타원곡선형 공개키 연산기의 통합설계에서의 인터페이스 통신 부담 축소화 방안

이완복, 노창현
중부대학교

Interface Communication Overhead Reduction In A Codesign Case Study of ECC Crypto Algorithm

Wan-Bok Lee, Chang-Hyun Roh
Joongbu University
E-mail : wblee@joongbu.ac.kr

요 약

최근 반도체 기술과 회로 설계 기술이 발달하면서, 하드웨어와 소프트웨어 부분을 별도로 분리하여 설계하지 않고, 통합하여 설계함으로써 적은 비용으로 고성능의 시스템을 구축할 수 있는 통합설계 기반이 구축되었다. 그러나, 하드웨어와 소프트웨어가 혼재할 경우에는 두 영역 사이에서의 통신 부담이 비교적 큰 편으로 발생하여 오히려 소프트웨어로만 설계한 경우보다 성능이 떨어질 소지가 있다. 본 논문에서는 이러한 통신 부담을 줄일 수 있는 방안에 대해 세 가지를 간략히 소개하고 있다.

Ⅰ. 서 론

근래에 온라인 증권거래와 인터넷 뱅킹 서비스가 초고속 정보통신망을 기반으로 생활화 된 것에 이어서, 최근에는 휴대 단말기에서도 전자상거래가 활성화될 전망이다. 특히, 공개키 암호 알고리즘에 기반을 둔 전자 결재 솔루션들은, 휴대 단말기에서도 곧 적용될 전망이지만, 공개키 연산 자체의 많은 계산량과 휴대 단말기의 제약점으로 인해 하드웨어적으로 구현하거나 하드웨어적인 도움을 통해 계산량을 줄일 필요가 있게 되었다.

더우기, 최근에는 반도체 기술과 회로 설계 기술이 발달하면서, 칩 하나에 시스템을 구현할 수 있는 SoC 기술이 개발되었으며, 시스템을 구현할 때 하드웨어와 소프트웨어 부분을 별도로 분리하여 설계하지 않고, 통합하여 설계함으로써 적은 비용으로써 고성능의 시스템을 구축할 수 있는 통합 설계 기반이 구축되었다. 그러나, 하드웨어와 소프트웨어가 혼재할 경우에는 두 영역 사이에서의 통신 부담(Overhead)이 비교적 큰 편이어서 오히려 소프트웨어로만 설계한 경우보다 성능이 떨어질 소지가 있다.

본 논문에서는 통합설계시의 이러한 통신 부담을 줄일 수 있는 방안에 대해 간략히 제시하고자 한다. 본 논문의 구성은 다음과 같다. 먼저 2장에서는 공개키 암호 연산 알고리즘과 암호 알고리즘 부분 중 계산력이 가장 많이 소요되는 부분에 대해 언급한다. 3장에서는 통합설계 방법과 하드웨어 소프트웨어간의 통신 부담을 줄일 수 있는 설계 방안에 대해 소개하고 4장에서 결론을 맺는다.

Ⅱ. 공개키 암호 연산 알고리즘

2.1 암호 알고리즘 개요

현대 암호방식은 키 관리 측면에 따라 크게 대칭키 암호 알고리즘과 공개키 암호 알고리즘으로 분류된다. 대칭키 암호 알고리즘은 송.수신자가 동일한 키에 의하여 암호화 및 복호화 과정을 수행하는 방식을 일컫는다. 이러한 알고리즘의 예로는 DES, AES 등이 있으며, 국내에서 개발된 SEED도 여기에 해당된다. 대칭키 암호알고리즘의 최대의 난제는 암호화 과정에서 사용되는 키의 안전한 분배가 어렵다는 것이다. 암호가 군사 혹은 외교 등의 한정된 분야에만 사용되던 시대와는 달리 현재는 불특정다수에 의한 데이터의 교환이나 프라이버시 보호 또는 각종 전자상거래를 포함한 민간 부분에서도 수요가 급격히 증가하고 있다. 따라서 비밀 통신을 하고자 하는 양자간에 키의 안전한 전송은 매우 중요한 문제였었다. 이 문제는 Diffie와 Hellman이 1976년 발표한 논문에서 제시한 공개키 암호방식을 통하여 키의 일부를 공개함으로써

해결될 수 있었다. 이 방식에는 암호용 키가 공개키와 개인키 쌍으로써 존재하며, 공개키는 누구나 사용 가능하고 개인키는 비밀리에 보관되어 사용되어 지는 형태이다. 이 방식은 비밀키 암호시스템의 키 관리 및 분배의 문제점을 해결하게 되었으며, 동시에 디지털 서명에도 사용 가능하다는 측면에서 그 응용 범위가 매우 넓은 편이다. 이제까지 제안된 공개키 암호 중에는 RSA, ElGamal, 타원곡선 암호 (ECC) 등이 대표적으로 많이 사용되고 있고, 이들은 모두 가환군 내에서의 이산대수나 소인수분해 문제에 기반하고 있으며 효과적인 알고리즘도 계속 연구되고 있다. 가장 널리 알려져 있는 RSA 공개키 암호 방식은 실제적인 암호 공격을 견디기 위해서 키의 길이가 상당히 긴 편이기 때문에, 연산능력이 제한된 이동 통신 단말기에는 사용이 제한될 수 있다. 이에 1995년 Koblitz와 Miller는 타원곡선을 이용한 공개키 암호 시스템을 구성할 수 있다고 제안하였다. 이 방식은 비트당 안전도가 타 공개키 시스템보다 효율적이라는 것이 알려져 있다. 타원곡선 암호 방식은 짧은 키 길이를 가지고도 다른 공개키 암호 시스템과 동등한 안전도를 제공할 수 있다. 또한 짧은 키 길이를 가지기 때문에 요구되는 대역폭과 메모리가 작아지며, 이로 인해 메모리와 처리능력이 제한된 스마트카드, 이동통신에서의 보안성 제공 같은 응용에서 중요한 기반 암호 기술로서 예측되고 있다. 또한 공개키 암호의 사용 범위는 계속 넓어지고 있으며, 시장 규모도 커지고 있어 세계 각국은 자국의 고유한 공개키 암호 방식을 확보하려 노력하고 있는 실정이다. 최근에는 휴대폰, PDA 등을 비롯한 각종 휴대 단말에서의 암호 방식에 기반을 둔 전자상거래가 많이 활성화되고 있다. 최근 몇 년 사이에 인터넷 온라인 증권 매매 거래가 매우 활성화 되었듯이, 앞으로는 대부분의 전자상거래와 뱅킹 서비스 등이 휴대 단말기 상에서 제공될 가능성이 매우 농후해졌다. 그러나, 휴대 단말기들은 계산 능력과, 소비전력 등이 제약되기 때문에 고 수준의 암호 방식을 지원하기가 어려우며, 이로 인해 높은 수준의 보안성을 제공하기가 현실적으로 어려운 문제점이 있다. 따라서, 암호 알고리즘을 저전력 및 고속으로 수행할 수 있도록 알고리즘 자체를 하드웨어적으로 구현하거나, 또는 소프트웨어와 하드웨어의 통합설계를 통하여 성능을 선하는 것이 필요하다.

2.2 타원 곡선형 공개키 연산 알고리즘의 분석

타원 곡선형 공개키 알고리즘은 고속 및 저전력 요구사항에 잘 부합하는 알고리즘이다. polynomial 기반 타원곡선 암호하드웨어는 Certicom에서 제안한 163비트 타원곡선 상에서의 타원곡선 암호 알고리즘이 널리 사용되고 있다. 이는 이미 안전성이 검증되어 타원곡선을 사용함으로 안전성을 보장하기 용이하기 때문이다. 제안된 163비트 polynomial 기반 타원곡선 파라미터는 다음과 같다. 이때, m은 타원곡선 파라미터의 비트 길이이고, f(x)는 최소다항식이며, a와b는 타원곡선 상수계수값이다. 또한G는 타원곡선의 base point 좌표값으로 (c)는 compressed된 값이며, (uc)는 uncompressed된 값이다. 그리고 n은 G의 order값이다.

163비트 polynomial 기반 타원곡선 파라미터
- T(m, f(x), a, b, G, n, h)
- m163
- f(x)x163 + x7 + x6 + x3 + 1
- a = 07 B6882CAA EFA84F95 54FF8428 BD88E246 D2782AE2
- b = 07 13612DCD DCB40AAB 946BDA29 CA91F73A F958AFD9
- G(c) = 0303 69979697 AB438977 89566789 567F787A 7876A654
- G(uc) = 040369 979697AB 43897789 56678956 7F787A78 76A65400 435EDB42 EFAFB298 9D51FEFC E3C80988 F41FF883
- n = 03 FFFFFFFF FFFFFFFF FFFF48AA B689C29C A710279B

파라미터 중 타원곡선 암호프로세서에 직접적인 영향을 미치는 값은 최소 다항식 값인 f(x)와 타원곡선 상수 값인 a값이다. 이 두 값은 하드웨어 최적화를 위하여 연산기 내부에 값을 내장하여 사용하며, 그 외 다른 파라미터 값들은 프로그램 코드로 저장하여 이용하는 것이 좋다. 위의 파라미터들이 적용되는 polynomial 기반의 타원곡선은 정의되며, 타원곡선 암호 알고리즘 연산에서 사용되는 모든 좌표는 이 타원곡선 상의 한 점이어야 한다. 타원곡선 암호알고리즘에서 수행되는 기본 연산은 타원곡선의 스칼라 곱셈 연산이다. 스칼라 곱셈 연산은 임의의 랜덤수 k와 타원곡선 위의 한 점 P의 곱셈 연산으로 정의되며, 타원곡선 위의 점 P의 k번 덧셈연산으로 계산할 수 있다. 이때 타원곡선의 덧셈연산은 결과값이 다시 타원곡선 위의 점이 되도록 <알고리즘11-1>과 같이 정의 된다.

<알고리즘 1> 타원곡선 덧셈 연산

Input : $P1 = (x1, y1)$, $P2 = (x2, y2)$.
Output : $P3 = P1 + P2 = (x3, y3)$.
1.$P1 = P2$(doubling)
  $x3 = 2 ++ a, y3 = x12 + (+1)\, x3 (= x1 + y1 / x1)$
2.if $P1P2$ (point addition)
  $x3 = 2 ++ x1 + x2 + a, y3 = (x1 + x3) + x3 + y1$ where $(= (y2 + y1) / (x2 + x1))$
Return $(x3, y3)$

실제 타원곡선 암호 알고리즘을 이용한 프로토콜에서 스칼라 곱셈의 입력값들은 안전성을 위하여 order가 최대인 기저점과 163비트의 임의의 랜

덤수가 된다. <알고리즘 11-1>의 타원곡선 덧셈 연산의 대부분의 연산시간은 polynomial 기반 유한체의 곱셈 연산과 역승산 연산이 차지하기 때문에 전체 암호화 성능에 가장 밀접한 연관을 갖는다.

<알고리즘 2> Polynomial 기반 곱셈 연산 알고리즘

Input : Binary Polynomials $a(x)$ and $b(x)$.
Output : $c(x) = a(x)b(x) \bmod f(x)$.
1. 0.
2.i from m 1 to 1 do
2.1If bi = 1 then c $c + a$.
2.2 c c x mod $f(x)$.
3. If $b0$ = 1 then c $c + a$.
4. Return (c).

Polynomial 기반 유한체의 역승산기는 AIA(Almost Inverse Algorithm) 알고리즘으로 다음과 같이 구성될 수 있다.

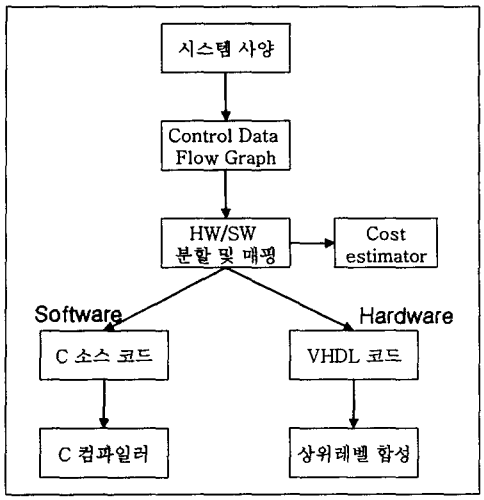
<알고리즘 3> Polynomial 기반 유한체의 역승산 알고리즘(MAIA)

Input : a F2m , a 0.
Output : $a1$ mod$f(x)$.
1. 1, c 0, u a, v f, k 0.
2.x divides u do :
2.1 u / x.
2.2x divides b then b b / x
else b ($b + f$) / x.
3.u = 1 the return (b).
4.deg(u) < deg(v) then : uv, bc.
5. $u + v$, b $b + c$.
6.step 2.

III. 공개키 연산기의 효과적인 통합설계 방안

3. 1. 통합설계 방법론

Smart card, PC Card, 또는 휴대폰에 내장될 암호 연산기는 그 특성상 고속 및 저전력으로 동작되어야 그 의의가 있다. 그러나, 일반적인 HW 구현에 동작 속도와 소비전력은 반비례 관계가 있으며, HW 설계에 있어서 그 타협점을 찾는 것은 대안이 될 수 있는 경우의 수가 매우 많기 때문에 쉽지 않은 문제점이 된다. 전통적으로 이러한 문제를 해결하기 위해서 제시된 이론이 하드웨어 소프트웨어 통합설계 이론이다.



<그림 1> 하드웨어 소프트웨어 통합설계 흐름도

하드웨어 소프트웨어 통합설계 기법은 <그림 1>에서 보이는 바와 같이 시스템의 Control data flow 그래프를 분석한 후, 개별적인 작업의 단위를 하드웨어 또는 소프트웨어로 분할 및 매핑을 하게 된다. 전체 시스템이 계산 속도, 소비 전력, 칩 면적 등을 만족하는지 살펴보기 위해 하드웨어 소프트웨어 코시뮬레이션을 통하여 디자인의 적절성을 평가하게 된다. 통합설계에 관해서는 현재까지 많은 연구가 관련되어 진행되어 왔으며, 멀티미디어 데이터 인코더 같은 어플리케이션에 특화된 몇몇의 사례에서는 성공적으로 적용되어 보고된 것도 있다. 통합설계 기법은 반도체 기술이 발전하고, SoC 설계 기술이 점차 발전함에 따라, 추 후 더욱 유망할 것으로 기대되고 있지만, 아직도 여전히 많은 연구가 필요하다. 예를들어, 코시뮬레이션 과정은 많은 컴퓨터 계산력을 필요로 하고, 하드웨어 소프트웨어 분할 및 매핑 문제 등도 원천적으로 풀기 어려운 문제들이기 때문이다.

3.2. ECC 공개키 알고리즘의 통합 설계시 고려사항

ECC 공개키 알고리즘을 소프트웨어적으로 구현할 때 가장 많은 시간이 멱승 계산에 소요된다. 따라서 멱승 계산을 하드웨어적 구현하여 효율적으로 할 수 있다면, 전체 성능을 개선할 수 있을 것이다. 그러나, 멱승 연산 자체를 하드웨어로 분리할 경우에는 잦은 하드웨어 호출과 그 때 발생하는 통신 부담으로 인해 속도가 느려질 전망이다. 이에, 다음과 같이 하드웨어와 소프트웨어간의 통신 부담을 축소할 수 있는 세가지 방안을 고안한다.

3.3. 하드웨어 소프트웨어 Interface 통신 부담의 축소화 방안

하드웨어로 구현된 시스템은 소프트웨어로만 구성된 시스템보다 빠르게 동작하는 것이 일반적이

다. 그러나, 많은 하드웨어가 사용될수록 비용이 비싸지므로 하드웨어와 소프트웨어간의 적절한 배분을 하면서 설계하는 것이 요구되는데, 이러한 배경에서 태동된 것이 하드웨어 소프트웨어 통합설계론이라고 볼 수 있다. 다시 말해서, 하드웨어와 소프트웨어는 각각의 장단점이 있기 때문에 혼합되는 것이 바람직하지만, 이 혼합과정에서 하드웨어와 소프트웨어간의 체계적이고도 면밀한 분석이 없이 조합되었을 경우에는 오히려 소프트웨어로만 구성되었을 때보다도 더 느려질 수 있다. 이것은 하드웨어와 소프트웨어가 상호 인터페이스하는 과정에서 비교적 많은 시간과 부담(Overhead)이 발생하는 데에서 기인한다.

이러한 부담은 소프트웨어가 하드웨어를 제어하기 위해서 Device Driver, 운영체제의 시스템 콜 등의 서비스 계층들을 거치기 때문인데, 하드웨어 제어 방안을 경량화시켜서 극복할 수 있다. 이러한 방안으로 다음과 같은 세가지 방안을 생각해 볼 수 있다.

첫째, 고속의 GPIO(General Purpose Input Output) 포트를 활용하여 하드웨어를 구동한다. 보통의 GPIO는 사우스 브리지 아래에서 비교적 느린 속도로 동작하지만, 하드웨어를 쉽게 적용할 수 있는 잇점이 있다. 하드웨어와의 접속을 간략히 하기 위해 GPIO는 자주 사용되는데, 그 속도가 빠를 경우에는 GPIO를 이용하여 별도의 하드웨어 제어를 위해 사용할 수 있다.

둘째, 마이크로프로세서의 명령어 레벨에서 하드웨어를 구동할 수 있다. Xtensa, ARC-tangent, Jazz 등은 기본적인 코어와 명령어 집합을 제공한다. 이들은 응용에 따라 다르게 구성되거나 확장될 수 있도록 설계되어 있기 때문에 구성 가능한 프로세서 (configurable processor) 라고 부른다. NIOS(Altera)도 구성 가능한 프로세서로서 재구성 가능한 논리회로 상에 프로그램 될 수 있는 형태로 제공된다. 그러나 구성을 할 수 있는 정도는 제한되어 있어서 단지 다섯 개의 opcode만이 사용자가 지정하는 명령어를 만드는데 사용될 수 있다.

셋째, 내부의 고속 버스에 memory mapped 방식으로 하드웨어를 연동시킨다. 특히, Altera사의 Excalibur 칩은 이러한 방식으로 통합 설계에 활용될 수 있는 좋은 플랫폼이 된다. Excalibur칩은 ARM9 프로세서와 100만게이트 로직을 하나의 칩에 집적하였기 때문에 내부의 AMBA 버스와 연동되는 로직을 설계하여 특정 메모리에 매핑되도록 할 수 있다.

IV. 결 론

본 논문에서는 통합설계시의 하드웨어와 소프트웨어간의 통신 부담을 줄일 수 있는 방안에 대해 세가지를 간략히 제시하였다. 통합설계는 비교적 저렴한 하드웨어 가격으로 고성능을 발휘할 수 있는 시스템을 구성하는데 효과적일 수 있다. 그러나, 하드웨어와 소프트웨어간의 통신 부담이 클 경우에는 오히려 소프트웨어만으로 구성된 시스템보다 느릴 수 있기 때문에 주의가 필요하다. 이러한 문제점을 극복하기 위해 본 논문에서 제안하는 방안들은 참조될 수 있으며, 향후 공개키 연산 알고리즘을 통합 설계할 시 시뮬레이션을 통하여, 그 효과성을 보일 수 있다.

참고문헌

[1] 조성제, 권용진, 타원곡선 암호시스템을 위한 기저체 연산기의 FPGA 구현 FPGA, 대한전자공학회 00 추계 종합학술대회 논문집, pp. 148 -151, 2000.

[2] 신형철, 하드웨어-소프트웨어 통합 설계 기술, 전자공학회지 v.024, n.012, pp.69-78 , 1997,12.