[4-35]

$GF(2^m)$상의 Scalable 몽고메리 곱셈기†

이광진*, 장용희*, 권용진*

*한국항공대학교 정보통신공학과

{kjlee*, yhjang*, yjkwon*}@tikwon.hangkong.ac.kr

A Scalable Architecture of Montgomery Multiplier on $GF(2^m)$

Kwang-jin Lee*, Yong-hee Jang*, Yong-jin Kwon*

*Department of Telecom. & Inform. Eng., of HanKuk Aviation University

요 약

몽고메리 곱셈 알고리즘을 이용한 모듈러 곱셈기 연구의 대부분은 하드웨어 성능의 향상과 면적의 감소만을 고려한 구조로 데이터의 길이가 증가될 경우 곱셈기의 재설계가 필요하다. 이러한 문제점을 해결하기 위해 본 논문에서는 데이터의 길이가 변경될 경우 곱셈기의 재설계 없이 연산이 가능한 scalable 구조의 모듈러 곱셈기를 설계한다. 설계된 구조는 FPGA를 이용하여 하드웨어로 구현하고 시뮬레이션 툴을 이용하여 기능 및 타이밍 시뮬레이션을 수행한다. 또한 면적 및 수행속도를 분석하여 효율적인 하드웨어의 구조를 제시하고 타 논문과의 수행속도를 비교 분석한다. 설계된 곱셈기의 구조는 데이터의 길이에 독립적으로 수행되므로 다양한 시스템에 쉽게 적용이 가능하다.

Ⅰ. 서론

최근 급속한 인터넷의 발전과 더불어 전자상거래가 활성화됨에 따라서 인증 및 정보의 암호화에 대한 중요성이 증가하고 있다. 그래서 정보보안에 필요한 여러 가지 보안 기술이 개발되고 있고 암호화 시스템의 중요성이 크게 부각되고 있다. 암호화 시스템은 공개키 또는 비밀키 암호 알고리즘을 기반으로 하고 있으며 특히 공개키 암호 알고리즘을 기반으로 한 암호화 시스템은 전자 서명 및 인증이 용이함으로 전자상거래 등에 주로 사용된다. 이러한 암호화 시스템을 소프트웨어로 구현할 경우 쉽게 구현이 가능하며 적은 양의 데이터를 암·복호화할 경우 빠른 수행속도로 처리할 수 있다. 그러나 날로 증가하는 대량의 데이터와 실시간으로 암·복호화가 요구되는 인터넷 환경에서는 적합하지 않다. 그래서 최근에 암호화 시스템을 효율적으로 수행하는 하드웨어의 개발이 관심의 대상이 되고 있다. RSA, ECC[1] 암호 알고리즘과 같은 대부분의 공개키 암호 알고리즘은 유한체상에서의 연산으로 덧셈과 뺄셈의 경우는 연산의 복잡도가 낮은 각각의 비트 논리 연산만을 사용하기 때문에 하드웨어의 효율성은 연산의 복잡도가 높은 곱셈 연산에 의해 좌우된다. 따라서 모듈러 곱셈연산을 효율적으로 계산하기 위한 많은 방법들이 연구되어 왔으며 하드웨어 구현시 속도와 면적의 효율성 문제로 몽고메리 곱셈 알고리즘을 이용한 몽고메리 곱셈기의 연구가 활발히 진행되어 왔다. 그러나 곱셈기연구의 대부분은 수행속도와 면적만을 고려한 구조로써 암호화 강도 또는 암호알고리즘의 변경으로 인한 암호화 시스템의 키 길이가 변경될 경우 하드웨어의 재설계가 불가피하다. 그래서 최근 암호화 시스템에서는 키 길이가 변경되어도 하드웨어 구조의 변경 없이 재사용 가능한 모듈러 곱셈기의 scalable 구조가 연구되고 있다[1][2][3][4][5][6].

본 논문에서는 키 길이와 같은 데이터의 길이 변경시 하드웨어의 재설계 없이 사용할 수 있는 $GF(2^m)$상의 scalable 몽고메리 곱셈기의 구조를 설계한다.

본 논문의 구성은 다음과 같다. 제 2장에서는 $GF(2^m)$상의 scalable 몽고메리 곱셈 알고리즘에 대해 살펴보고 제 3장에서는 $GF(2^m)$상의 scalable 몽고메리 곱셈기의 구조와 PE의 구조 및 메모리의 구성을 알아본다. 제 4장에서는 본 논문에서 설계된 곱셈기구조의 하드웨어 구현 결과와 그 구현에 따른 하드웨어 면적과 속도를 분석하고 타 논문과의 수행속도를 비교분석한다.

Ⅱ. $GF(2^m)$상의 scalable 몽고메리 곱셈 알고리즘

식(1)은 몽고메리 곱셈 알고리즘을 나타내는 것으로 모듈러 곱셈연산을 나눗셈으로 처리하지 않고 쉬프트와 덧셈

† 본 논문은 과학기술부·한국과학재단 지정 「한국항공대학교 인터넷정보검색연구센터」의 연구비 지원으로 수행되었음.