

논문 2011-48SP-3-12

ASIP을 위한 움직임 추정 전용 연산기 구조 및 명령어 설계

(Motion Estimation Specific Instructions and Their Hardware Architecture for ASIP)

황 성 조*, 선우 명 훈**

(Sung Jo Hwang and Myung Hoon Sunwoo)

요 약

본 논문은 H.264나 MPEG4등, 다양한 영상압축 코덱을 지원할 수 있는 ME ASIP의 전용 IME 명령어와 재구성 가능한 하드웨어 구조를 제안한다. 제안하는 전용의 명령어와 하드웨어 가속기는 HD급의 고화질 영상을 지원할 수 있는 성능을 가지고 있다. 제안하는 IME명령어는 다수의 병렬 연산과 패턴 정보를 이용한 연산기 제어를 통하여 전역탐색을 비롯한 각종 고속 탐색 알고리즘을 지원한다. 제안한 하드웨어 구조는 256개의 Processor Elements로 구성되어 있는 Processor Element Group (PEG) 하나당 77,860 게이트를 가진다. 16개의 PEG로 구성된 ASIP은 160MHz의 동작 주파수를 가지고 있으며, HD급 1080p의 해상도를 가지는 영상을 실시간으로 동작 시킬 수 있다.

Abstract

This paper presents an ASIP (Application-specific Instruction Processor) for motion estimation that employs specific IME instructions and its programmable and reconfigurable hardware architecture for various video codecs, such as H.264/AVC, MPEG4, etc. With the proposed specific instructions and hardware accelerator, it can handle the real-time processing requirement of High Definition (HD) video. With the parallel operations and SAD unit control using pattern information, the proposed IME instruction supports not only full search algorithm but also other fast search algorithms. The hardware size is 77K gates for each Processing Element Group (PEG) which has 256 SAD PEs. The proposed ASIP runs at 160MHz with sixteen PEGs and it can handle 1080p@30 frame in real time.

Keywords : H.264, video coding, ASIP, reconfigurable coding, motion estimation

I. 서 론

DTV, 휴대폰, PMP와 같은 디지털 멀티미디어 기기들에 대한 관심과 수요가 크게 증가하면서 영상 압축 표준 또한 많은 발전을 하였다. 특히 H.264/AVC^[1]는 Variable Block Matching 기술과 다양한 정화소 및 부

화소 움직임 추정 방법을 이용하여 기존 영상 압축 표준에 비해 높은 효율의 성능을 보여준다^[2~3].

영상 압축 표준은 소프트웨어와 하드웨어로 구현된다. 특히 고해상도 실시간 처리를 하기 위해서는 하드웨어 구현이 필수적이다. 현재까지는 Application Specific Integration Circuit (ASIC) 방식의 하드웨어 구현이 주로 연구 되었다. 복잡한 연산기 구조 때문에 필요한 동작 성능을 가지기 위해서는 특정 알고리즘에 최적화하여 ASIC으로 구현하는 것이 최선의 방식이었다. 하지만 ASIC으로 구현하는 것은 개발 시간과 비용이 많이 들고, 성능 향상을 위한 개선 작업이 어렵다는 단점이 있다.

ASIC의 단점을 개선하기 위해서 Application Specific

* 학생회원, ** 정회원-교신저자, 아주대학교 정보통신대학 전자공학부

(School of Electrical and Computer Engineering, Ajou University)

※ This work 2011-0016671 was supported by Mid-career Researcher Program through NRF grant funded by the MEST and IC Design Education Center(IDEC)

접수일자: 2010년8월16일, 수정완료일: 2010년11월30일

Instruction-set Processor (ASIP) 방식의 구현 방식이 새로운 대안이 되고 있다. ASIP 방식은 특정 응용 분야에 대한 전용의 명령어 셋을 가지고 있는 프로세서를 개발하여, 어셈블리 코드와 같은 프로그램 가능한 언어를 이용하기 때문에 효율적이고 유연한 형태로의 구현이 가능하다^[4].

본 논문에서는 효율적인 연산이 가능하도록 움직임 추정 전용 ASIP을 위한 Sum of Absolute Different (SAD)와 SAD전용 비교기 명령어를 제안한다. 그리고 이들 명령어를 위한 하드웨어 구조를 보여준다.

본 논문의 구성은 다음과 같다. II장에서는 DSP와 기존 ASIP구조에서의Integer Motion Estimation (IME) 구현에 대하여 설명하고 III장에서는 새로운 구조에서의 IME명령어를 제안한다. IV장에서는 제안하는 명령어의 하드웨어 구조에 대해서 설명한다. V장에서는 기존의 ASIP과 성능을 비교한다. 마지막으로 VI장에서 결론을 맺는다.

II. 기존 프로세서 구조

1. 기존 DSP 구조

표 1은 일반적인 DSP명령어를 이용한 SAD 연산 수행 방식이다. 일반적인 명령어로 1 픽셀에 대한 SAD 연산을 수행하기 위해서는 DSP 명령어로 3 사이클의 명령어가 필요하다. 움직임 추정 연산에서는 하나의 매크로 블록을 처리하기 위해서 256픽셀 이상의 SAD 연산이 필요하기 때문에 연산 부하를 줄이기 위해 TMS320C6X에서는 그림 1에서와 같은 명령어를 제안하였다^[5]. 그림 1은 4개의 픽셀에 대한 SAD연산을 3 사이클의 명령어를 이용하여 수행한다. 하지만 기본 처리단위인 16x16 매크로 블록에 대한 연산을 하기 위해서는 여전히 많은 64번의 명령이 필요하다. 예를 들어 -32~32의 탐색창을 갖는 1080p의 HD급 영상을 전역 탐색으로 처리할 때 2,139,095,040번의 SAD연산을 33.3ms안에 처리해야 한다. 위의 성능을 만족시키기 위

표 1. 일반적인 DSP의 SAD 명령어
Table 1. SAD instruction of general DSP.

동작
SUB R0, R1, R2
ABS R2
ADD R2, R3, R3

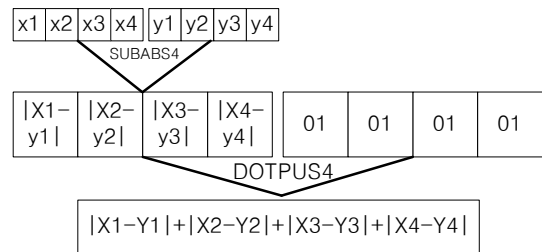


그림 1. TMS320C6X의 SAD 명령 구조
Fig. 1. SAD architecture of TMS320C6X DSP.

해서는 프로세서의 동작속도가 64GHz이상이 되어 하므로 소모되는 전력이 커지고, 사이즈도 ASIC보다 크기 때문에 매우 비효율적이라는 것을 알 수 있다.

2. 기존 ASIP구조

앞서 연구된 ASIP^[6]은 4x4블록에 대한 SAD연산을 할 수 있는 명령어를 제안하고 있다. 이 논문의 특징은 VBM 블록에서 가장 작은 블록 단위를 이용하여 16x16블록 모드까지 지원이 가능하다는 점이다. 그래서 16x16모드에 대한 결과를 얻기 위해서는 16번의4x4 SAD 명령어의 반복이 필요하다. 위의 명령어를 이용하여 -32~32 탐색창의 1080p의 HD급 영상을 전역 탐색으로 처리하기 위해서는 534,773,760번의 연산을 필요로 하기 때문에 약 16GHz의 동작 속도를 유지해야 한다. 하지만 실제 동작 속도는 16GHz에 훨씬 못 미치기 때문에 Enhanced Predictive Zonal Search 등과 같은 탐색 지점을 줄여서 탐색하는 알고리즘에 최적화되었다.

또 다른 ASIP^[7]에서는 16x16블록단위의 하나의 탐색 지점에 대한 연산을 수행하는 명령어를 제안하였다. 그래서 앞선 ASIP보다는 더 많은 SAD연산을 수행할 수 있지만, 16x16, 8x8모드에 대한 블록 모드만 지원한다. 또 다른 구조적인 특징은 재구성 가능한 하드웨어를 지원한다는 점이다. 그래서 알고리즘에 따른 연산량의 변화에 맞추어 하드웨어 크기와 속도의 조절할 수 있다. 하지만 HD급 전역탐색의 경우 연산 지점이 많기 때문에 탐색지점을 줄여 탐색하는 고속 탐색에 대해서만 지원 가능한 단점이 있다.

III. 제안하는 IME 명령어 구조

본 절에서는 효율적으로 IME를 처리하기 위한 ASIP 전용의 명령어를 제안한다. HD급의 높은 해상도를 처

리하기 위해서 빠르고 많은 연산을 병렬 처리할 수 있는 구조가 필요하다. 그리고 명령어는 다양한 패턴을 지원하는 구조로 정의 되어야 한다.

본 절에서는 효율적으로 SAD 연산을 수행하는 명령어를 제안하고, SAD 명령어 전용의 비교기를 제안하여 위와 같은 문제점을 해결하였다. 그리고 프로그램 가능성이 높기 때문에 H.264/AVC 뿐만 아니라 MPEG4와 같은 다른 영상 표준에서도 적용이 가능하다. 본 논문은 PEG를 다이아몬드 탐색에서는 4개, 헥사곤 탐색에서는 6개를 사용해야만 성능을 만족시키는 것을 볼 수 있다. 또한 제안한 구조는 2가지 모드만 지원하는 논문^[7]과는 달리, 모든 모드의 블록이 지원이 가능하다.

1. VBSAD 명령어

제안하는 Variable Block SAD (VBSAD) 명령어는 탐색 시작 지점과 패턴 정보, 연산 크기를 입력 받아서 다양한 패턴과 알고리즘에 적합하게 수행할 수 있도록 정의하였다. 기존 ASIP용 SAD 명령어는 한 지점에 대한 SAD연산을 수행하도록 정의되어 있는 반면에 제안하는 VBSAD 명령어는 최대 256 (16x16) 지점에 대한 SAD연산을 수행할 수 있도록 제안하였다. -32~32 탐색창의 1080p의 HD급 영상을 전역 탐색으로 처리하기 위해서는 522,240번의 SAD연산이 필요하고, 15MHz의 동작 속도를 가지고 실시간 처리가 가능하다. 가장 처리 지점이 많은 전역탐색을 실시간 처리가 가능하기 때문에 전역 탐색보다 탐색 지점이 적은 고속 탐색 알고리즘은 충분히 적용이 가능하다.

다양한 탐색 패턴을 이용한 고속 탐색 알고리즘을 효율적으로 구현하기 위해서는 명령어 내에서 다양한 탐색 패턴을 지원해야 한다. 하나의 알고리즘에서 여러 가지 탐색 패턴을 사용하기 때문에 탐색 패턴을 저장하는 여러 개의 전용의 레지스터를 할당하고, 각 패턴 정보를 전용 레지스터에 저장하여 사용하도록 정의하였다.

고속 탐색 알고리즘의 경우, 고정된 시작 지점을 가지지 않고 참조 영상 내의 임의의 지점에서 탐색을 시작할 수 있어야 한다. 이 문제를 해결하기 위해서 명령어 내에 연산 시작 지점을 입력 받아 원하는 지점에서 연산이 수행할 수 있도록 제안하였다. 시작 지점에 대한 정보를 입력 받으면 참조 데이터 메모리에서 원하는 지점에 대한 픽셀 데이터를 입력해 주고, 입력 받은 데이터에 한하여 연산을 수행한다.

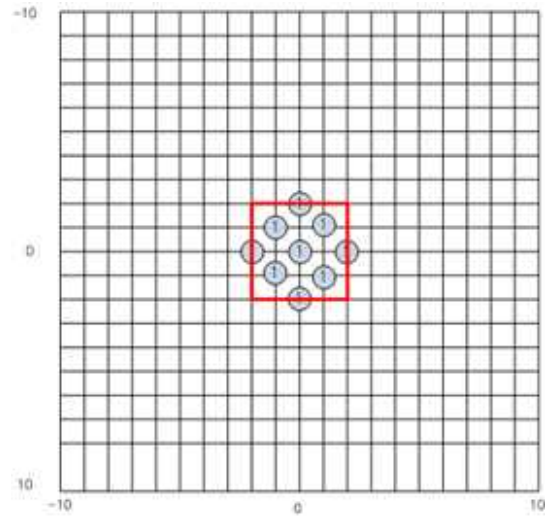


그림 2. 제안하는 SAD 명령어의 예

Fig. 2. Example of proposed SAD instruction set.

VBSAD 명령어의 연산 수행은 다음과 같이 진행된다. 외부 참조 영상 메모리에서 시작지점에 대한 참조 영상 데이터를 입력 받은 후에, 참조 영상에서의 위치를 탐색 범위와 비교하여 연산을 수행 하는지 여부에 대해 판단한다. 그리고 패턴 정보에 따라 다시 연산 수행 여부에 대해서 판단을 하고 이러한 동작을 VBSAD 명령의 탐색 범위 내에서 수행이 끝날 때까지 반복한 후에 연산을 마치게 된다. 그림 2는 제안하는 VBSAD 연산의 예시이다. 그림에서 보여준 탐색 패턴을 지원하기 위해서 위에 표시한 5x5 사이즈의 정사각형의 범위를 설정하고 패턴 대로 패턴 레지스터에 정보를 입력하여 패턴을 적용시킨다. 탐색 시작 지점은 (-2,-2)의 지점이다.

제안한 VBSAD 명령어에 의한 결과 값은 Variable Block Matching (VBM) 기술을 적용하기 위해서 16x16, 8x8, 4x4 와 같이 3가지 모드의 연산 결과를 가진다. 3가지 모드에 대한 결과 값은 움직임 벡터 값과 함께 SAD 메모리에 전송된다. VBM을 지원 하기 위해서는 7가지 모드 (16x16, 8x16, 16x8, 8x8, 4x8, 8x4, 4x4)를 모두 만족해야 하지만, 연산 부하를 경감시키고 프로그램 가능성을 높이기 위해서 SAD 명령어에서는 16x16, 8x8, 4x4 모드에 대해서만 연산을 수행한다. 그리고 이외의 모드에 대해서는 SADCMP 명령어에서 연산이 수행되도록 뒤에서 설명 할 예정이다.

2. SADCMP 명령어

제안하는 SAD specific compare (SADCMP) 명령어

는 기존에 일반적인 compare 명령어와는 다르게 오로지 SAD 결과값 비교하여 최적 지점을 선택하는 SADCMP 명령어이다. 그리고 사용자가 원하는 모드만 연산이 되도록 조건을 입력 받아 특정 블록 모드에 대한 연산이 가능한 구조로 정의되어 있다. SAD 결과값이 저장된 메모리에서 4x4, 8x8, 16x16 모드의 SAD 값과 움직임 벡터를 입력 받아 나머지 모드(8x16, 16x8, 4x8, 8x4)에 대한 SAD값을 생성한다. 모든 모드에 대한 SAD값이 생성이 완료 되면 각 모드 별로 비교 연산을 수행하여 각 최적의 지점을 선택하게 된다.

IV. 제안하는 명령어의 하드웨어 구조

이번 절에서는 III절에서 제안한 명령어를 수행하는 전용의 하드웨어 가속기를 제안하고, 구조를 설명하려고 한다. 그림 3은 제안하는 하드웨어 가속기의 전체 구조도이다.

패턴 정보와 연산크기, 시작 위치 등의 정보를 이용하여 SAD 연산기의 PEG (Processor Element Group)를 제어하고 움직임 벡터를 생성하도록 회로가 구성되어 있다.

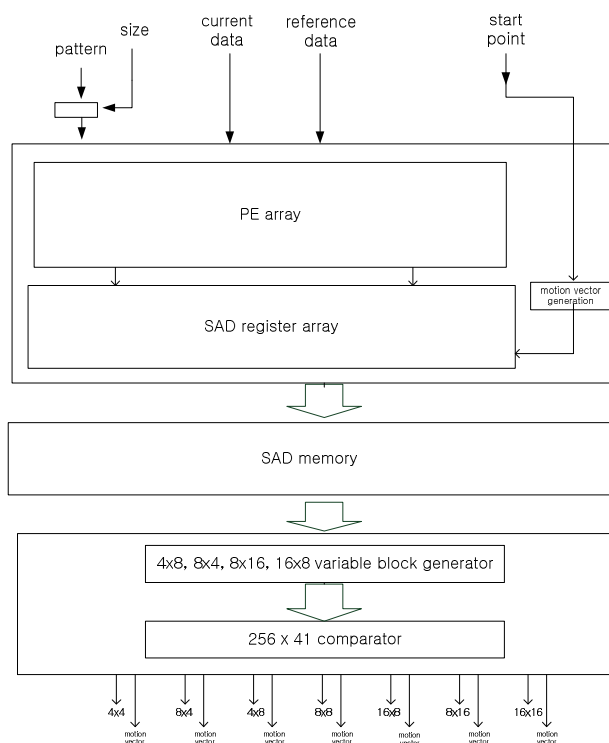


그림 3. 제안하는 IME 구조
Fig. 3. Proposed IME architecture.

1. SAD 구조

SAD 연산기는 연산 효율을 높이기 위해서 2-D SAD 구조를 이용하여 설계하였다. 그림 4는 2-D SAD 연산기의 구조를 보여준다^[9]. 우선 시작 지점에 대한 정보를 바탕으로 외부 참조 영상 메모리에서 필요한 위치의 픽셀 데이터를 입력 받아 참조 영상 레지스터에 저장한다. 그리고 현재 영상 데이터와 참조 영상 데이터는 각 사이클마다 SAD 연산기에 입력을 하여 SAD 연산을 수행한다. 다수의 지점을 연산하기 위해서 다수의 SAD PEG를 구성하여 병렬 연산을 하고 연산 결과를 4x4 모드의 SAD 값으로 재구성한다.

세부적으로는 64개의 4x1에 대한 연산을 수행한 후, 연산결과를 이용하여 4x4 모드에 대한 16개의 SAD 값을 구성한다. 생성된 16개의 4x4모드의 값은 다시 8x8, 16x16모드에 대한 SAD 값을 생성하는데 사용되며, 연산 결과는 출력으로 메모리에 저장한다.

움직임 벡터는 입력 받은 시작 지점에 대한 정보를 이용하여 각 PEG의 연산 결과와 동기화하여 출력되도록 설계하였다. 이렇게 출력된 움직임 벡터는 SAD 연산 결과와 마찬가지로 SAD 전용 메모리에 저장이 된다.

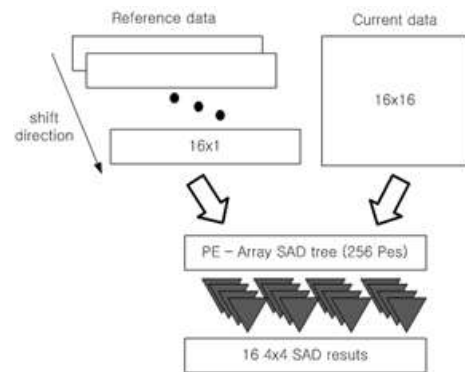


그림 4. 2D SAD 구조도
Fig. 4. 2D SAD architecture.

2. 비교 (compare) 연산 구조

SAD 메모리에서 4x4, 8x8, 16x16 모드에 대한 SAD 값과 움직임 벡터를 이용하여 각 모드 별로 비교를 수행하도록 설계하였다. 하지만 위 세 가지 모드 이외의 모드에 대해 수행할 수 있도록, 먼저 위의 세 가지 모드를 이용하여 4x8, 8x4, 16x8, 8x16 모드에 대한 SAD 값을 생성 시킨다. 하지만 모드 조건 정보를 이용하여, 사용자가 원하는 모드에 대해서만 비교를 하도록 제어할 수 있다.

표 2. 제안하는 ASIP의 스루풋
Table 2. Throughput of proposed ASIP.

algorithm	Throughput (MB/sec)	Freq. (Mhz)
full search (PEG 16)	243,902	160
Diamond search (PEG 1)	428,571	135
Hexagon search (PEG 1)	432,900	135
UMHS (PEG 2)	317,460	160

표 3. 다른 ASIP의 스루풋
Table 3. Throughput of other ASIP.

algorithm	[6]		
	Throughput	Freq. (Mhz)	PEG 수
Full search	-	-	-
Diamond	568,889	135	4
Hexagon	426,667	135	6
UMHS	379,256	135	16

최종 비교 연산은 모드 정보에 의해서 생성된 SAD 값과 움직임 벡터를 이용하여 수행된다. 비교기는 총 7개 모드에 대해서 비교를 수행한다. 16개의 4x4, 8개의 4x8, 8x4, 4개의 8x8, 2개의 8x16, 16x8, 1개의 16x16모드 값을 가지고 있기 때문에 41개의 비교기를 병렬로 구성하여 설계하였다. 다수의 지점에 대한 효율적인 비교 연산을 위해서 다수의 비교기를 두어 병렬로 빠르게 처리할 수 있도록 하였으며, 이는 고화질 영상의 실시간 처리에 유리하다.

V. 결 론

본 논문은 다양한 비디오 압축 알고리즘을 위한 ASIP전용의 IME명령어와 하드웨어 구조를 제안한다. 제안한 IME 명령어는 다양한 알고리즘을 효과적으로 지원하는 SAD, compare와 같은 명령어로 구성되어 있다. 제안된 IME 연산기의 하드웨어 구조는 다수의 SAD 연산기와 1개의 전용 메모리, SAD전용의 비교 연산기로 구성되어 있으며, 일반적인 비디오 용 VLIW 코어나 슈퍼 하버드 구조의 프로세서에 비해 적은 칩 면

적을 가지고, 비디오 시스템에서 더욱 우수한 성능을 보인다.

이전에 제안된 ASIP과의 비교하여 보다 적은 연산기를 가지고 효율적으로 연산이 가능하며 각 종 탐색 알고리즘에 대한 스루풋을 비교하여 제안한 구조에서 HD급 영상의 실시간 처리가 가능하다는 것을 보여주었다.

참 고 문 헌

- [1] Joint Video Team (JVT) of ISO/IEC & ITU-T VCEG, "Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification," ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC, July 2004.
- [2] T. Wiegand, G.J. Sullivan, G. Bjontegaard and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 560-576, July 2003.
- [3] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi, "Video coding with H.264/AVC: Tools, Performance, and Complexity," *IEEE Circuits and Systems Magazine 1*, pp. 7-28, Apr. 2004.
- [4] Jung H. Lee, Sung D. Kim, and Myung H. Sunwoo, "ASIP Instructions and Their Hardware Architecture for H.264/AVC," *JSTS*, vol.5, no.4, pp. 237-242, Dec 2005.
- [5] TMS320C6000 CPU and Instruction Set Reference Guide, Texas Instruments Inc., Dallas, TX, 2000
- [6] T. Dias, S. Momcilovic, N. Roma, and L. Sousa, "Adaptive motion estimation processor for autonomous video devices", *EURASIP Journal on Embedded Systems*, Vol. 2007, pp. 1-10, Jan 2007.
- [7] S. Momcilovic, N. Roma, and L. Sousa, "An ASIP approach for adaptive motion estimation on AVC", in *Proceedings of the IEEE 3rd Conf on Ph.D. Research in Microelectronics and Electronics (PRIME 2007)*, Bordeaux, France, July 2007, pp. 165-168.
- [8] J. L. Nunez-Yanez, E. Hung, and V. A. Chouliaras, "A configurable and programmable motion estimation processor for the H.264 video codec," in *International Conference on Field Programmable Logic and Applications*, Sept. 2008, pp. 149-154.
- [9] T. C. Chen, and et al., "Analysis and Architecture Design of an HDTV720p 30

Frames/s H.264/AVC Encoder,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no.6, pp. 673-688, June 2006. boundary discontinuity criterion, *IEEE Trans. Circuits and Systems for Video Tech.*, Vol. 8, no. 3, pp. 345-357, June 1998.

— 저 자 소 개 —



황 성 조(학생회원)
 2008년 아주대학교 정보통신대학
 전자공학 학사 졸업.
 2010년 아주대학교 정보통신대학
 전자공학 석사 졸업.
 2010년 현재 삼성전자(주) 연구원

<주관심분야 : H.264, ASIC, ASIP, 영상처리>



선우 명 훈(정회원)-교신저자
 IEEE Fellow
 1980년 서강대학교 전자공학 학사
 졸업.
 1982년 한국과학기술원 전자공학
 석사 졸업.
 1982년~1985년 한국전자통신
 연구소(ETRI) 연구원
 1985년~1990년 Univ. of Texas at Austin
 전자공학 박사.
 1990년~1992년 Motorola, DSP Chip Division
 (미국)
 1992년~1996년 아주대학교 전기전자공학부
 조교수.
 1996년~2001년 아주대학교 전자공학부 부교수
 2001년~현재 아주대학교 전자공학부 교수
 2010년~현재 대한전자공학회 반도체 소사이어티
 수석 부회장
 2009년~현재 IEEE CASS Distinguished
 Lecturer
 2011년 IEEE CASS Board of Governors
 <주관심분야 : SoC 설계, VLSI Architecture, 통
 신 및 멀티미디어 ASIP 설계, 저전력 설계>