AES 암호화 모듈을 내장한 IC카드 인터페이스 칩셋 개발

(Implementation of IC Card Interface Chipset with AES Cryptography)

김 동 순 † 이 성 철 ††
(Dong-Sun Kim) (Sung-Chul Lee)

요 약 본 논문에서는 각종 전자화폐 및 신용카드를 수용할 수 있도록 WindowsCE 운영체제를 지원하고, 국제적인 표준인 ISO-7816과 호환 가능한 IC카드용 칩의 구현에 관해 기술하였으며, 고성능의 32비트 ARM720T Core와 AES(Advanced Encryption System) 암호 모듈을 내장한 IC카드 칩셋의 구성 방법에 관해 제안하였다. 본 논문에서 제안한 IC카드 칩셋은 T=0, T=1 프로토콜을 지원하는 6개의 ISO 7816 전용 인터페이스포함하고 있으며, 이중 2개는 사용자카드와의 인터페이스를 위해 사용되고 나머지 4개는 SAM 카드와 인터페이스를 위해 사용되도록 설계되었다. 본 논문에서 제안한 IC카드 인터페이스 칩셋은 소프트웨어 기반의 인터페이스 칩셋과 비교해 약 70%의 속도 향상을 얻을 수 있었으며, 하이닉스의 0.35um 공정을 이용해 제작 검증하였다.

키워드 : IC Card, AES, 암호, 전자상거래

Abstract In this paper, we propose the implementation techniques of IC card chipset that is compatible with international standard ISO-7816 and supports WindowsCE operating system to expropriate various electronic cash and credit card. This IC card interface chip set is composed with 32 bit ARM720T Core and AES(Advanced Encryption System) cryptography module for electronic commerce. Six IC card interfaces support T=0, T=1 protocol and two of them are used to interface with user card directly, the others are used for interface with SAM card. In addition, It supports a LCD controller and USB interface for host. We improved the performance about 70% than software based IC card chip set and verified using Hynix 0.35um process.

Key words : IC Card, AES, Cryptography, e-Commerce

1. 서 론

차세대 화폐지불수단으로 대두되기 시작한 전자화폐의 모습이 다양한 방식으로 제안되고 있다. 해외는 물론 국내 업계의 관심이 최근 전자지불시스템에 집중되기 시작하면서 다양한 전자화폐들이 생겨나기 시작하였으며 이를 조정하기 위한 전자지불관련 포럼 기구가 발족하였고 시장에서는 앞 다투어 개발 및 적용을 시도하고 있다. 현재는 각 지방자치단체별로 제한적으로 적용하고 있는 지하철 및 버스카드의 이용이 가장 활발히 이루어지고 있으며 향후 현금카드 및 각종 카드를 지갑에 넣고 다니지 않고 휴대전화에 장착해서 편리하게 갖고 다닐 수 있도록 대기업들을 중심으로 서비스들이 진행될 예정이다. 특히 IC카드를 이용한 전자지불수단은 온라인을 벗어나 오프라인에서 편리하게 사용할 수 있다는 장점으로 인하여 이를 적용하기 위한 다양한 이용환경이 대두되기 시작하였으며 이를 지원할 수 있는 PC형 단말기나 판매시점정보관리(POS) 등의 고급 기술개발에 전문 업체들의 역량이 집중되고 있는 추세다. 이에 따라 그동안 국제표준 규격 인증에 소홀했던 업계도 PC/SC, EMV (Europay Mastercard Visa) 등 해외 인증 자격 취득에 주력하고 있으며, 일부 선도업체들을 중심으로 그 성과도 속속 생겨나고 있고 이동통신제휴카드나 건강보험카드 등 공공·민간 부문에서 시장의 기폭제가 될 만한 움직임이 일어나고 있는 상황이다. IC카드형의 경우 2000년 7월 본덱스 및 K-cash가 각각 시범서비스를 실

† 정 회 원 : 전자부품연구원 DMB 사업단 연구원
dskim@keti.re.kr
†† 비 회 원 : 전자부품연구원 SOC연구센터 연구원
leesc@keti.re.kr
논문접수 : 2003년 1월 16일
심사완료 : 2003년 6월 2일

시함으로써 국내에 도입되기 시작하였으며, 2000년 10월 마이비가 최초로 상용서비스를 시작한 이래로 현재 5개의 전자화폐업체가 영업 중에 있다. 아직까지는 폐쇄형 전자화폐방식을 이용하고 있으나 향후 개방형 전자화폐 지불방식으로 전환될 것으로 본다. 해외의 경우 비자는 2004년말까지 유럽의 88%가 EMV로 전환될 것으로 예상한 가운데 2005년 1월부터는 의무적으로 시행할 예정이며 마스터카드는 라틴아메리카에서 2004년 1월부터 EMV를 의무적으로 시행할 준비를 하고 있다. 시장조사기관인 데이타모니터가 발표한 자료에 의하면 세계 시장규모가 재작년 한해 22억 달러였으나 다가올 2006년에는 80억 달러에 이르러 본격적인 전자지불시대를 맞이할 것으로 보인다. 그동안 IC카드 시장이 활발히 형성되는데 있어 걸림돌이었던 시스템간의 호환성 부족 및 막대한 인프라 구축비용 등의 문제는 CEPS(Common Electronic Purse Specification)등과 같은 개방형 화폐환경의 등장으로 인하여 해결할 수 있게 되었다. IC카드 산업은 핵심부품의 개발보다는 주로 곧바로 적용하기 쉬운 시스템분야에 치우치고 있는 경향으로, IC카드 인터페이스 칩의 경우를 살펴보면, 국내에서도 일부 업체에서 칩 개발에 주력하고 있지만 아직까지 대부분 시스템의 일부분에 귀속되어 IC카드 인터페이스를 제어하기 위한 칩으로써의 기능만을 제공하기 때문에 일부 IC카드 단말기에 채택되어 사용되고 있는 실정이다. 또한 국내 ASIC 설계기술은 IP를 기반으로 한 시스템IC 설계기법이 급속하게 변화하는 시장에 대처하기 위한 신기술로 자리 잡으면서 발전하고 있다. 최근 ARM(Adanced RISC Machine)이나 MIPS® 등과 같은 32비트 RISC Core를 이용한 통신, 가전 등에서의 활용이 본격화되기 시작하였다.

이에 본 논문에서는 각종 전자화폐에 대한 처리능력 및 신용카드기능을 수용하기 위해서 운영체제를 탑재할 수 있도록 32비트 RISC Core를 내장함은 물론, 국제적인 표준이 적용된 IC카드 인터페이스기능 및 최근 대두되고 있는 보안 문제를 해결하기 위한 AES 암호화 모듈을 내장한 IC카드 단말기용 칩 개발 기술에 대해 논하고자 한다. 본 논문의 구성은 2장에서 IC 카드 컨트롤러의 구성 및 AES 모듈의 구성에 대해 설명하며, 3장에서 구현 및 검증 결과에 대해 설명하고자 한다.

2. IC 카드 컨트롤러 및 AES 암호화 모듈의 설계

IC카드 컨트롤러의 구성은 그림1과 같이 ARM720T의 프로세서 부분과 화면 및 메모리 제어 블록, IC카드 인터페이스 블록, 전원 관리 블록과 AES 블록으로 나눌 수 있으며, 본 절에서는 각 블록들에 대해 자세히 기술하고자 한다.

2.1 프로세서

ARM720T를 사용하였으며, MMU(Memory Management Unit), Cache 및 Write Buffer들이 내장되어 RTOS(Real-Time OS)의 탑재가 가능하며 ARM7-TDMI CPU Core에 8K unified cache가 직접 연결되

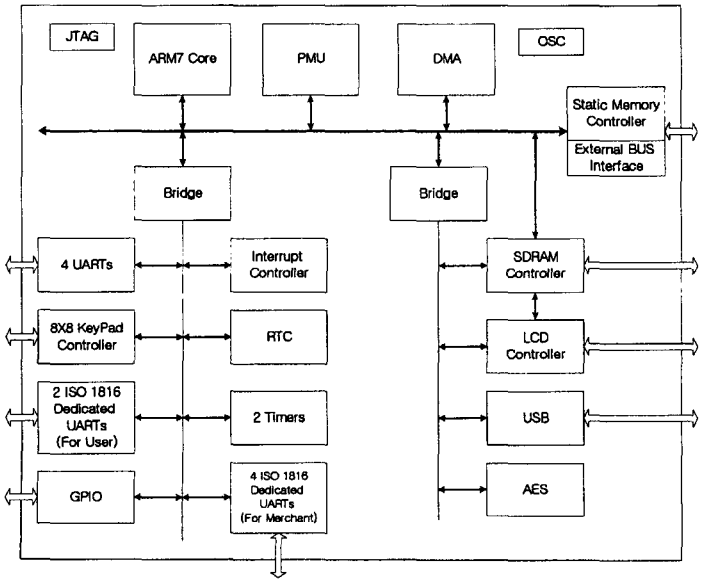


그림 1 IC 카드 칩셀 블록도

어 있는 구조로, Cache타입은 512lines×4words 크기의 4-way set associative이다. 메모리 관리기능을 갖춘 MMU는 가상주소를 실제 사용할 물리적 주소로 변화시키며 64 entry translation loo aside buffer(TLB)를 내장한다.

2.2 화면 제어 블럭

화면 크기가 320×240인 단일 패널형태의 4 또는 8그레이 레벨을 갖는 mono STN LCD를 제어하는 블록으로 SDRAM으로부터 데이타를 입력받고 제어신호는 Advanced Micor-controller Bus Architectures(AMBA)의 규격중 저속 주변기기를 위한 인터페이스 방식인 APB버스로부터 전달받아서 화면을 제어하게 된다.

2.3 메모리 제어 블럭

메모리 제어 블럭은 Dynamic RAM 제어 블럭과 Static RAM 제어 블럭으로 구성된다. Dynamic RAM 제어 블럭은 SDRAM을 제어하는 부분으로 최대 128Mbytes까지 지우너하게 된다. SDRAM은 최대 4개까지 연결가능하다. Static RAM 제어 블럭은 SRAM, FlashROM 및 ROM을 제어한다.

2.4 전원 관리 블럭

전체 칩의 전원손실을 방지하기 위해 내장된 전원 관리 블럭은 보다 적은 전원소모를 위해 컴퓨팅 동작이 없을 경우 정상 상태에서 전원관리모드로 전환된다. 대기모드의 경우 모든 블록의 동작은 정지하게 되며 Real-Time Clock만 유일하게 동작하게 된다. 대기 모드에서는 SDRAM의 데이타의 손실을 막기 위해 self-refresh 모드로 전환되고, 이 모드를 빠져나오기 위해서 RTC wake-up 신호나 UART ring-indicate 입력을 이용하나 본 설계에서는 UART ring-indicate 신호는 포함시키지 않았다. 나머지 정지모드와 슬로우 모드는 프로세서를 정지시키거나 동작속도를 느리게 함으로써 파워소모를 줄인다. 주변 장치 블럭에서 발생하는 인터럽트에 의해서만 이 모드를 벗어날 수 있다.

2.5 IC카드 인터페이스 및 외부통신 블록

6개의 IC카드 인터페이스 블록은 다양한 카드단말기 응용에 사용할 수 있으며 주로 하드웨어 인터럽트를 통하여 카드와의 통신을 수행하도록 되어 있다. 4개의 UART 및 GPIO는 주변기기들과의 인터페이스를 위해 설계되어져 있으며 칩안에 내장된 USB는 IC카드와 관련된 각종 응용프로그램들의 고속 데이타 수신 및 OS의 업그레이드를 용이하게 하기 위해 내장되었다. ISO 7813-3의 비동기 프로토콜인 T=0, T=1을 지원하며, IC Card(ICC)와의 데이타처리는 소프트웨어 처리 없이 전부 하드웨어 인터럽트 발생에 의한 4단계의 카드동작과정을 통해 수행한다. 카드동작과정의 단계는 카드삽입 및 활성화, 정보교환, 카드 비활성화 및 제거의 순서로

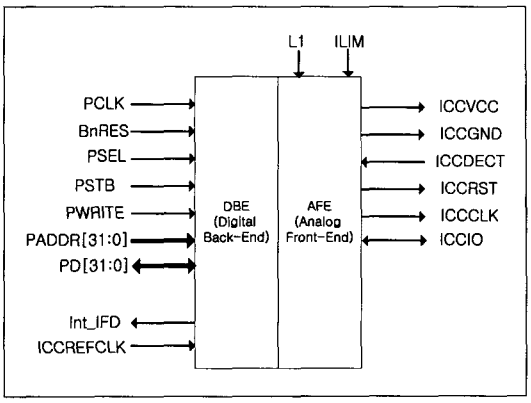


그림 2 ICCIF port configuration

나누어진다. 모든 처리는 인터럽트 상태에 따른 인터럽트 서비스루틴에 의해 이루어진다. IC Card Interface (ICCIF)는 그림 2와 같이 크게 Digital Back-End와 Analog Front로 구성되어 있다.

IC카드의 클럭의 경우 표준안에서와 같이 듀티 사이클은 정상동작동안 주기의 45%에서 55%사이에 있어야 하며, 주파수는 1MHz에서 5MHz 범위에서 동작하고 카드 사용 중에는 ±1%이상 주파수 변동이 있어서는 안된다. IC카드 인터페이스의 동작에 관해 설명하면 다음과 같다.

2.5.1 IC 카드 인터페이스의 초기화

초기화 과정은 reset signal 및 Clock signals에 의해 이루어지며, 내부 레지스터 및 각 블록들의 초기 값으로 초기화 한 뒤 IC 카드의 삽입여부를 대기하게 된다.

2.5.2 IC 카드 삽입 및 인터럽트 생성

ICC가 ICC connector에 삽입되면 ICCDECT 신호를 Low로 세팅함으로써 ICCIF에 카드가 삽입되었다는 것을 알린다. 이 신호에 의해 ICCIF의 상태레지스터(SR)의 삽입확인비트[1]가 1로 셋팅되고 ICCDECTINT인터럽트가 발생한다. 이 인터럽트에 의해 삽입확인타이머(Insertion Check Timer)가 가동된다. ICC가 주어진 삽입확인시간동안 ICC connector에 계속 삽입되어져 있으면 시간 경과후 카드삽입이 성공적으로 이루어졌음을 호스트에게 알려주게 된다. 이때 ICC에는 전원이나 클럭이 인가되지 않으며 ICCIO는 Low로 유지되게 되고, 이 신호들은 활성작업 시퀀스에 의해 동작된다.

2.5.3 Contact 활성 작업 시퀀스 및 Cold 리셋

ARM7으로부터 AMBA버스를 통해 ICCIF에게 ICC와 통신을 하도록 허락하면 정해진 순서에 의해 카드가 활성화된다. 활성작업 시퀀스는 3단계로 나뉘게 되며 ICCIF 활성작업 시간은 ICCIFACTTIME 레지스터에 의해 프로그램된 값으로 결정된다. ICCIF 활성작업 시

퀀스는 Cold 리셋을 포함하며 리셋시간은 ICCIFACTIME 레지스터에 의해 결정되고, 이에 따른 활성작업 시퀀스는 다음과 같다.

1) Asser nICCRST low
2) Wait for ICCIFACTIME OCC clock cycles
3) Enable ICCVCC, configure ICCIO signal as high-Z
4) Wait for ICCIFACTIME ICC clock cycles
5) Enable ICCCLK clock
6) Wait for ICCIFACTIME ICC clock cycles
7) De-assert nICCRST high

제어레지스터의 비트[0](STARTUP)에 1을 세팅함으로써 활성시퀀스를 수행하게 되고, ICCIF는 카드수행 종료 인터럽트 신호인 ICCDONEINTR을 발생시키고 상태레지스터 비트[2](ICCDONE)를 세팅시킨다. ICC로부터의 I/O라인상의 ATR은 리셋이 해지된 후 400에서 40000 ICC Clock 사이클 사이에 발생한다.

2.5.4 ATR

ATR 시퀀스는 삽입된 카드에 대한 정보를 담고 있다. ATR 스트림내의 첫 번째 문자를 TS 문자라 부르며 Conversion(Direct or Invrse format) 정보가 포함되어 있다[2]. ATR 시퀀스가 갖는 configuration value는 clock frequency, Baud rate, Guard time 및 Protocol type이며, ATR 시퀀스의 나머지를 수신하고 Rx FIFO를 통해 선택된 convention으로 읽는다.

2.5.5 Contact 비활성작업 시퀀스

일반적인 카드동작의 마지막 단계가 contact deactivation으로 정해진 순서에 따라 ICC 관련 신호와 전원이 제거된다. contact 비활성작업은 카드에 전기적 충격을 주는 것을 방지하기 위해 다른 모든 동작보다 우선해서 수행된다. 비활성작업 시퀀스는 제어레지스터의 종료비트[2](finish)에 소프트웨어적으로 1을 writing함으로써 시작한다. 비정상적인 비활성동작은 카드 동작중 갑자기 카드제거가 감지될 때 즉 ICCDECT신호가 Low가 됨으로써 contact 비활성작업을 시작하는 경우와 ICCIF의 Analog Front-End에 내장된 과전류보호회로나 저전압 감지센서에 의해 DC/DC 컨버터를 자동 OFF 시킴으로써 비활성작업을 수행하고 종료신호인 DEACTACK를 상태레지스터에 통보하는 경우로 분류된다.

2.5.6 Block Guard Timing

Asynchrous T0,T1 프로토콜에는 Transmit→Receive, Receive→Transmit시에 두 개의 인접한 블록간의 leading edge간의 최소한의 guard time이 정해지는데 이는 Receive block의 마지막 문자와 Transmit block의 첫 번째 전송문자와의 지연 시간이며, T0의 경우 16 재가 ETU, T1의 경우 22 재가 ETU이다[2].

2.5.7 Character Guard Timing

Asynchrous T0,T1 프로토콜에는 Transmit→Receive, Receive→Transmit시에 두 개의 인접한 문자간의 leading edge간 최소한의 guard time이 정해진다. 규격에서 정하는 character guard time은 12ETU+(N×π×1/f)이며, 이때의 N은 CGT값이 되고 N=255이면 T0에서는 12 work ETU이고, T1에서는 11 work ETU가 된다[2].

2.5.8 IC Card Interface Digital Back-End(DBE) 설계

IC 카드 인터페이스 블록은 6 채널 ISO/IEC 7816-3 IC Card Controller로 구성되어 있으며, 데이타 길이는 8bit, 제어 데이타 길이는 32bit으로 이루어져 있다. 컨트롤러 내부에 패러티 비트 생성 및 확인이 가능하며, baud rate generator가 내장되어 있다. 또한 컨트롤러의 정확한 제어를 위해 각 채널당 Transmit buffer empty, Receive buffer full, Receive parity error retry time out, Debounce check complete 및 ATR sequence Not assert interrupt within vaild system clock의 5개 인터럽트 상태를 갖고 있다. 그림 3과 표 1은 각각 내부 블록도와 IC카드 컨트롤러의 레지스터 구조를 보여주고 있다.

IC카드 인터페이스 블록은 크게 9개 블록으로 구성되어 있다. AMBA의 APB 인터페이스 및 내부 레지스터로 구성되어 있는 버스 인터페이스 부와 로직의 제어를 위해 필요한 클럭을 분주하는 클럭 생성부, IC 카드의 삽입 여부와 삽입상태를 확인하는 IC카드 Detection 블록, Activation과 Cold Reset 생성부 및 Deactivation 제어부로 나누어져 있다. 또한 IC카드와 데이타를 주고받는 UART부와 UART제어부도 함께 설계되어 있으며, 설계된 IC카드 인터페이스 블록은 Modelsim을 통해 pre-simulation을 수행하였으며, 그림 4는 IC카드의 h활성화 및 비활성화 과정에 관한 시뮬레이션 결과 화면을 보여주고 있다.

2.6 AES 암호화 모듈

표준안으로 채택된 Rijindael 알고리즘에 기반을 둔 AES 암호화 모듈을 설계하였으며, 128, 192 및 256 bit의 차등 키 길이를 지원하도록 되어 있다. AES 암호화 알고리즘은 블록 암호화 방법의 일종으로 128, 192, 및 256 bits의 블록 사이즈를 지원하게 된다[3]. 암호화 및 복호화는 블록 및 키 길이에 따라 10, 12 및 14번의 라운드 형식에 따라 진행되며, 각각의 라운드는 키 제어부를 통해 서로 다른 키 값을 가지게 된다. AES 암호화 모듈은 그림 5와 같이 크게 키 가산부, Shift Row, Mix Column 및 Substitution 블록으로 나누어지게 된다. 키 가산부는 라운드 키 값과 암호화 데이타간의 바이트 단위 XOR 연산이며, Shift Row와 Substitution

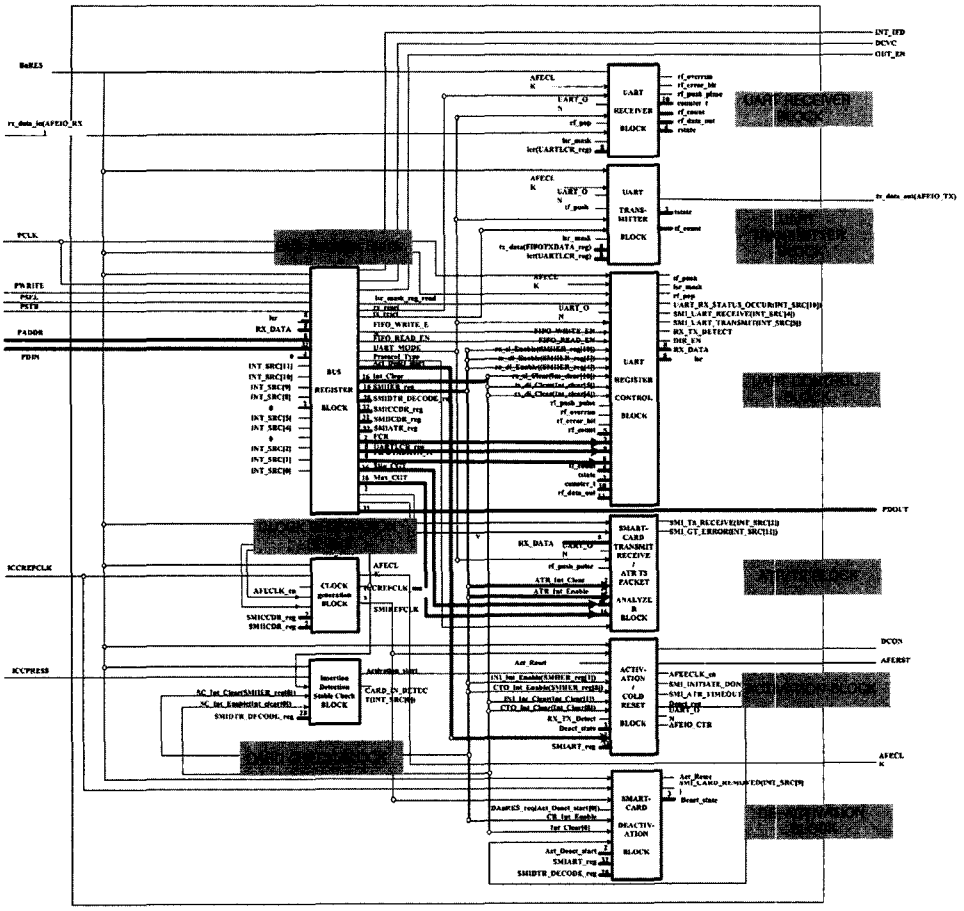


그림 3 IC카드 인터페이스 전체 블록도

표 1 Register configuration

Name	Attribute	Initial	Address
Data register	R/W	0x00	ICCIF Base + 0x00
Control register 0	R/W	0x00	ICCIF Base + 0x04
Control register 1	R/W	0x00	ICCIF Base + 0x08
Interrupt Status register	R/W	0x00	ICCIF Base + 0x0C
Retry Limit register	R/W	0xff	ICCIF Base + 0x10
Receive Reset register	R/W	0x00	ICCIF Base + 0x14
Transmit Reset register	R/W	0x00	ICCIF Base + 0x18
Smart Card status register	R/W	0x00	ICCIF Base + 0x1C
Debounce timer	R/W	0xffff	ICCIF Base + 0x20
Activation event timer	R/W	0xffff	ICCIF Base + 0x24
Deactivation event timer	R/W	0xffff	ICCIF Base + 0x28
ATR reception start time	R/W	0xffff	ICCIF Base + 0x2C
Block receive time-out	R/W	0xffff	ICCIF Base + 0x30
Character transfer time-out	R/W	0xffff	ICCIF Base + 0x34
Baud rate clock	R/W	0x0000	ICCIF Base + 0x38
Baud cycles	R/W	0x00	ICCIF Base + 0x3C
Character transfer guard time	R/W	0xff	ICCIF Base + 0x40
Block guard time	R/W	0xff	ICCIF Base + 0x44
Interrupt clear register	R/W	0x0	ICCIF Base + 0x48

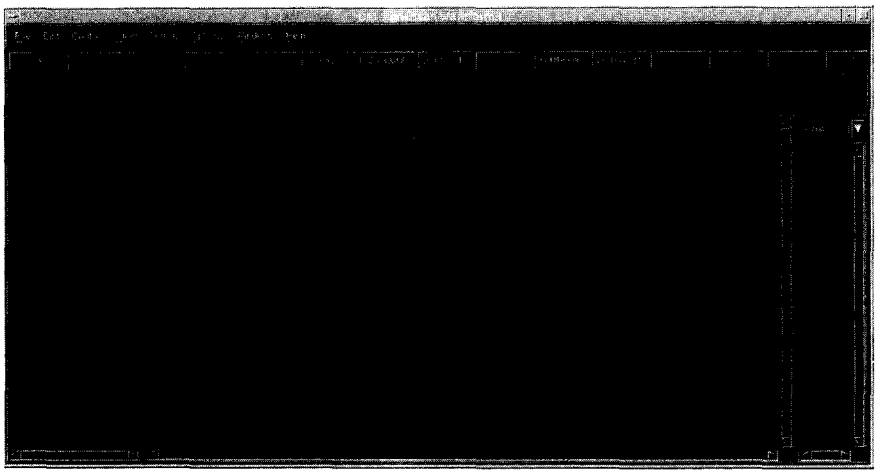


그림 4 Pre-Simulation results

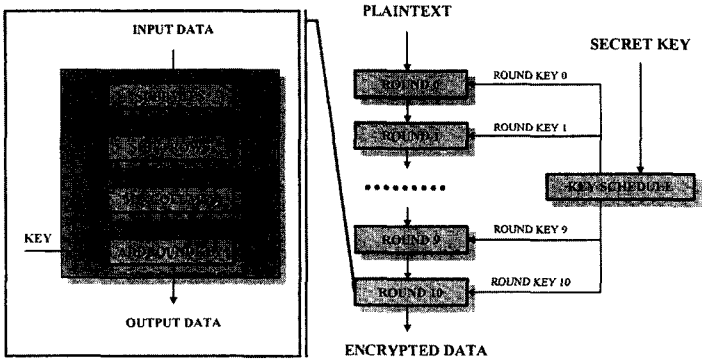


그림 5 AES 알고리즘 흐름도

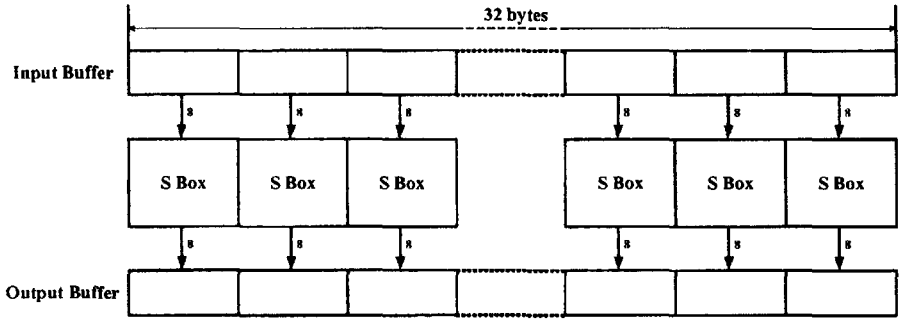


그림 6 Substitution부의 블럭도

블록은 내부 레지스터 및 look-up 테이블로 구성되어 진다. 마지막으로 Mix Column 블록은 GF(2^8)상의 곱셈 및 XOR 연산으로 구성되어 진다. 이와 함께 키 제어부는 현재의 키 값과 이전의 키값의 관계를 이용해 다음 서브 라운드의 키값을 생성하게 되며, 각 블록의 세부 블록도와 동작 원리는 아래와 같다.

2.6.1 Substitution

Substitution 블록은 데이타 256bit을 8bit단위의 32개 그룹으로 나누고, 각각의 비트를 S-box의 어드레스로 사용하여 새로운 256bit의 데이타로 치환하는 역할을 한다. 병렬 연산을 위해 32개의 S-box를 내부 레지스터에 저장하도록 구현하였으며, 블록도는 그림 6과 같다.

2.6.2 Shift Row

Shift Row 블록의 경우 256bit의 데이타를 2개의 그룹으로 나누게 된다. 각각의 128bit 단위의 2개 그룹은 그림 7과 같이 바이트 단위로 쉬프트 연산을 수행하게 되며, 이렇게 함으로써 전체 데이타의 순서를 규칙성 있게 뒤 썩게 된다.

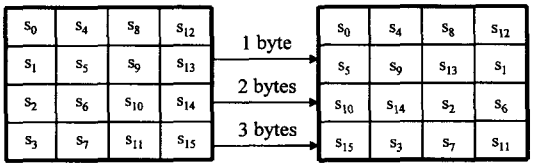


그림 7 Shift Row부의 블록도

2.6.3 Mix Column

Mix Column 블럭은 아래 그림 8-(a)와 같이 shift row한 데이타를 계수 값과 매트럭스 연산을 통해 얻어진다[2,4,5]. 따라서, 행렬 곱셈을 전개하게 되면 GF(2^8) 상에서의 곱셈과 덧셈 연산으로 분리되게 된다. Mix Column의 역변환 역시 그림 8-(b)에서와 같이 계수 값만 바뀌게 되므로, 암호 또는 복호시 MUX를 통해 계수 값을 선택해 사용함으로써 동일한 하드웨어를 반복해 사용하도록 설계하였다.

2.6.4 Key Addition

Key Addition 블럭은 256bit의 데이타를 256bit의 키 값과 bit단위 XOR를 하는 부분으로 전체 블록도는 그림 9와 같다.

2.6.5 AES 암호화 모듈의 검증

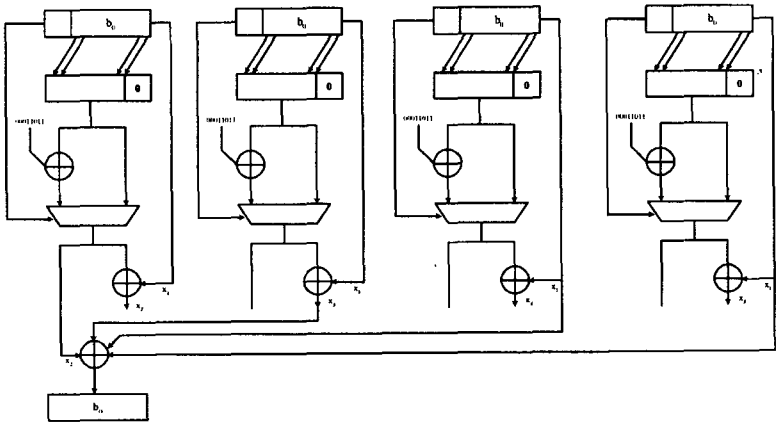
설계된 AES 암호화 모듈은 그림 10과 같이 구성되어 있고, 임의의 텍스트 문서를 암호화 한 후 다시 이 결과 파일을 복호기에 넣어 원문을 복원하는 방식으로 검증하였으며, Modelsim을 통한 pre-simulation 결과는 그림 11과 같다.

$$\begin{bmatrix} s'_0 & s'_4 & s'_8 & s'_{12} \\ s'_1 & s'_5 & s'_9 & s'_{13} \\ s'_2 & s'_6 & s'_{10} & s'_{14} \\ s'_3 & s'_7 & s'_{11} & s'_{15} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \otimes \begin{bmatrix} s_0 & s_4 & s_8 & s_{12} \\ s_1 & s_5 & s_9 & s_{13} \\ s_2 & s_6 & s_{10} & s_{14} \\ s_3 & s_7 & s_{11} & s_{15} \end{bmatrix}$$

(a) Mix Column의 수식적 표현

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 0e & 0b & 0d & 09 \\ 09 & 0e & 0b & 0d \\ 0d & 09 & 0e & 0b \\ 0b & 0d & 09 & 0e \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

(b) Mix Column의 역변환



(c) Mix Column의 전체 블럭도

그림 8 Mix Column부의 구현

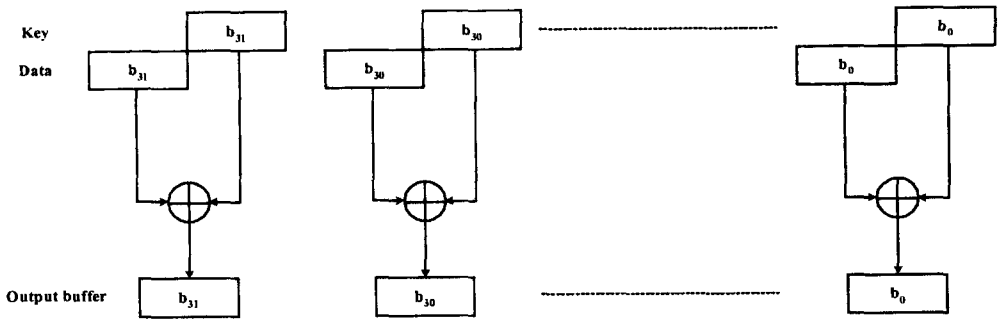


그림 9 Key Addition 블럭

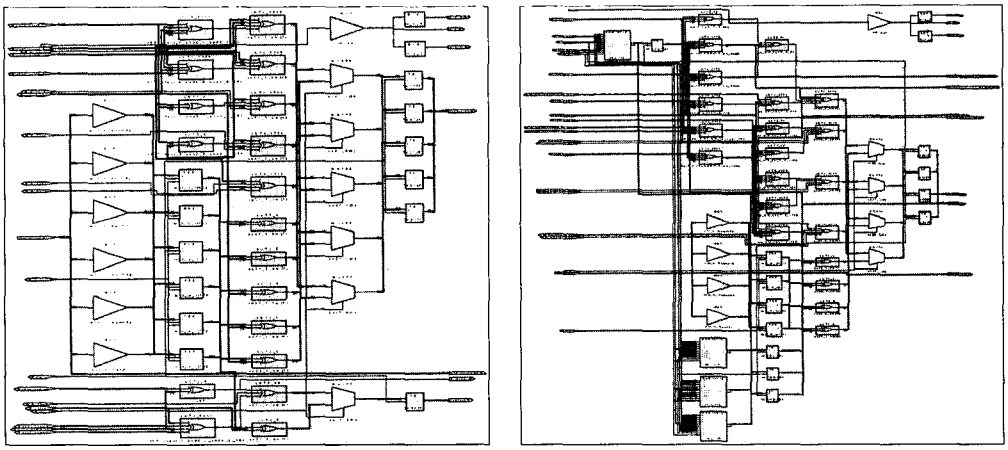


그림 10 AES 암호화 모듈의 구현도

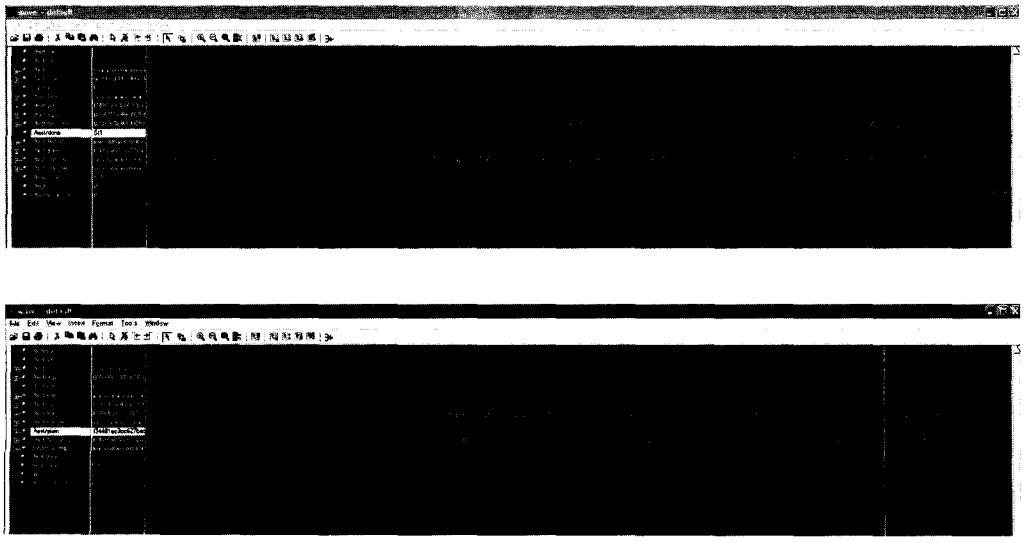


그림 11 AES 암호화 모듈의 시뮬레이션 결과

3. IC 칩셑의 검증

3.1 Validation을 통한 IC 카드 칩셑의 검증

ARM사의 코어를 기반으로 하는 SoC 설계를 위해 제공되는 검증 환경을 Validation이라 하며, 이러한 Validation의 주요 역할은 디자인된 블록이 시스템에 사용되는 경우 이들 동작을 테스트하는 데 있다. 특히 블록 디자이너가 고려하기 힘든 블록간의 interaction을 체크하는 데 있다. 이러한 validation을 위해서는 다음의 것들이 고려되어야 한다. TrickBox란 기본적으로 ASIC Chip 상에서 외부 interface를 가지는 장치들에 대해 validation programmer가 self-checking-code를 만들기 위해 필요한 여러 가지 function을 구현한 I/O이다. 이를 통해 칩의 여러 신호와 포트들에 대해 read/write 테스트를 할 수 있다. 또한 장치를 검증하기 위해 필요한 레지스터도 컨트롤 할 수 있다. 이러한 TrickBox를 구현해 놓으면 validation code작성에 유용하다. 또한 Evaluation board상에서 사용될 수 없는 time critical 한 입출력을 테스트 하고자 할 때는 Virtual TrickBox를 사용할 수 있다. 이는 주로 HDL 모의 실험 시간을 줄일 수 있는 수단을 제공한다. 큰 용량의 메모리 없이 전체 어드레스 버스 영역에 해대 테스트를 수행하기 위해서는 Virtual Memory를 사용한다. 또한 내/외부 버스의 동작 여부를 알아보기 위해서는 Bus Watcher를 사용할 수 있다. 물론 앞에 언급된 것들은 simulation 환경에서만 이용 가능하다. 그림 12는 이렇게 구현된 validation code를 이용한 결과 화면을 보여주고 있으며, APB를 통해 설계된 블록에 데이타를 쓰고 읽는 과정을 보여주고 있다.

3.2 ARM7 Based SoC 설계 및 ASIC 제작

설계된 chip의 패키지 타입은 352Pin PBGA로 설계의 안정된 동작검증을 위하여 Validation이라는 검증기법을 사용하여 각각의 블록들의 동작상황을 설정하여 테스트하였으며 ASIC 설계 공정은 현대 2 ploy 3 metal의 0.35 um process를 이용하여 설계하였으며, 전체 칩의 레이아웃 도는 그림 13과 같다.

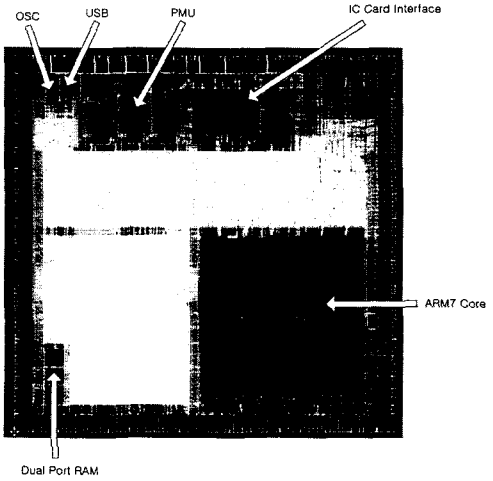


그림 13 Full chip layout

4. 결 론

본 논문에서는 스마트카드 단말기의 핵심 부품으로 ARM7 Core를 중심으로 사용자 인터페이스를 위해 LCD 및 키패드 등의 주변 블럭의 설계와 다채널 IC카

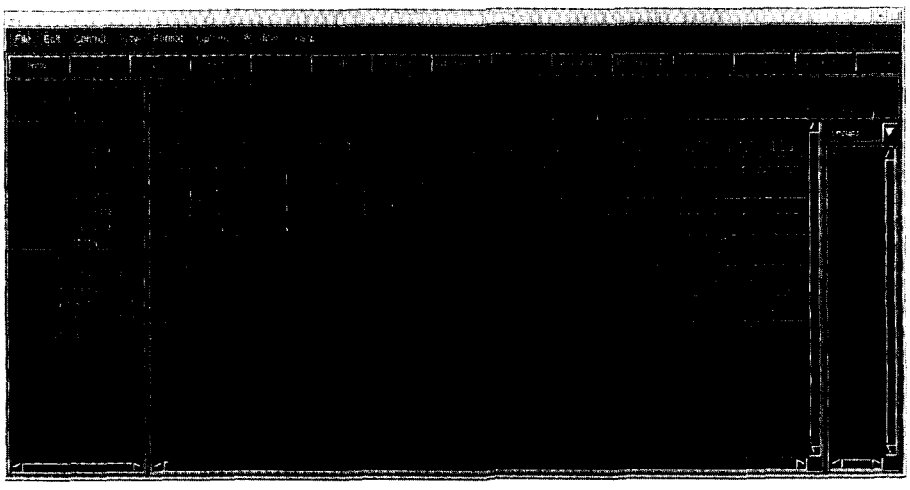


그림 12 Validation simulation 결과

표 2 AES 암호화 모듈의 성능 비교

CPU	Feature	Key Schedule	Encryption	Decryption
ARM7TDMI	Gladman AES	449	1641	2763
	ANSI-C DES	-	430	870
ARM9TDMI	Gladman AES	333	1374	2439
	ANSI-C DES	-	387	675
Full H/W	-	5	25	27
Efficiency	-	89.8/66.6	65.64/54.96	102.33/90.33

드 인터페이스 기능을 갖는 칩을 구현함과 동시에 AES 암호화 모듈을 내장함으로써 최적의 IC카드 단말기 시스템 구현에 핵심적인 역할을 수행 할 수 있도록 하였다. 구현된 AES 모듈은 소프트웨어로 구현된 기존의 암호화 모듈보다 고속의 암호화가 가능해지며, IC카드 칩셀이 부가적인 서비스 역시 가능해지리라 생각된다. 표 2는 Gladman의 AES C 코드 및 DES의 C 코드를 이용해 ARM사의 개발 툴 ADS 1.1을 통해 메모리 액세스가 zero wait일 경우를 가정해 측정한 결과로 하드웨어로 구현한 AES 암호화 모듈이 암호화시 평균 60배, 복호화시 100배의 성능 향상을 얻을 수 있었다. 이러한 하드웨어로 구현함으로써 얻을 수 있는 성능 향상은 실시간 기반의 전자 상거래 시스템에 보다 효과적으로 적용 할 수 있다. 이와 함께 상대적으로 암호화에 필요한 마이크로 컨트롤러의 연산 량을 줄여줌으로써, 다양한 부가 서비스 및 원가 절감에 기여 할 수 있을 것이라 생각한다.

본 논문에서 파생되는 시스템IC 설계기술은 향후 프로세서와 OS의 탑재가 필수적인 전자지불용 휴대형 무선 통신 시스템과 근거리 네트워크에서 사용자 인증을 관장하는 지능형 보안모듈을 내장하는 IC설계에도 활용될 것으로 보이며, 개방형 전자화폐 시스템은 2003년을 기점으로 지금까지 폐쇄형 전자화폐시스템이 주도하였던 전자지불시장에 본격적으로 시장 진입을 이룰 것이며, 이 경우 개방형 전자화폐시스템에 적합한 IC카드 단말기 제품 경쟁력을 확보하여 내수 및 수출 증대에 기여할 것으로 생각한다.

참 고 문 헌

[1] E. Biham, "New types of cryptanalytic attacks using related keys," Advances in Cryptology, Proceedings Eurocrypt'93, LNCS 765, T. Helleseth, Ed., Springer-Verlag, 1993, pp. 398-409.

[2] EVM'96 Integrated Circuit Card Specification for payment systems.

[3] R. Anderson, E. Biham, and L. Knudsen, "Serpent: A Proposal for the Advanced Encryption Standard," First Advanced Encryption Standard (AES) Conference, Ventura, CA, 1998.

[4] J. Daemen and C. Clapp, "Fast hashing and stream Encryption with PANAMA," Fast Software Encryption, LNCS 1372, S. Vaudenay, Ed., Springer-Verlag, 1998, pp. 60-74.

[5] V. Rijmen, "Cryptanalysis and design of iterated block ciphers," Doctoral Dissertation, October 1997, K.U.Leuven.

[6] M. Matsui, "Linear cryptanalysis method for DES cipher," Advances in Cryptology, Proceedings Eurocrypt'93, LNCS 765, T. Helleseth, Ed., Springer-Verlag, 1994, pp. 386-397.

[7] W. Rankl and W. Effing, Smart Card Handbook, John Wiley & Sons, LTD, 2000.

김 동 순

1997년 2월 인하대학교 전자재료공학과 공학사. 1999년 2월 인하대학교 전자재료공학과 공학석사. 1999년 1월~현재 KETI DMB 사업단 전임연구원 근무 관심분야는 통신용 신호처리, 고성능 MCU설계, IC카드, SoC Design

이 성 철

1993년 2월 전북대학교 정보통신공학과 공학사. 1995년 2월 전북대학교 정보통신공학과 공학석사. 1995년 3월~현재 KETI SOC연구센터 선임연구원 근무 관심분야는 통신용 신호처리, IC카드, SoC Design