR_PUK)를 송신하여 칩의 공개키를 다시 전송 받는다. 오류 반복 시 칩을 폐기(Disuse)(3303) 시킬 수 있다.
- [98] ㉕ 인증서 발급 : TSM은 CRC 검사(3304), 칩의 공개키의 유효성 검사(3305)를 수행한다. 그리고 수신한 공개키에 자신의 개인키로 서명한 인증서를 생성(3307)하여 발급기기으로 전달하고 발급기기는 이를 칩에게 전달한다. 참고로 도 3b 내지 3d에 표시된 인증서에는 칩의 ID와 공개키, 그리고 이들에 대한 서명만 포함되어 있지만, 필요한 경우 서명하는 대상 및 인증서 내용에 ID 및 칩의 공개키 외에도 발급 기관, 유효 기간 등의 추가 정보가 더 포함될 수 있다.
- [99] * 오류 처리(CRC 오류) : TSM이 발급기기로부터 받은 메시지에서 CRC 오류 발생 시 발급기기는 칩에게 CRC 오류 메시지(ERR_CERT)로 응답하여 재전송을 요청한다.
- [100] * 오류 처리(공개키 오류) : TSM이 발급기기로부터 받은 칩의 공개키(Q_{Chip})가 유효하지 않을 경우 오류 메시지(ERR_CERT)로 응답하여 칩의 공개키를 재요청한다. 발급기기는 이를 받아 오류 메시지(ERR_PUK)를 칩에게 전달한다. 오류 반복 시 칩을 폐기(Disuse)(3306) 시킬 수 있다.
- [101] * 오류 처리(CRC 오류) : 발급기기가 TSM으로부터 받은 메시지에서 CRC 오류 발생 시(3308) 발급기기는 TSM에게 인증서 요청 메시지(REQ_CERT)를 다시 전송한다.
- [102] ㉖ 인증서 검증 : 칩은 CRC를 검사하고(3309) 인증서에 포함된 ID와 공개키가 자신의 것과 동일한지 확인한다. 그리고 ID와 공개키에 대한 인증서의 서명을 확인한다(3310). 서명이 확인되면 갱신 정보(UpNumECC)와 TSM으로부터 받았던 인증서를 비휘발성 메모리에 저장한다(3313).

- [103] * 오류 처리(CRC 오류): 칩이 발급기기로부터 받은 메시지에서 CRC 오류 발생 시(3311) 칩은 발급기기에 CRC 오류 메시지(ERR_REG_CERT)로 응답하여 재전송을 요청한다. 재전송 후 에러처리이면 폐기(3312) 가능하다.
- [104] * 오류 처리(인증서 오류): 인증서 오류(ID, 공개키, 서명 오류) 발생 시 칩은 인증서 오류 메시지(ERR_REG_CERT)를 발급기기에 보내고 발급기기는 TSM에게 보내 인증서를 재요청(REQ_CERT)하고 이 후 과정을 반복한다.
- [105] 이상이 발급과 재발급에 공통되는 도 3c 부분이며, 다시 발급 과정인 도 3b의 ID, Q_{TSM} 를 저장하는 단계(3210) 직전으로 돌아가서 단계(3210)이 수행된다.
- [106] 발급이 끝나면 각 칩은 발급받은 ID와 초기화한 갱신 정보($UpNum_{ECC}$), TSM의 공개키(Q_{TSM}), 그리고 자신의 공개키에 대한 TSM의 인증서(TSM의 개인키로 칩의 공개키를 서명한 것)를 비휘발성 메모리에 가지고 있게 된다. 발급 후 TSM은 발급된 칩들에 대한 SN-ID-공개키 목록을 보유하게 된다. 칩과 발급자(발급기기), TSM, 제조사 간의 연결은 안전하다고 가정한다. 즉 발급 단계에서는 공격자의 개입이 불가능하다고 가정한다.
- [107]
- [108] 갱신 프로토콜
- [109] 발급이 끝난 칩에 전원이 켜져 칩이 TSM에게 사용 요청을 하였을 때 TSM에서 해당 ID의 칩의 유효 기간을 확인한 결과 유효 기간이 남아 있지 않은 경우 칩 사용을 불허하며, 칩에게 공개키 갱신을 요청한다. 안전하지 않은 통신 환경이기에 새로 생성한 칩의 공개키를 전달할 때 기존의 개인키(갱신 전 개인키)로 서명하여 전달함으로써 해당 칩으로부터 온 새 공개키임을 보장한다. 자세한 프로토콜은 도 3d를 참고하여 설명한다.
- [110] ㉠ 유효 기간 확인 요청: ID와 랜덤 수(RN)을 TSM에게 전달한다(3401).
- [111] ㉡ 유효 기간 확인 및 갱신 요청: TSM이 유효 기간을 확인한다(3402). 유효 기간이 남아 있지 않은 경우 갱신을 요청한다. 공격자가 임의로 기기를 갱신시킬 수 없도록 ID와 수신한 랜덤 수, 그리고 갱신 코드(RENEWAL)를 TSM의 개인키로 서명하여 갱신 요청 메시지에 덧붙인다.
- [112] ㉢ 공개키 쌍 갱신 및 전달: 칩은 수신한 갱신 요청 메시지의 ID와 서명을 확인(3403 및 3404)한 후, 1 증가시킨 $UpNum_{ECC}$ 를 이용하여 새로 공개키쌍(d'_{Chip} , Q'_{Chip})을 생성한다(3405). 그리고 그 중 공개키를 ID와 함께 갱신 전 개인키(d_{Chip})로 서명하여 TSM에게 전달한다(3406).
- [113] * 오류 처리(ID 오류): 칩이 TSM으로부터 받은 ID가 자신의 것이 아니라면 오류 메시지(ERR_RENEWAL)로 응답한다. TSM은 ID를 확인하고 맞다면 갱신 요청 메시지(REQ_RENEWAL)부터 반복한다.
- [114] * 오류 처리(서명 오류): 칩이 TSM으로부터 받은 서명이 유효하지 않을 경우, 오류 메시지(ERR_RENEWAL)로 응답한다. TSM은 기존 갱신 요청 메시지(REQ_RENEWAL)을 재전송하거나 서명 생성부터 반복한다.
- [115] ㉣ 인증서 발급: TSM은 칩의 공개키의 유효성 검사와 서명의 유효성 검사를

수행한다(3407 내지 3409). 그리고 수신한 공개키에 자신의 개인키로 서명한 인증서를 생성하여 칩에게 전달한다. 상술한 바와 같이 인증서를 재생성할 때에도 공개키 외에 추가 정보가 함께 붙을 수 있다.

[116] * 오류 처리(공개키 오류): TSM이 칩으로부터 받은 칩의 공개키(Q'_{Chip})가 유효하지 않을 경우 오류 메시지(ERR_RENEWAL)로 응답하여 칩의 공개키를 재요청한다. * 오류 처리(서명 오류): TSM이 칩으로부터 받은 메시지의 서명이 유효하지 않을 경우 칩에게 오류 메시지(ERR_RENEWAL)로 응답하여 재전송을 요청한다. 칩은 서명 생성부터 반복한다.

[117] © 인증서 검증: 칩은 인증서에 포함된 ID와 공개키가 자신의 것과 동일한지 확인한다(3410). 그리고 ID와 공개키에 대한 인증서의 서명을 확인한다(3411). 서명이 확인되면 $UpNum_{ECC}$ 와 TSM으로부터 받은 인증서를 비휘발성 메모리에 저장한다(3413).

[118] * 오류 처리(인증서 오류): 인증서 오류(ID, 공개키, 서명 오류) 발생 시 폐기(3412)가 있을 수도 있지만, 칩은 인증서 오류 메시지(ERR_REG_CERT)를 TSM에게 보내 인증서를 재요청하여 과정을 반복할 수도 있다. 갱신이 끝나면 칩은 공개키 갱신에 사용된 1증가된 $UpNum_{ECC}$ 와 새로 갱신한 공개키에 대한 TSM의 인증서를 비휘발성 메모리에 가지고 있게 된다. 그리고 단계(3414)와 단계(3415)를 거쳐 갱신 프로토콜을 마무리된다.

[119]

[120] 재발급 프로토콜

[121] 갱신 시엔 새 공개키를 기존의 개인키로 서명함으로써 새 공개키가 칩의 것임을 보장하였다. 그러나 유효 기간 만료가 한참 지난 경우엔 기존의 개인키를 더 이상 사용할 수 없으므로 발급 기관에서 칩을 다시 재발급 받아야 하며, 자세한 프로토콜은 다음과 같다.

[122] ① 유효 기간 확인 요청: 카드 삽입(3501) 후 칩은 재발급 상태인지 모르므로 갱신과 마찬가지로 ID와 랜덤 수(RN)를 생성(3502)하여 함께 송신하는 명령을 발급기기로 전달한다. 발급기기는 이를 받아 TSM에게 재발급 명령을 요청한다.

[123] ② 재발급 요청: TSM은 갱신 요청과 비슷한 방식으로 ID와 수신한 랜덤 수, 그리고 재발급 코드(ISSUE)를 TSM의 개인키로 서명(3503)하여 재발급 요청 메시지에 덧붙인다.

[124] © 재발급 요청 확인: 칩은 CRC와 ID, 서명을 확인한 후(3504 내지 3506) $UpNum_{ECC}$ 를 1증가시킨다(3507).

[125] * 오류 처리(CRC오류): 칩이 발급기기로 부터 받은 메시지에 CRC 오류가 있을 경우 오류 메시지(ERR_REPUK)를 송신하여 ID, 서명을 다시 받는다.

[126] * 오류 처리(ID 오류): 칩이 TSM으로부터 받은 ID가 자신의 것이 아니라면 오류 메시지(ERR_REPUK)로 응답한다. TSM은 ID를 확인하고 맞다면 재발급 요청 메시지(REQ_REPUK)부터 반복한다.

[127] * 오류 처리(서명 오류): 칩이 TSM으로부터 받은 서명이 유효하지 않을 경우

오류 메시지(ERR_REPUK)로 응답한다. TSM은 기존 갱신 요청 메시지(REQ_REPUK)를 재전송하거나 서명 생성부터 반복한다.

- [128] 그리고 여기서 상술한 도 3c로 설명한 발급 프로토콜의 ㉔ 내지 ㉔가 반복된다.
- [129] ㉔ (발급의 ㉔) 공개키 생성 및 전달(3301) : ID, PUF, 갱신 정보를 바탕으로 공개키 쌍을 생성하여 그 중 공개키는 발급기기로 전달하고 발급기기는 이를 TSM으로 전달한다. 이 구간은 신뢰되는 것으로 간주하며 통신상의 오류를 배제할 수 있도록 CRC 코드를 덧붙인다.
- [130] * 오류 처리(CRC 오류) : 발급기기가 칩으로부터 받은 메시지에서 CRC 오류 발생 시(3302) 오류 메시지(ERR_PUK)를 송신하여 칩의 공개키를 다시 전송 받는다. 오류 반복 시 칩을 폐기(Disuse)(3303) 시킬 수 있다.
- [131] ㉔ (발급의 ㉔) 인증서 발급 : TSM은 CRC 검사(3304), 칩의 공개키의 유효성 검사(3305)를 수행한다. 그리고 수신한 공개키에 자신의 개인키로 서명한 인증서를 생성(3307)하여 발급기기로 전달하고 발급기기는 이를 칩에게 전달한다. 이 경우에도 발급 시와 마찬가지로 필요에 따라 다양한 추가 정보(이들테면, 발급기관, 유효기간)들이 함께 추가될 수 있다.* 오류 처리(CRC 오류) : TSM이 발급기기로부터 받은 메시지에서 CRC 오류 발생 시 발급기기는 칩에게 CRC 오류 메시지(ERR_CERT)로 응답하여 재전송을 요청한다.
- [132] * 오류 처리(공개키 오류) : TSM이 발급기기로부터 받은 칩의 공개키(Q_{Chip})가 유효하지 않을 경우 오류 메시지(ERR_CERT)로 응답하여 칩의 공개키를 재요청한다. 발급기기는 이를 받아 오류 메시지(ERR_PUK)를 칩에게 전달한다. 오류 반복 시 칩을 폐기(Disuse)(3306) 시킬 수 있다.
- [133] * 오류 처리(CRC 오류) : 발급기기가 TSM으로부터 받은 메시지에서 CRC 오류 발생 시(3308) 발급기기는 TSM에게 인증서 요청 메시지(REQ_CERT)를 다시 전송한다.
- [134] ㉔ (발급의 ㉔) 인증서 검증 : 칩은 CRC를 검사하고(3309) 인증서에 포함된 ID와 공개키가 자신의 것과 동일한지 확인한다. 그리고 ID와 공개키에 대한 인증서의 서명을 확인한다(3310). 서명이 확인되면 UpNum_{ECC}와 TSM으로부터 받았던 새 인증서를 비휘발성 메모리에 저장한다(3313).
- [135] * 오류 처리(CRC 오류) : 칩이 발급기기로부터 받은 메시지에서 CRC 오류 발생 시(3311) 칩은 발급기기에 CRC 오류 메시지(ERR_REG_CERT)로 응답하여 재전송을 요청한다. 재전송 후 에러처리이면 폐기(3312) 가능하다.
- [136] * 오류 처리(인증서 오류) : 인증서 오류(ID, 공개키, 서명 오류) 발생 시 칩은 인증서 오류 메시지(ERR_REG_CERT)를 발급기기에 보내고 발급기기는 TSM에게 보내 인증서를 재요청(REQ_CERT)하고 이 후에 단계(3508)에서 도 3c에서 설명한 내용이 수행된다. 이것은 도 3c를 참조하여 설명한 발급 과정의 ㉔ 내지 ㉔이다. 이 부분이 수행된 이후 다시 도 3e의 마무리 단계로 돌아간다. 재발급 결과 1증가한 UpNum_{ECC}와 새로 발급받은 인증서(R', S')가 새롭게 저장된다(3507). 발급 프로토콜과 마찬가지로 발급 단계에서는 공격자의 개입이

불가능하다고 가정한다.

[137]

[138] 사용 프로토콜

[139] 도 3f를 참조하여 설명한다. 사용 프로토콜은 TSM과 상호 인증을 수행하여 core로 하여금 secure mode의 권한을 수여하는 프로토콜이다. 도 3a에서 설명되는 인증 과정을 수행하는 예시적 프로토콜이다.

[140] ① 유효 기간 확인 요청 : 갱신, 재발급 프로토콜과 동일하게 ID와 랜덤 수(RN)를 생성(3601)하여 TSM에게 전달하며 프로토콜이 시작한다.

[141] ② 유효 기간 확인 및 세션 요청 : TSM이 유효 기간을 확인한다(3602). 유효 기간이 남아 있는 경우 칩과 키를 공유하기 위한 ECDH 과정으로 임시 키 쌍(a, A)를 생성한다. 그리고 이를 TSM의 개인키로 서명하여 칩에게 전달한다.

[142] ③ 공유키 생성 : 칩은 수신한 ID를 확인하고(3603), 수신한 공개키(A)의 유효성, 그리고 서명의 유효성을 확인한다(3604 및 3605). 그리고 ECDH 과정으로 칩의 임시 키 쌍(b, B)를 생성한다. 그리고 TSM으로 받은 임시 공개키(A)와 칩이 생성한 임시 개인키(b)를 이용하여 공유 비밀정보(W)를 만들고 이를 키 유도 함수(Key derivation function)로 확장하여 공유 비밀키(K1, K2, IV)를 생성한다(3606). 생성한 공개키와 이에 대해 칩의 개인키로 서명한 서명, 그리고 hand shake를 위해 공유 비밀키로 암호화한 정보(C₁)를 TSM에게 전달한다(3607 내지 3608).

[143] * 오류 처리(ID 오류) : 칩이 TSM으로부터 받은 ID가 자신의 것이 아니라면 오류 메시지(ERR_SESSION)로 응답한다. TSM은 ID를 확인하고 맞다면 세션 요청 메시지(REQ_SESSION)부터 반복한다.

[144] * 오류 처리(공개키 오류) : 칩이 TSM으로부터 받은 공개키가 유효하지 않은 경우 오류 메시지(ERR_SESSION)로 응답한다. TSM은 세션 요청 메시지(REQ_SESSION)부터 반복한다.

[145] * 오류 처리(서명 오류) : 칩이 TSM으로부터 받은 서명이 유효하지 않은 경우 오류 메시지(ERR_SESSION)로 응답한다. TSM은 서명 생성을 다시 하거나 세션 요청 메시지(REQ_SESSION)부터 반복한다.

[146] ④ Hand shake : TSM은 수신한 공개키(B)의 유효성과 서명의 유효성을 확인한다(3609 및 3610). 그리고 전달받은 칩의 공개키(B)와 TSM에서 생성하였던 임시 개인키(a)를 이용하여 공유 비밀정보(W)를 만들고 이를 키 유도 함수(Key derivation function)으로 확장하여 공유 비밀키(K1, K2, IV)를 생성한다(3611). 이를 이용하여 C₁을 복호화함으로써 칩이 동일한 공유 비밀키들을 생성하였음을 확인한다. TSM 또한 공유 비밀키로 암호화한 정보(C₂)를 칩에게 전달한다(3612).

[147] * 오류 처리(공개키 오류) : TSM이 칩으로부터 받은 공개키가 유효하지 않은 경우 오류 메시지(ERR_HS)로 응답한다. 칩은 hand shake 요청 메시지(REQ_HS)부터 반복한다.

- [148] * 오류 처리(서명 오류): TSM이 칩으로부터 받은 서명이 유효하지 않은 경우 오류 메시지(ERR_HS)로 응답한다. 칩은 서명 생성을 다시 하거나 hand shake 요청 메시지(REQ_SESSION)부터 반복한다. 만약 서명 오류가 반복될 경우 hand shake 과정을 멈추고 인증 실패를 위한 오류 메시지(ERR_HS)를 인증 실패 코드(ER_AUTH)와 함께 보내 인증 결과를 실패($F_{Auth} = \text{False}$)로 설정하도록 한다.
- [149] * 오류 처리(Hand shake 오류): TSM이 칩으로부터 받은 C_1 의 확인 결과 실패할 경우 오류 메시지(ERR_HS)로 응답한다. 비밀키 생성이나 C_1 생성부터 다시 수행한다.
- [150] © Hand shake2: 칩은 수신한 C_2 를 복호화하여 확인함으로써 TSM 또한 동일한 공유 비밀키들을 생성하였음을 확인한다(3613).
- [151] * 오류 처리(Hand shake 오류): 칩이 TSM으로부터 받은 C_2 의 확인 결과 실패할 경우 오류 메시지(ERR_HS)로 응답한다. 공유 비밀키 생성이나 C_2 생성부터 다시 수행한다.
- [152] 위 과정에서 서로 주고받은 서명을 통해 서로를 인증할 수 있으며, 공유된 비밀키(K_1, K_2, IV)는 향후 TSM과 통신 시 암호화 키, MAC용 키, initial vector로 각각 활용하게 된다. 인증이 성공적으로 수행된 경우 secure mode에 대한 권한을 부여한다($F_{Auth} = \text{True}$)(3614).
- [153] 이상이 Pre-authentication의 수행, 발급, 갱신, 재발급, 사용 각각의 구체적 프로토콜이다. 이러한 과정을 통해 Pre-authentication 인증부가 장치와 TSM에 이르는 하드웨어를 인증하게 된다.
- [154]
- [155] 하드웨어의 소프트웨어 인증
- [156] 기존의 TrustZone 등의 TEE(Trusted Execution Environment)에서 플랫폼의 무결성을 보장하기 위하여 secure booting 기능을 포함하고 있다. 이는 OS 이미지 자체가 오염된 상태로 시스템이 시작하는 것을 방지하기 위한 것으로, 부팅 시 OS 이미지의 무결성을 확인 후 실행함으로써 플랫폼이 무결한 상태에서 시작할 수 있도록 하는 기술이다.
- [157] 도 4는 일실시예에 따르면 이러한 보안 부팅(secure booting)의 흐름을 설명하기 위한 흐름도이다. 디바이스가 파워-온(410)되면, ROM의 SoC 부트로더(420)가 가장 먼저 실행되어 권한을 가진다. bootloader는 on-SoC ROM에 저장되어 있어 수정이 불가능하므로 신뢰 근원(Root of trust)으로서 secure booting 절차의 근간이 된다. 또한 on-SoC 하드웨어는 TSM 인증기관의 공개키를 수정이 불가능한 OTP 형태 등으로 가지고 있다. 부트로더(420)는 이 공개키를 이용하여 플래시 디바이스 부트로더(flash device bootloader)(430)의 서명을 검증함으로써 무결함을 확인하게 되고, 그 후 플래시 디바이스 부트로더(430)에게 권한을 주어 실행될 수 있게 한다. 권한을 넘겨받은 플래시 디바이스 부트로더(430)는 secure world OS의 이미지의 서명을 검증함으로써 무결함을 확인하게 되고 이를 실행(시큐어 부팅)시키게 된다(440). 이러한 과정을 뒤에서도 반복하면서 normal

world 부트로더(450)가 권한을 받아 normal world OS를 부팅하고(460) 시스템이 구동된다. 수정이 불가능하여 무결성이 보장된 하드웨어 차원의 신뢰 근원으로부터 스텝 바이 스텝으로 소프트웨어 인증을 수행하면서 앞으로 나아가는 이러한 과정을 신뢰 체인이라고 한다.

- [158] 이러한 방식으로 하드웨어는 하드웨어에서 수행될 소프트웨어를 신뢰 체인으로 인증하고, 도 3a 내지 3f를 참고하여 설명한 것처럼 소프트웨어도 하드웨어를 인증하여 상호 인증이 수행되는 것이다.

[159]

[160] PUF를 이용한 보안의 강화

- [161] 실시예에 따른 PUF(Physical Unclonable Function)는 반도체 제조과정에서 발생하는 공정편차를 이용하여 칩 고유의 true random bit을 생성할 수 있다. 반도체 제조과정에서 발생하는 공정편차는 칩 마다 상이하므로, 칩의 대량 생산 시 PUF는 칩의 고유 ID로 활용될 수 있으며, 이를 이용하면 암호/복호화, 메모리 보호, 전자서명 등에 활용되는 근원 키(root key)가 제공될 수 있다.
- [162] 이러한 PUF의 값은 칩 외부로 유출되기 어려워 신뢰-근원(Root-of-Trust)으로서 최상위 보안등급을 제공할 수 있다. 현재 대륙간탄도미사일(ICBM) 등 군사용 무기, 차량용 보안시스템(HSM), 스마트폰의 AP, 스마트카드 등의 많은 응용분야에서 PUF를 보안에 이용하기 위한 표준화 및 상용화가 진행되고 있다.

- [163] 도 5는 일실시예에 따른 PUF의 구현을 설명하기 위한 개요도이다. 예시적인 구현으로서, 이에 한정되는 것은 아님에 유의해야 한다. 일실시예에 따르면 PUF는 통상적인 디자인 룰(도시된 예에서는 250nm*250nm)에서 정한 via hole size보다 작은 via hole을 디자인하여 제조된 비아(또는 인터-레이어 콘택) 어레이를 갖는다. 이 경우 via hole 어레이에서 open과 short가 random하게 분포되며, 이는 무작위적(random)이고 실질적으로 시간 불변인 (invariant over time) 디지털 값을 제공할 수 있다. 제시된 실시예에 따라 제조된 VIA PUF는 공인평가기관에서 JEDEC 표준에 따른 다양한 테스트를 통과하여 Reliability가 검증되었고, 랜덤성도 NIST 표준인 SP 800-90B의 true random number testing suite를 통과하였다.

[164]

[165] 추가적인 실시예의 설명

- [166] 도 6은 일실시예에 따라 물리적 공격으로부터 근본적으로 보호되는 보안 SoC 플랫폼을 도시한다.
- [167] 보안 코어(601)에 연결되어 REE 환경의 컴퓨팅을 수행하는 일반 버스(610) 외에 secure world를 위한 히든 버스(620)가 제공된다. SRAM(611)이나 Peripheral IP들(612)가 일반 모드에서 동작하는 중 보안 모드 실행이 필요한 경우에는 보안 ISA를 통해 명령어 수준에서 보안 IP들(621)이 동작할 수 있다.
- [168] 일실시예에 따르면 보안 코어(601)는 메모리 영역을 코드 영역과 데이터 영역으로 분할한다. 분할되는 메모리 영역은 normal world의 것일 수도 있으나,

다른 실시예에서는 secure world의 것일 수도 있다. 보안 코어(601)는 상기 코드 영역에는 쓰기(write)를 금지하고, 데이터 영역에서는 실행(execution)을 금지시키는 메모리 보호 유닛(Memory Protection Unit, MPU)을 포함할 수 있다. 여기서 상기 코드 영역과 상기 데이터 영역은 상기 Core-A 프로세서의 ASR에 대한 명령어를 이용하여 처리되는 보안 명령어에 의해 설정될 수 있다. 나아가, 상기 코드 영역과 상기 데이터 영역 외에 시스템 영역도 설정될 수 있다.

[169]

[170] [표1]

영역\모드	U/N	U/S	P/N	P/S
코드 영역	RX	RX	RWX	RWX
데이터 영역	RW	RW	RW	RW
시스템 영역	-	-	RWX	RWX

[171]

[172] 위 표에서 모드 "U"는 User mode를 의미하고, "P"는 Privilege mode를 의미한다. 그리고 "N"은 Normal mode를, "S"는 Secure mode를 의미한다. 코드 영역 별로 허용되는 동작이 제시되어 있다. 이를테면, 유저 모드이고 일반 모드일 때는 R인 읽기, 그리고 X인 실행(eXecution)은 가능하나, 쓰기 W는 허용되지 않는다.

[173] 한편, 일실시예에 따르면 상기 프로세서 코어는 코드 시퀀스에 포함되는 리턴 어드레스를 백업하는 하드웨어 기반 새도우 스택을 포함할 수 있다. 이 경우, 상기 새도우 스택은 기존의 메모리 상에 상주하는 스택과는 별도로 동일 또는 별도의 추가된 메모리, 또는 별도의 추가된 레지스터 등에 위치할 수 있다. 상기 새도우 스택은 버스를 사용하지 않고, OS나 소프트웨어에 의해 접근되지 않으며, 추가된 하드웨어 기반 제어 회로에 의해 메모리 스택 관련 명령 수행 시 자동으로 수행되어 상기 리턴 어드레스를 이중으로 관리할 수 있다. 새도우 스택을 이용한 리턴 어드레스의 이중 관리로, 소프트웨어 공격으로 인해 기존의 메모리 상에 상주하는 스택에 저장되어 있는 리턴 어드레스가 변경되는 것을 감지할 수 있다. 새도우 스택은 하드웨어 회로로만 제어되고 소프트웨어로 접근 불가능하므로 새도우 스택에 저장된 내용을 변경하려는 공격도 원천적으로 차단할 수 있다. 위에서 설명한 새도우 스택은 리턴 어드레스를 이중으로 관리하는 예시적 구성이며, 리턴 어드레스를 주기적으로 백업해 이중으로 관리하는 스택이면 특정한 예시적 구현에 한정되는 것은 아니다.

[174]

[175] 시스템 부팅 시 동작

[176] 도 7 내지 도 8은 일실시예에 따른 SoC의 동작을 설명하기 위한 흐름도이다. 일실시예에 따르면 디바이스 파워-온 시 인증서 서버로부터 인증을 받는 절차(710)가 수행된다. 여기서 하드웨어 기반 인증부인 pre-authentication IP가

말단간 상호인증을 통해 외부 인증서 서버와 통신하는 내용은 도 3a를 통해 상술한 바와 같고, 이 과정에서 PUF가 이용되는 것은 도 5를 참조하여 설명하였다.

[177] 단계(720)에서 인증에 성공한 경우라면, Secure ISA가 사용 가능하도록 관리되고(730), 디버깅 포트 접근이 가능해지고 secure world OS가 부팅된다(740). 그리고 이어서 노말 모드 부팅을 통해 시스템은 구동 가능하게 된다(750).

[178] 그러나, 단계(720)에서 인증에 실패하는 경우라면, 도 8의 단계(810)에서 secure world는 접근 불가능하게 된다. 그리고 노말 월드만 부팅되어 REE 환경의 컴퓨팅 구동만 가능해질 수 있다. 이러한 과정에 대해서는 도 1, 도 2 및 도 4 등을 참조하여 상술한 바와 같다.

[179]

[180] 이상에서 설명된 장치는 메모리의 하드웨어 구성요소, 메모리를 제어 소프트웨어 구성요소, 및/또는 하드웨어 구성요소 및 소프트웨어 구성요소의 조합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 장치 및 구성요소는, 예를 들어, 프로세서, 컨트롤러, ALU(arithmetic logic unit), 디지털 신호 프로세서(digital signal processor), 마이크로컴퓨터, FPA(field programmable array), PLU(programmable logic unit), 마이크로프로세서, 또는 명령(instruction)을 실행하고 응답할 수 있는 다른 어떠한 장치와 같이, 하나 이상의 범용 컴퓨터 또는 특수 목적 컴퓨터를 이용하여 구현될 수 있다.

[181] 소프트웨어는 컴퓨터 프로그램(computer program), 코드(code), 명령(instruction), 또는 이들 중 하나 이상의 조합을 포함할 수 있으며, 원하는 대로 동작하도록 처리 장치를 구성하거나 독립적으로 또는 결합적으로(collectively) 처리 장치를 명령할 수 있다. 소프트웨어 및/또는 데이터는, 처리 장치에 의하여 해석되거나 처리 장치에 명령 또는 데이터를 제공하기 위하여, 어떤 유형의 기계, 구성요소(component), 물리적 장치, 가상 장치(virtual equipment), 컴퓨터 저장 매체 또는 장치, 또는 전송되는 신호 파(signal wave)에 영구적으로, 또는 일시적으로 구체화(embody)될 수 있다. 소프트웨어는 네트워크로 연결된 컴퓨터 시스템 상에 분산되어서, 분산된 방법으로 저장되거나 실행될 수도 있다. 소프트웨어 및 데이터는 하나 이상의 컴퓨터 판독 가능 기록 매체에 저장될 수 있다.

[182] 실시예에 따른 메모리 동작 제어 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시예를 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기

매체(magnetic media), CD-ROM, DVD와 같은 광기록 매체(optical media), 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다. 상기된 하드웨어 장치는 실시예의 동작을 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.

- [183] 실시예들이 비록 한정된 도면에 의해 설명되었으나, 해당 기술분야에서 통상의 지식을 가진 자라면 상기의 기재로부터 다양한 수정 및 변형이 가능하다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다. 그러므로, 다른 구현들, 다른 실시예들 및 특허청구범위와 균등한 것들도 후술하는 특허청구범위의 범위에 속한다.

[184]

[185]

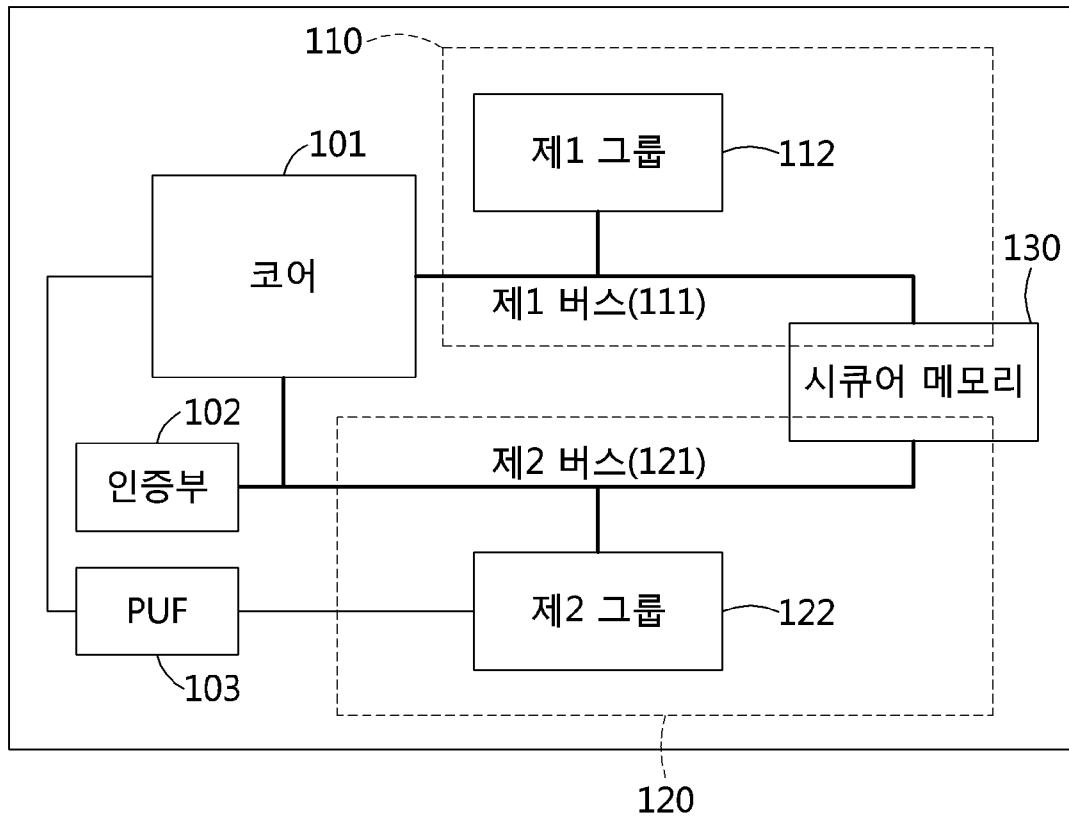
청구범위

- [청구항 1] 반도체 칩에 있어서,
프로세서 코어; 및
외부 신뢰 기관과 상호 인증을 수행하여 상기 반도체 칩 및 상기 외부
신뢰 기관 사이의 인증을 수행하는 인증부
를 포함하는 반도체 칩.
- [청구항 2] 제1항에 있어서,
상기 프로세서 코어에 제1 버스를 통해 연결되는 엘리먼트들을 포함하는
제1 그룹; 및
상기 프로세서 코어에 제2 버스를 통해 연결되며 보안 모드에서 동작하는
엘리먼트들을 포함하는 제2 그룹
을 더 포함하는 반도체 칩.
- [청구항 3] 제2항에 있어서,
상기 인증부가 상기 인증에 성공한 경우에만 상기 제2 그룹을 사용
가능하도록 활성화 하는 반도체 칩.
- [청구항 4] 제2항에 있어서,
상기 제2 그룹은 Cryptographic IP, secure SRAM, secure DMA 및 boot ROM
중 적어도 하나를 포함하는 반도체 칩.
- [청구항 5] 제1항에 있어서,
상기 인증부가 상기 외부 신뢰 기관과 상기 상호 인증을 수행하는 데에
이용되는 키를 제공하는 PUF를 더 포함하는 반도체 칩.
- [청구항 6] 제2항에 있어서,
상기 제2 그룹은 상기 제1 그룹과 상이한 메모리 어드레스를 이용하여
데이터를 처리하는 반도체 칩.
- [청구항 7] 제2항에 있어서,
상기 프로세서 코어는 32 비트 RISC(Reduced Instruction Set Computing)
타입의 임베디드 프로세서인 Core-A 프로세서를 포함하는 반도체 칩.
- [청구항 8] 제7항에 있어서,
상기 제2 버스는 상기 Core-A 프로세서의 ASR (Application Specific
Register)을 상기 제2 그룹에 포함되는 엘리먼트를 제어하는 주소
공간으로 사용하여 구현되는 히든 버스를 포함하는 반도체 칩.
- [청구항 9] 제8항에 있어서,
상기 제2 그룹은 상기 보안 모드에서 상기 ASR에 대한 데이터 처리
명령인 MTA(Move to ASR) 명령과 MFA(Move from ASR) 명령어를 보안
명령어(secure instruction) 처리에 이용하는 반도체 칩.
- [청구항 10] 제1항에 있어서,
상기 인증부는 하드웨어 로직이며, 기존의 컴퓨팅 언어와 상이한 독자

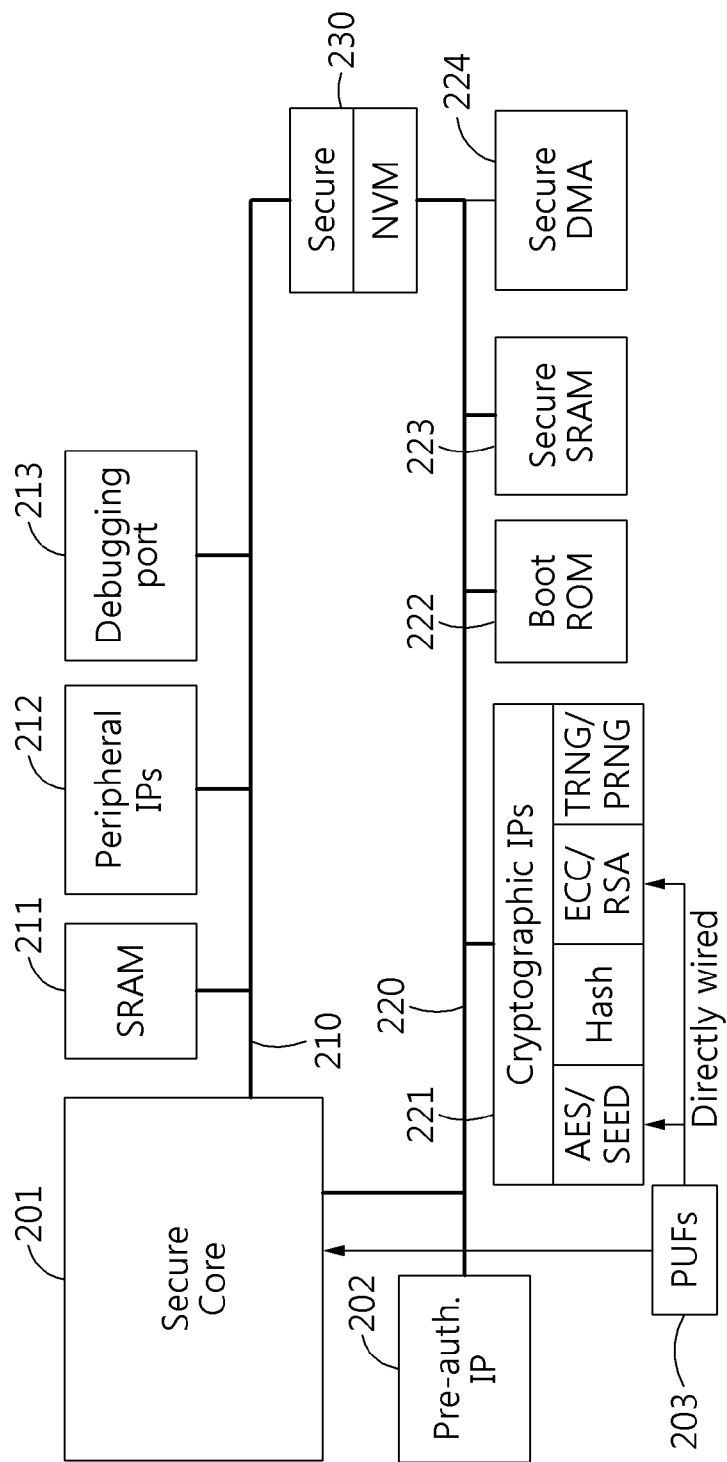
- 언어로 구동되어 상기 인증 절차를 수행하는 반도체 칩.
- [청구항 11] 제10항에 있어서,
상기 인증부는 기존의 컴퓨팅 언어와 상이한 독자 언어로 구성되는 인증 절차를 수행하는 반도체 칩.
- [청구항 12] 제11항에 있어서,
상기 인증부는 ROM이나 상기 반도체 칩 내의 IP와 다르게 일반 셀로 구성되는 반도체 칩.
- [청구항 13] 소프트웨어를 임베드한 하드웨어 장치에 있어서,
상기 소프트웨어를 구동하는 프로세서 코어; 및
상기 하드웨어 장치를 통해 상기 소프트웨어를 구동하기 위해 외부 신뢰 기관과 상호 인증을 수행하여 상기 반도체 칩 및 상기 외부 신뢰 기관 사이의 인증을 수행하는 인증부를 포함하는 장치.
- [청구항 14] 제13항에 있어서,
상기 반도체 칩은 상기 프로세서 코어에 노말 모드에서 동작하는 엘리먼트들을 포함하는 제1 그룹을 연결하는 제1 버스; 및
상기 프로세서 코어에 보안 모드에서 동작하는 엘리먼트들을 포함하는 제2 그룹을 연결하는 제2 버스를 더 포함하는 장치.
- [청구항 15] 제14항에 있어서,
상기 인증부가 상기 인증에 성공한 경우에만 상기 제2 그룹을 사용 가능하도록 제2 버스로의 접근을 활성화 하는 장치.
- [청구항 16] 제13항에 있어서,
상기 인증부가 상기 외부 신뢰 기관과 상기 상호 인증을 수행하는 데에 이용되는 키를 제공하는 PUF를 더 포함하는 장치.
- [청구항 17] 반도체 칩의 동작 방법에 있어서,
인증부가 외부 신뢰 기관과 상호 인증을 수행하는 단계; 및
상기 인증이 성공하는 경우에 상기 반도체 칩의 보안 모드를 활성화하는 단계를 포함하는 방법.
- [청구항 18] 제17항에 있어서,
상기 반도체 칩은 상기 프로세서 코어에 노말 모드에서 동작하는 엘리먼트들을 포함하는 제1 그룹을 연결하는 제1 버스; 및 상기 프로세서 코어에 보안 모드에서 동작하는 엘리먼트들을 포함하는 제2 그룹을 연결하는 제2 버스를 더 포함하고,
상기 방법은 상기 인증이 성공하는 경우에 상기 제2 그룹 및 상기 제2 버스를 활성화 하고, 상기 인증이 실패하는 경우에 상기 제2 그룹 및 상기 제2 버스를 비활성화 하고 상기 제1 그룹 및 상기 제1 버스를 활성화 하는

- 방법.
- [청구항 19] 제18항에 있어서,
상기 제2 그룹은 Cryptographic IP, secure SRAM, secure DMA 및 boot ROM
중 적어도 하나를 포함하는 방법.

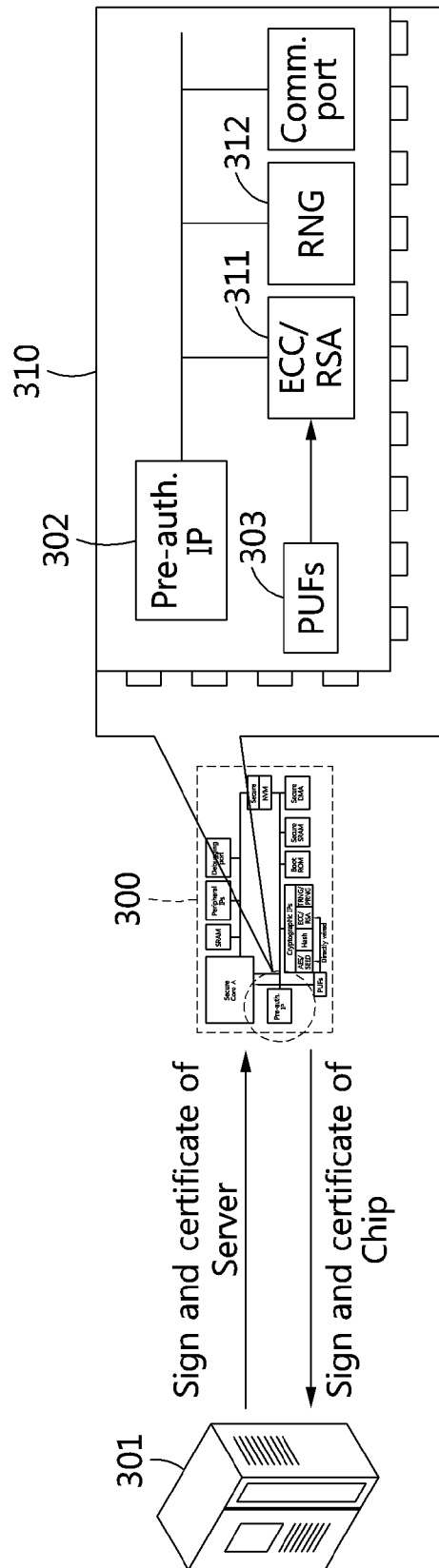
[도1]

100

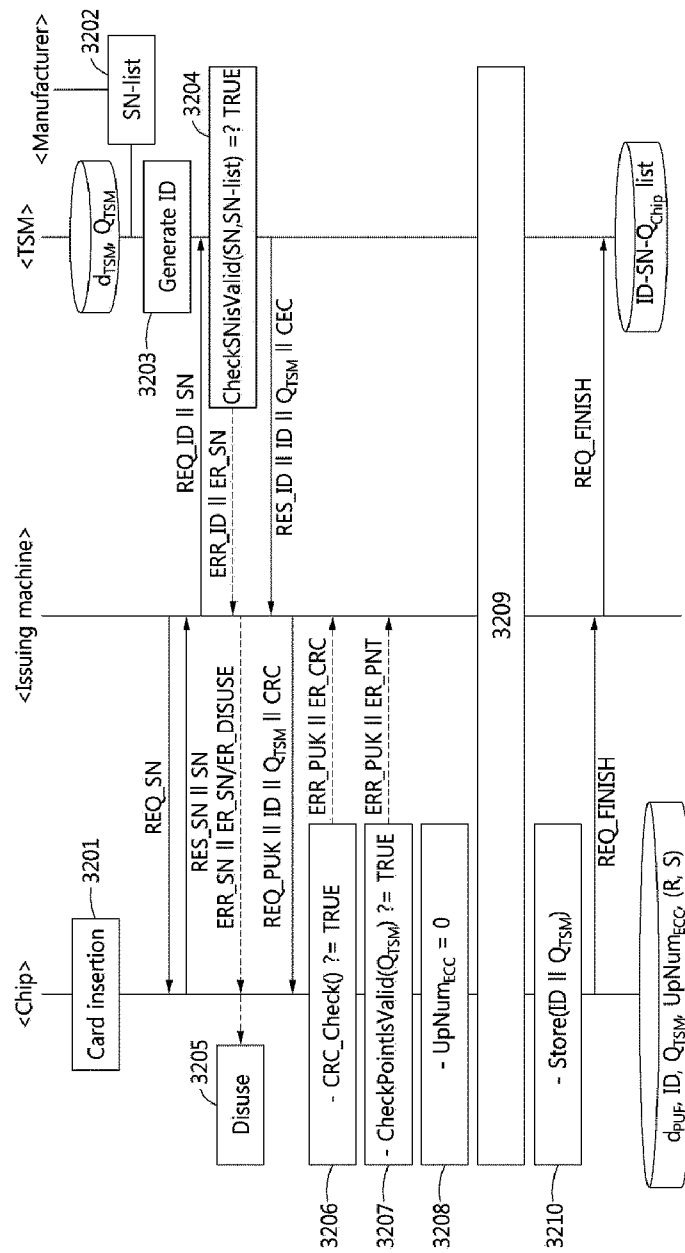
[도2]



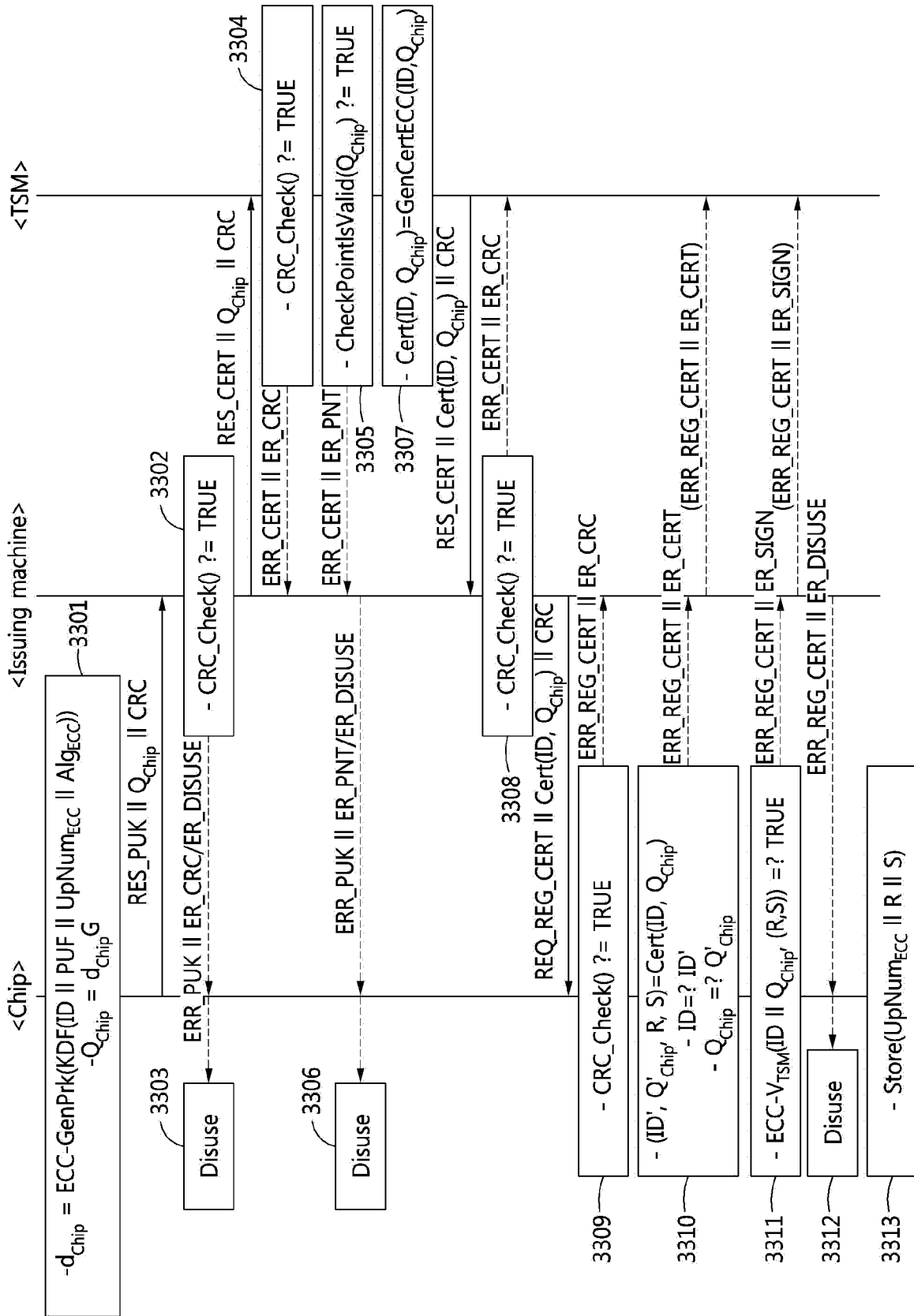
[도3a]



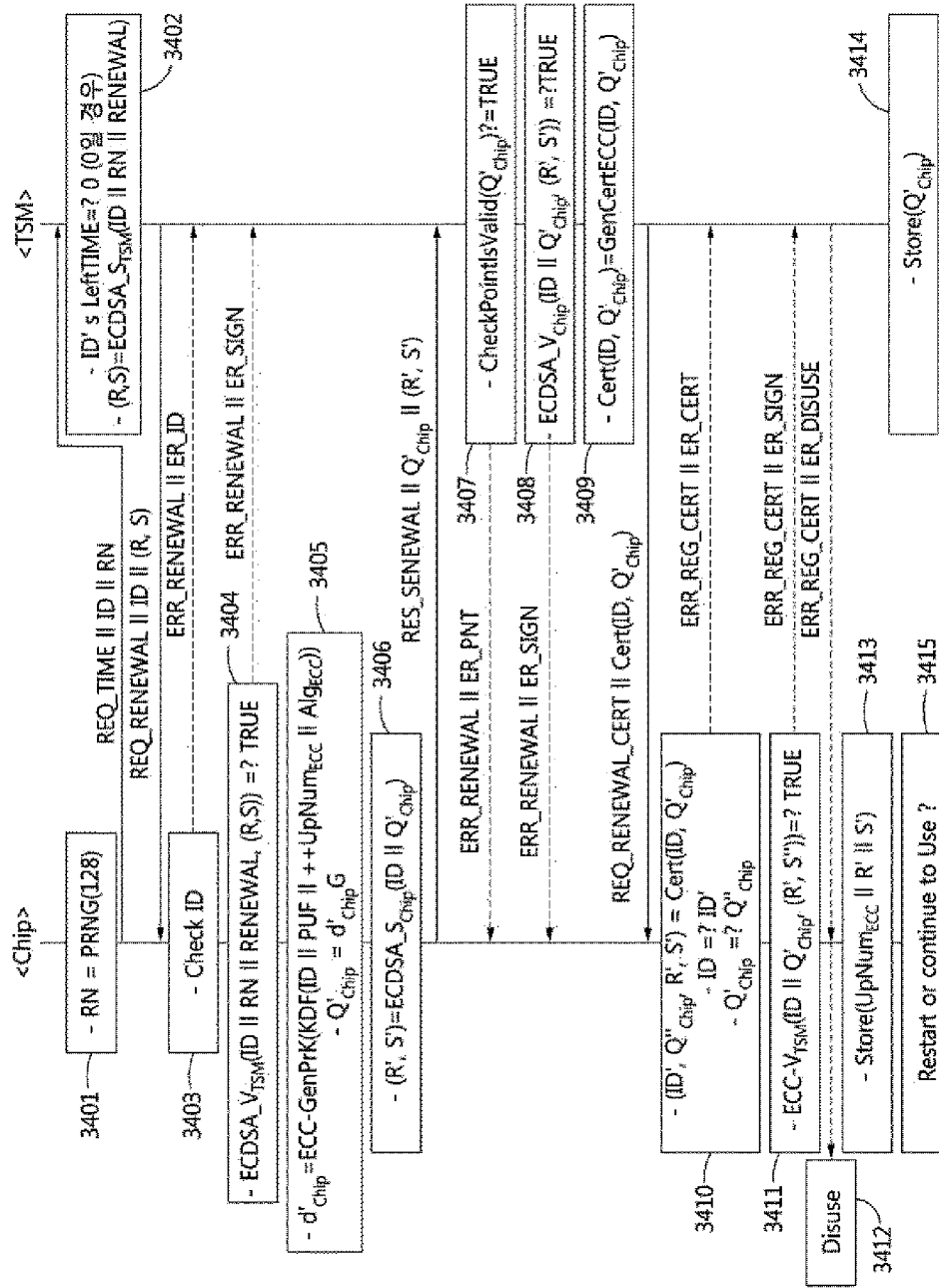
[도3b]



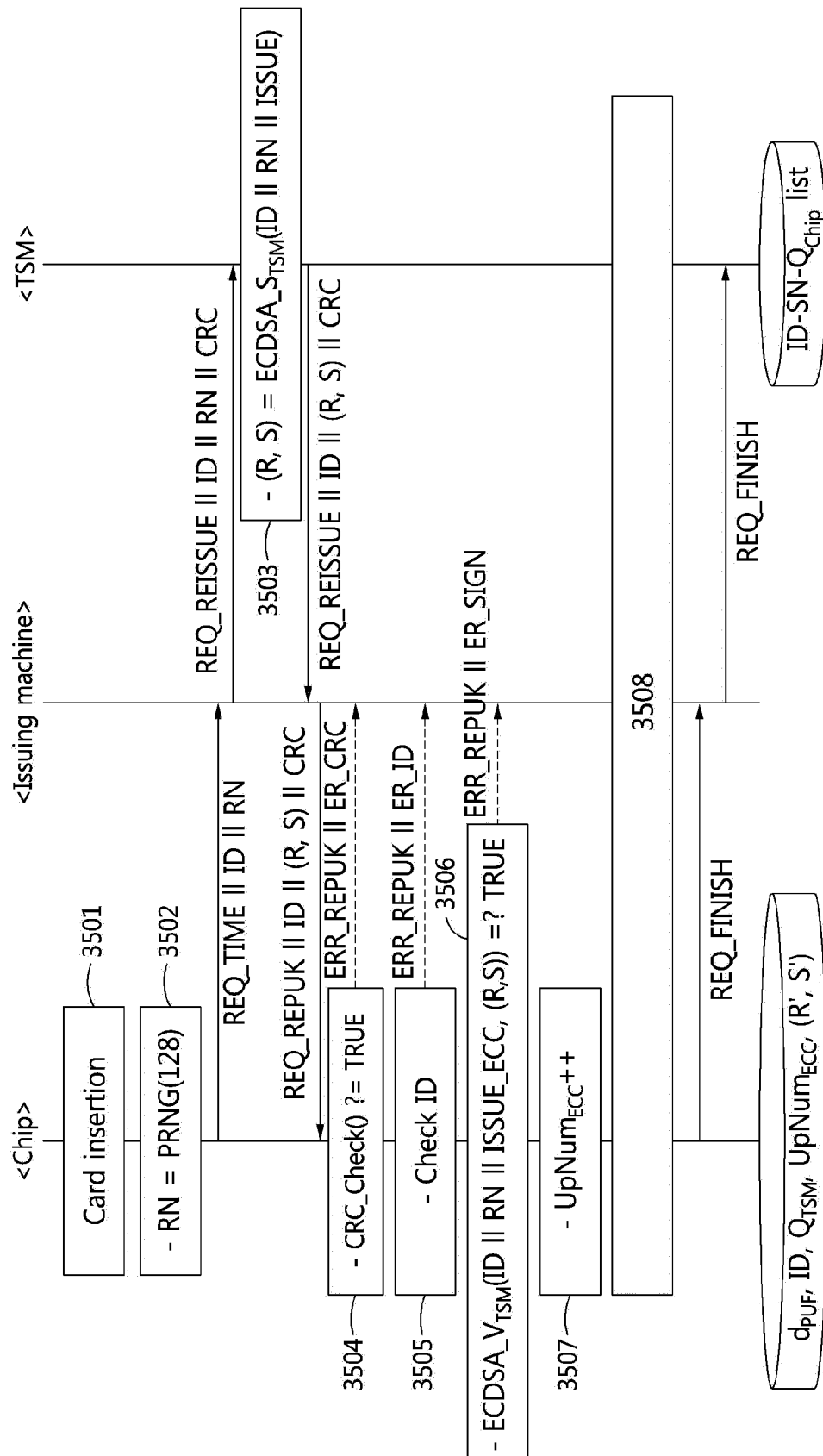
[도3c]



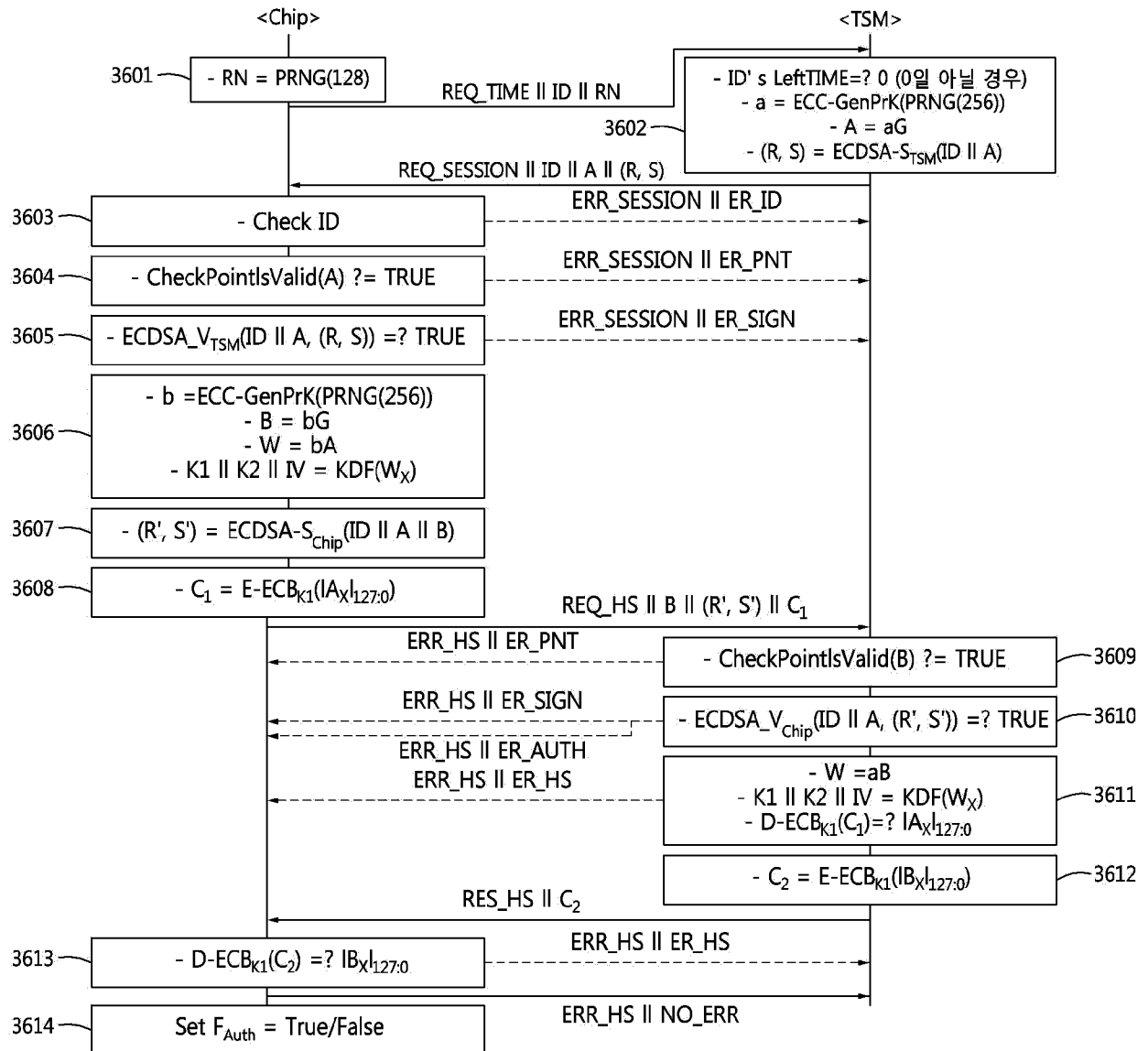
[도3d]



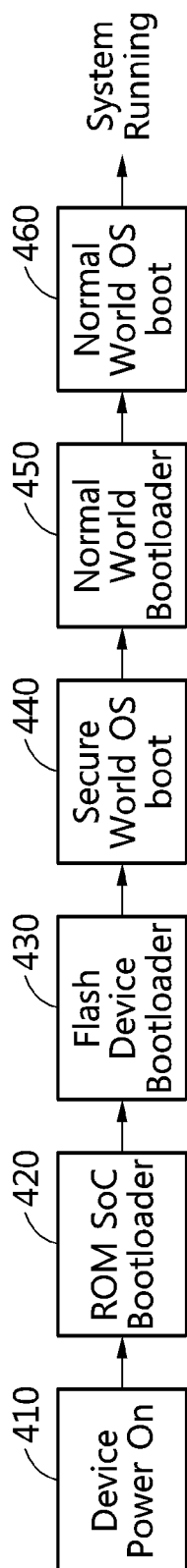
[도3e]



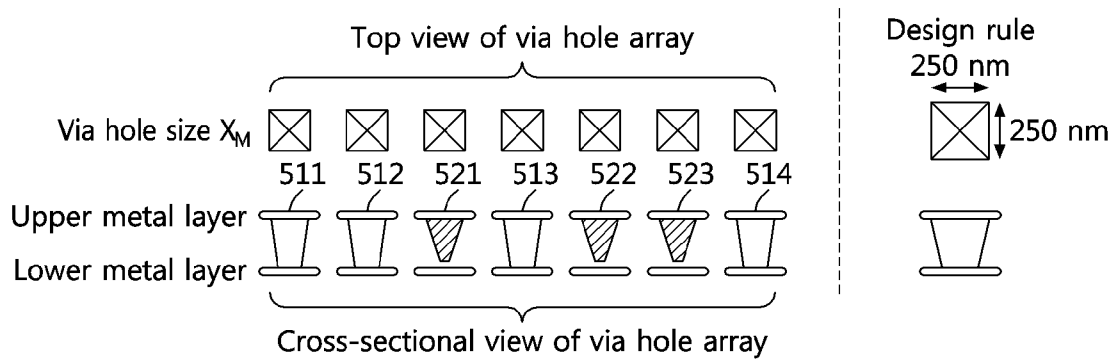
[도3f]



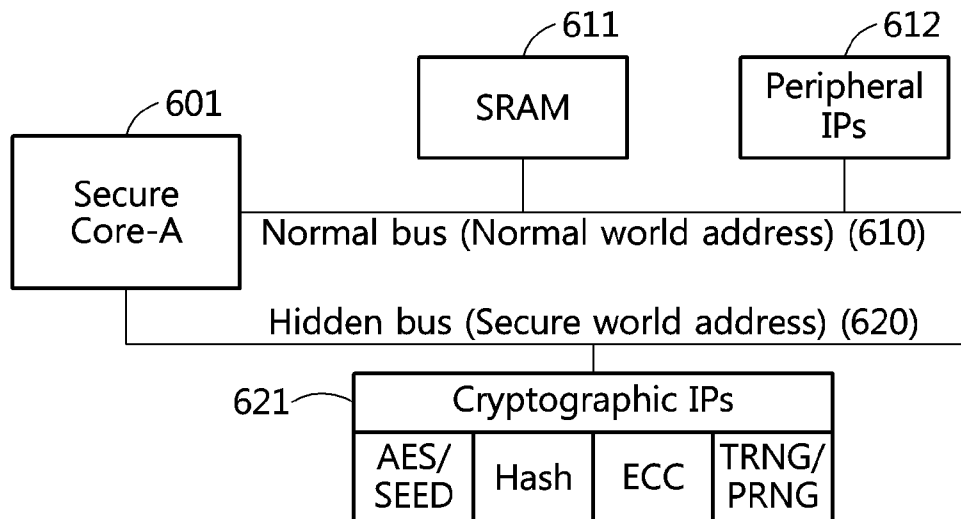
[도4]



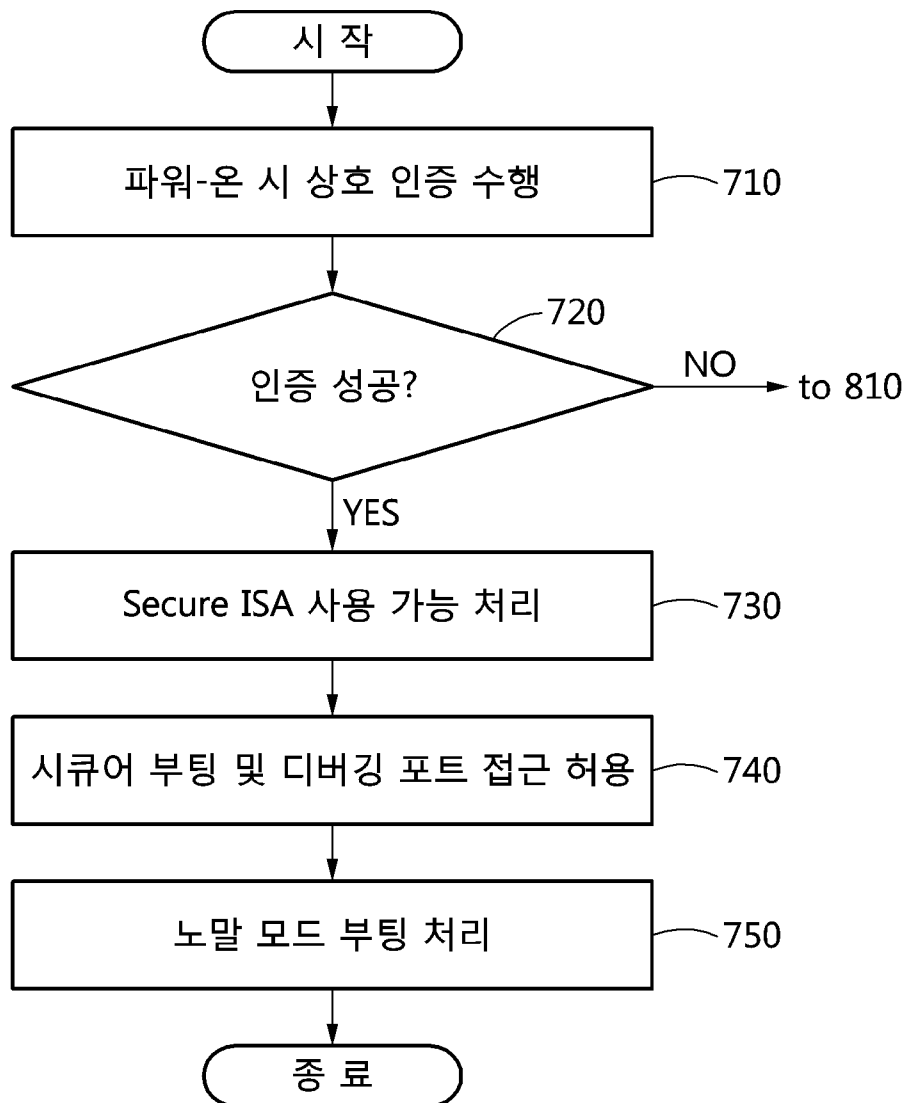
[도5]



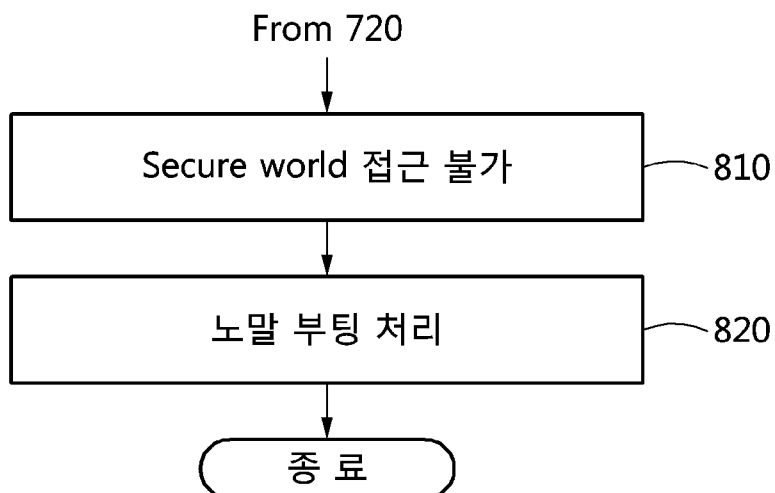
[도6]



[도7]



[도8]



INTERNATIONAL SEARCH REPORT

International application No.

PCT/KR2017/001560**A. CLASSIFICATION OF SUBJECT MATTER*****G06F 21/57(2013.01)i, G06F 21/55(2013.01)i, G06F 15/78(2006.01)i***

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F 21/57; H04L 9/14; G06F 21/44; G06F 17/00; G06F 21/78; G06F 13/38; G06F 1/26; H04L 9/32; G06F 12/14; G06F 21/55; G06F 15/78

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Korean Utility models and applications for Utility models: IPC as above

Japanese Utility models and applications for Utility models: IPC as above

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

eKOMPASS (KIPO internal) & Keywords: semiconductor chip, process core, first bus, second bus, secure mode, PUF, authentication

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2006-0015947 A1 (CONTI, Gregory Remy Philippe et al.) 19 January 2006 See paragraphs [0020]-[0021], [0026]; claim 1, and figures 1-2.	1-19
Y	KR 10-2013-0019358 A (ICTK CO., LTD.) 26 February 2013 See paragraphs [0044]-[0072]; claims 1, 11; and figures 1-2.	1-19
A	KR 10-1118826 B1 (IUCF-HYU (INDUSTRY-UNIVERSITY COOPERATION FOUNDATION HANYANG UNIVERSITY)) 20 April 2012 See paragraphs [0033]-[0043]; and figure 1.	1-19
A	KR 10-2014-0114263 A (SAMSUNG ELECTRONICS CO., LTD.) 26 September 2014 See paragraphs [0036]-[0046]; and figure 2.	1-19
A	KR 10-1477080 B1 (ARM LIMITED) 29 December 2014 See paragraphs [0035]-[0053]; and figures 1-3.	1-19



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

“A” document defining the general state of the art which is not considered to be of particular relevance

“E” earlier application or patent but published on or after the international filing date

“L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

“O” document referring to an oral disclosure, use, exhibition or other means

“P” document published prior to the international filing date but later than the priority date claimed

“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

“&” document member of the same patent family

Date of the actual completion of the international search

22 MAY 2017 (22.05.2017)

Date of mailing of the international search report

23 MAY 2017 (23.05.2017)

Name and mailing address of the ISA/KR

Korean Intellectual Property Office
Government Complex-Daejeon, 189 Sconsa-ro, Daejeon 302-701,
Republic of Korea

Facsimile No. +82-42-481-8578

Authorized officer

Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/KR2017/001560

Patent document cited in search report	Publication date	Patent family member	Publication date
US 2006-0015947 A1	19/01/2006	EP 1612684 A1	04/01/2006
		EP 1612684 B1	30/09/2009
		US 9063889 B2	23/06/2015
KR 10-2013-0019358 A	26/02/2013	CN 103748831 A	23/04/2014
		EP 2747335 A2	25/06/2014
		EP 2747335 B1	11/01/2017
		JP 2014-528195 A	23/10/2014
		KR 10-2013-0129334 A	28/11/2013
		US 2014-0310515 A1	16/10/2014
		WO 2013-025060 A2	21/02/2013
		WO 2013-025060 A3	11/04/2013
KR 10-1118826 B1	20/04/2012	CN 103621006 A	05/03/2014
		CN 103621006 B	07/09/2016
		EP 2677452 A1	25/12/2013
		JP 2014-506095 A	06/03/2014
		JP 2016-021772 A	04/02/2016
		US 2013-0322624 A1	05/12/2013
		US 2015-0195085 A1	09/07/2015
		US 2016-0301528 A1	13/10/2016
		US 9014371 B2	21/04/2015
		US 9397826 B2	19/07/2016
		WO 2012-111872 A1	23/08/2012
KR 10-2014-0114263 A	26/09/2014	US 2014-0283006 A1	18/09/2014
		US 9280655 B2	08/03/2016
KR 10-1477080 B1	29/12/2014	CN 101517549 A	26/08/2009
		CN 101517549 B	20/06/2012
		EP 2062145 A1	27/05/2009
		EP 2062145 B1	13/10/2010
		GB 2442023 A	26/03/2008
		GB 2442023 B	02/03/2011
		IL 194953 A	29/01/2015
		JP 04872001 B2	08/02/2012
		JP 2010-503909 A	04/02/2010
		KR 10-2009-0065531 A	22/06/2009
		MY 146182 A	13/07/2012
		TW 200817904 A	16/04/2008
		TW 1410797 B	01/10/2013

A. 발명이 속하는 기술분류(국제특허분류(IPC))

G06F 21/57(2013.01)i, G06F 21/55(2013.01)i, G06F 15/78(2006.01)i

B. 조사된 분야

조사된 최소문헌(국제특허분류를 기재)

G06F 21/57; H04L 9/14; G06F 21/44; G06F 17/00; G06F 21/78; G06F 13/38; G06F 1/26; H04L 9/32; G06F 12/14; G06F 21/55; G06F 15/78

조사된 기술분야에 속하는 최소문헌 이외의 문헌

한국등록실용신안공보 및 한국공개실용신안공보: 조사된 최소문헌란에 기재된 IPC
일본등록실용신안공보 및 일본공개실용신안공보: 조사된 최소문헌란에 기재된 IPC

국제조사에 이용된 전산 데이터베이스(데이터베이스의 명칭 및 검색어(해당하는 경우))

eKOMPASS(특허청 내부 검색시스템) & 키워드: 반도체 칩, 프로세서 코어, 제1 버스, 제2 버스, 보안 모드, PUF, 인증

C. 관련 문헌

카테고리*	인용문헌명 및 관련 구절(해당하는 경우)의 기재	관련 청구항
Y	US 2006-0015947 A1 (GREGORY REMY PHILIPPE CONTI et al.) 2006.01.19 단락 [0020]-[0021], [0026]; 청구항 1; 및 도면 1-2 참조.	1-19
Y	KR 10-2013-0019358 A ((주) 아이씨티케이) 2013.02.26 단락 [0044]-[0072]; 청구항 1, 11; 및 도면 1-2 참조.	1-19
A	KR 10-1118826 B1 (한양대학교 산학협력단) 2012.04.20 단락 [0033]-[0043]; 및 도면 1 참조.	1-19
A	KR 10-2014-0114263 A (삼성전자주식회사) 2014.09.26 단락 [0036]-[0046]; 및 도면 2 참조.	1-19
A	KR 10-1477080 B1 (에이알엠 리미티드) 2014.12.29 단락 [0035]-[0053]; 및 도면 1-3 참조.	1-19

☐ 추가 문헌이 C(계속)에 기재되어 있습니다.

☒ 대응특허에 관한 별지를 참조하십시오.

* 인용된 문헌의 특별 카테고리:

“A” 특별히 관련이 없는 것으로 보이는 일반적인 기술수준을 정의한 문헌

“T” 국제출원일 또는 우선일 후에 공개된 문헌으로, 출원과 상충하지 않으며 발명의 기초가 되는 원리나 이론을 이해하기 위해 인용된 문헌

“E” 국제출원일보다 빠른 출원일 또는 우선일을 가지나 국제출원일 이후에 공개된 선출원 또는 특허 문헌

“X” 특별한 관련이 있는 문헌. 해당 문헌 하나만으로 청구된 발명의 신규성 또는 진보성이 없는 것으로 본다.

“L” 우선권 주장에 의문을 제기하는 문헌 또는 다른 인용문헌의 공개일 또는 다른 특별한 이유(이유를 명시)를 밝히기 위하여 인용된 문헌

“Y” 특별한 관련이 있는 문헌. 해당 문헌이 하나 이상의 다른 문헌과 조합하는 경우로 그 조합이 당업자에게 자명한 경우 청구된 발명은 진보성이 없는 것으로 본다.

“O” 구두 개시, 사용, 전시 또는 기타 수단을 언급하고 있는 문헌

“&” 동일한 대응특허문헌에 속하는 문헌

“P” 우선일 이후에 공개되었으나 국제출원일 이전에 공개된 문헌

국제조사의 실제 완료일

2017년 05월 22일 (22.05.2017)

국제조사보고서 발송일

2017년 05월 23일 (23.05.2017)

ISA/KR의 명칭 및 우편주소



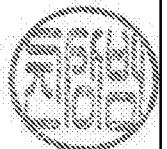
대한민국 특허청
(35208) 대전광역시 서구 청사로 189,
4동 (둔산동, 정부대전청사)

팩스 번호 +82-42-481-8578

심사관

진상범

전화번호 +82-42-481-8398



국제조사보고서에서 인용된 특허문헌	공개일	대응특허문헌	공개일
US 2006-0015947 A1	2006/01/19	EP 1612684 A1 EP 1612684 B1 US 9063889 B2	2006/01/04 2009/09/30 2015/06/23
KR 10-2013-0019358 A	2013/02/26	CN 103748831 A EP 2747335 A2 EP 2747335 B1 JP 2014-528195 A KR 10-2013-0129334 A US 2014-0310515 A1 WO 2013-025060 A2 WO 2013-025060 A3	2014/04/23 2014/06/25 2017/01/11 2014/10/23 2013/11/28 2014/10/16 2013/02/21 2013/04/11
KR 10-1118826 B1	2012/04/20	CN 103621006 A CN 103621006 B EP 2677452 A1 JP 2014-506095 A JP 2016-021772 A US 2013-0322624 A1 US 2015-0195085 A1 US 2016-0301528 A1 US 9014371 B2 US 9397826 B2 WO 2012-111872 A1	2014/03/05 2016/09/07 2013/12/25 2014/03/06 2016/02/04 2013/12/05 2015/07/09 2016/10/13 2015/04/21 2016/07/19 2012/08/23
KR 10-2014-0114263 A	2014/09/26	US 2014-0283006 A1 US 9280655 B2	2014/09/18 2016/03/08
KR 10-1477080 B1	2014/12/29	CN 101517549 A CN 101517549 B EP 2062145 A1 EP 2062145 B1 GB 2442023 A GB 2442023 B IL 194953 A JP 04872001 B2 JP 2010-503909 A KR 10-2009-0065531 A MY 146182 A TW 200817904 A TW I410797 B	2009/08/26 2012/06/20 2009/05/27 2010/10/13 2008/03/26 2011/03/02 2015/01/29 2012/02/08 2010/02/04 2009/06/22 2012/07/13 2008/04/16 2013/10/01