

논문 2009-46SC-4-3

# Active Star를 이용한 FlexRay 네트워크 구현

## (Implementation of FlexRay Network using Active Star)

장 인 곁\*, 전 창 하\*, 이 재 경\*, 정 진 균\*\*

(In-Gul Jang, Chang-Ha Jeon, Jae-Kyung Lee, and Jin-Gyun Chung)

### 요 약

FlexRay는 차량의 성능향상과 많은 전자제어 유닛으로 인해 발생하는 배선 및 성능저하를 개선하기 위한 고성능 통신 시스템 네트워크의 표준이다. 최대 10Mbps의 데이터 전송 속도를 가지며 2개의 채널을 통해 동시에 전송할 수 있다. FlexRay 시스템을 하드웨어로 구현하기 위해 SDL로 설계된 결과를 토대로 Verilog HDL을 이용하여 설계하였고, Magna/Hynix 0.18 um 공정을 이용해 Synopsys Design Compiler를 사용해 합성하였다. FlexRay 시스템에서 여러 노드 사이의 통신 제어를 위해 active star와 passive star가 사용되는데 active star는 최대 10Mbps의 빠른 데이터 전송에 주로 쓰인다. 본 논문에서는 active star를 이용한 FlexRay 네트워크를 구현하기 위해 설계한 FlexRay 시스템 외에 active star를 제어하기 위한 별도의 controller를 구현하였고 하나의 송신노드에 두 개의 수신노드를 연결하여 각각의 수신노드에 다른 프레임 메시지를 전송하는 실험을 통하여 올바른 동작이 이뤄짐을 확인하였다.

### Abstract

FlexRay is a new standard of network communication system which provides solutions to the degradation problems generated by many ECU (Electronic Control Unit) connections in automobiles and automation systems. The upper bound of the data rate is 10Mbps and it provides two channels for redundancy. In this paper, FlexRay system is first designed using SDL. For hardware implementation, FlexRay system is designed using Verilog HDL based on the SDL design result. The designed system is synthesized using Synopsys Design Compiler with the Magna/Hynix 0.18 um cell library. In this paper, to construct a FlexRay network, active star is used since active star systems can provide high speed data transmission up to 10Mbps. The performance of the star network is tested using one transmitter node and two receiver nodes.

**Keywords :** FlexRay, Communication Controller, Active Star, Automotive

### I. 서 론

차량의 성능향상과 운전자의 편의성 및 안전성에 대한 요구가 증가함에 따라 X-by-wire 기술은 가까운 미래의 지능형 자동차 산업에 큰 변혁을 일으킬 요소로 성장하고 있다. 또한 차량 내 전자 제어 유닛

(Electronic Control Unit, ECU)이 많아지면서 배선 무게가 증가되고 이로 인한 호환문제 및 성능저하 등 문제가 발생한다. 이에 따라, 최근에는 차량의 전자제어 유닛 시스템에 통신 네트워크를 접목시킨 차량 내 네트워크 (In-Vehicle Networking, IVN)가 활발히 연구되고 있다. 현재까지 차량 내 통신은 CAN, LIN 및 J1850 등과 같은 네트워크 통신 프로토콜들이 많이 사용되었지만 날로 늘어나는 ECU 사이의 통신을 수행하기 위해서 FlexRay와 같은 고성능의 네트워크 통신 시스템의 개발과 연구가 필요하다. 그림 1은 차량 내 네트워크 시스템의 발전 동향을 보여준다.<sup>[1]</sup>

FlexRay는 차세대 차량 내 전자 장치간의 통신을 위

\* 학생회원, \*\* 정회원, 전북대학교 전자정보공학부 (Div. of Electronic & Information Engineering Chonbuk University)

※ 본 연구는 지식경제부 및 정보통신연구진흥원의 IT 산업원천기술개발사업(B1120-0901-0002, IT특화연구소 설립)과 2단계 BK21 사업의 지원을 받아 수행하였음.

접수일자 : 2009년6월27일, 수정완료일: 2009년7월20일

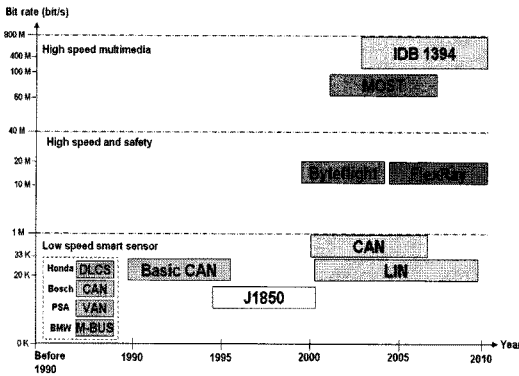


그림 1. 차량 내 네트워크 시스템의 발전 동향  
Fig. 1. Development trend of vehicle network systems.

해 고속의 시리얼 통신, time triggered bus, faulttolerant 통신을 제공하는 새로운 네트워크 통신 시스템의 표준이다.<sup>[2]</sup> FlexRay는 time division multiple access (TDMA) 방식과 부가적인 flexible time division multiple access (FTDMA) 방식을 지원하며 최대 데이터 전송 속도는 10 Mbps이고 2개의 채널을 통해 동시에 전송할 수 있다.<sup>[3]</sup>

본 논문에서는 SDL(Specification and Description Language)을 이용하여 FlexRay 프로토콜 규격과 기능 부분을 설계하고 설계한 SDL을 기반으로 Verilog HDL을 이용하여 하드웨어로 설계하였다.<sup>[4-5]</sup> 또한 설계된 FlexRay와 active star를 이용하여 FlexRay 시스템 네트워크를 구현하였다. FlexRay 시스템의 동작을 확인하기 위해 상용 transceiver 칩을 사용하였고 transceiver 칩을 구동하기 위한 제어회로를 설계하여 FlexRay 네트워크 시스템을 구성하였다.

본 논문의 구성은 다음과 같다. I 장의 서론에 이어 II 장에서는 FlexRay 구조를 간략하게 살펴보고 III 장에서는 SDL과 Verilog HDL을 이용한 FlexRay 시스템 설계 결과를 제시한다. IV 장에서는 여러 노드 사이의 통신 네트워크를 구성하는 star 대하여 설명한다. 마지막으로 V 장에서는 결론을 맺는다.

## II. FlexRay 시스템 구조

### 1. FlexRay 통신 구성

FlexRay 통신은 기본적으로 ECU에서 다른 ECU로 데이터를 전송한다. 그림 2와 같이 ECU\_1에서 메시지를 전송하면 FlexRay 송신단 (FlexRay Tx)에서 이리

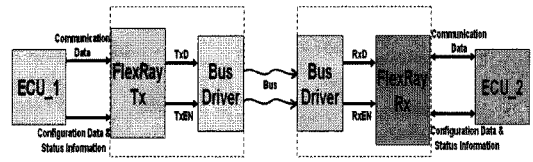


그림 2. FlexRay 통신 네트워크의 인터페이스  
Fig. 2. Interface of FlexRay communication network.

한 데이터를 입력받고 FlexRay 통신 시나리오에 맞게 프로세싱하고 송신단의 버스 driver로 데이터를 전송한다. FlexRay 버스 driver에서 이러한 디지털 시그널을 입력 받고 버스로 데이터를 증폭시켜 전송한다. 수신단의 버스 driver에서는 버스에서 입력되는 시그널을 디지털 시그널로 다시 전환시켜 FlexRay 수신단 (FlexRay Rx)으로 데이터를 입력 시킨다. FlexRay 수신단에서는 송신단에서 송신된 시그널을 프로세싱하여 ECU\_2로 전송시켜 최종적으로 ECU\_1에서 ECU\_2로 FlexRay 통신을 구성한다.

### 2. FlexRay 프레임 형식

FlexRay 프레임은 그림 3과 같이 header segment, payload segment, trailer segment로 이루어진다. 프레임의 처음 5 bits는 프레임의 기본 정보를 포함하고 있다. Frame ID(11 bits)는 ST(static segment)에서의 slot 위치를 나타내고, DYN(dynamic segment)에서는 프레임의 우선순위(낮은 ID 값이 더 높은 우선순위를 가진다)를 나타내기 위해 사용된다. Payload Length(7 bits)는 데이터의 길이(payload length x 2 = data bytes 의 길이)를 나타낸다.

Header CRC(11 bits)는 Sync Frame Indicator(1 bit), Startup Frame Indicator(1 bit), Frame ID(11 bits), Payload Length(7 bits)에 대해 계산한 Cyclic

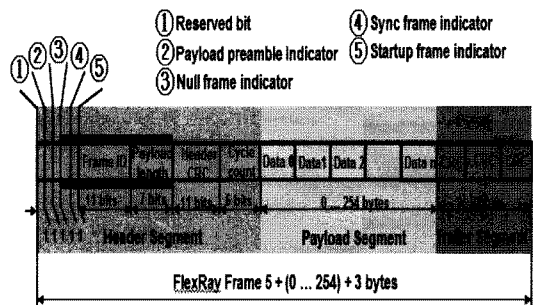


그림 3. FlexRay 프레임 구조  
Fig. 3. FlexRay Frame structure.

Redundancy Check이다. Cycle Count(6 bits)는 각 노드의 frame 순번이다. Payload Segment(0~254 bytes)에는 실제 데이터가 포함된다. Trailer Segment(24 bytes)는 header segment와 payload segment에 대해 계산한 Cyclic Redundancy Check이다.

### III. SDL과 Verilog HDL을 이용한 FlexRay 설계

#### 1. SDL을 이용한 설계

SDL(Specification and Description Language)은 ITU Telecommunication Standardization Sector (ITU-T)에서 Z.100으로 표준화 된 추상 시스템 디자인 언어이다.<sup>[6]</sup>

본 논문에서는 SDL을 이용하여 FlexRay CC와 BG 시스템의 각 블록을 블록별로 상세설계하고 MSC 시뮬레이션 과정을 통하여 기능을 검증하였다. 또한 FlexRay 통신 프로토콜에서 함수로 표현된 부분을 FlexRay 통신 원리에 맞게 설계하고 해당 블록에 추가하여 검증하였다. 모든 sub 블록을 설계한 후 하나의 top 시스템으로 조합하여 각 블록사이의 인터페이스를 조정하고 시뮬레이션 과정을 거쳐 FlexRay 시스템의 통신 시나리오대로 작동하는지를 검증하였다.

#### 2. Verilog HDL을 이용한 설계

SDL로 설계한 FlexRay 소스를 기반으로 Verilog HDL로 설계하고 FlexRay 시스템을 각 블록별로 타이밍 시뮬레이션을 수행하여 기능과 성능을 검증하였다. 우선 각 블록에서 모든 상태 변환의 경우에 따라 시뮬레이션 하여 모든 상태의 기능을 검증하였다. 각 블록별로 타이밍 시뮬레이션 과정을 수행한 후 블록과 블록을 서로 조합하여 타이밍 시뮬레이션을 수행하여 MSC 시뮬레이션 결과와 비교하여 FlexRay 통신 시나리오대로 실행되는지를 검증하였다.

그림 4는 CODEC 블록에서 static segment의 프레임 인코딩 과정을 보여준다. TransmitframeonA 시그널은 프레임의 전송시작을 의미하고 vTF\_A 시그널은 인코딩 하려는 프레임의 데이터를 표시하며 vType 시그널은 전송하려는 프레임이 ST인지 DYN인지를 구분하는 시그널이다. TxID\_A 시그널은 프레임 데이터를 시리얼로 출력하는 과정을 보여주고 TxEN\_A 시그널은 TxID\_A 시그널이 출력하기 시작할 때 '0'으로 출력되고 TxID\_A 시그널이 출력을 마칠 때 '1'로 출력되어 프레임

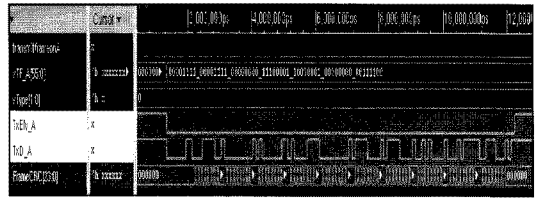


그림 4. ST 프레임 인코딩의 시뮬레이션 결과  
Fig. 4. Simulation result of ST frame encoding.

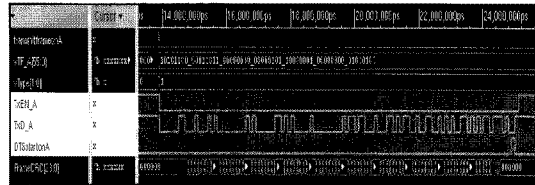


그림 5. DYN 프레임 인코딩의 시뮬레이션 결과  
Fig. 5. Simulation result of DYN frame encoding.

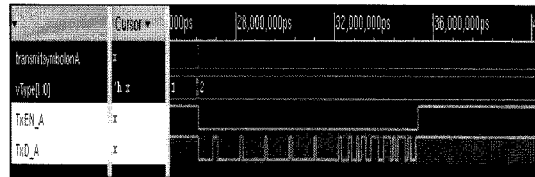


그림 6. Symbol window 인코딩의 시뮬레이션 결과  
Fig. 6. Simulation result of symbol window encoding.

입 출력 시그널의 상태를 나타낸다.

그림 5는 DYN의 프레임 인코딩의 타이밍 시뮬레이션 결과를 보여준다. ST와 DYN의 인코딩은 비슷한 원리로 수행 되지만 DYN의 인코딩 과정에서는 DTS 시그널이 추가로 출력되어 ST의 인코딩 과정과 구분된다.

그림 6은 symbol window의 인코딩 시뮬레이션 결과를 보여준다. TransmitsymbolonA 시그널은 symbol window의 인코딩 시작을 의미하고 symbol window의 인코딩 과정을 수행한 후 network idle time 과정이 수행되어 하나의 FlexRay 통신 사이클 주기의 인코딩을 마친다.

설계한 FlexRay 통신 시스템을 Magna/Hynix 0.18um CMOS 표준 공정을 이용하여 합성하였다. 합성 결과 총 199만 개의 셀로 합성되었고 전력 소모량은 14 mW이었다.

### IV. Star를 이용한 FlexRay 네트워크

차량 및 로봇 내 시스템에서 FlexRay 버스에 연결된

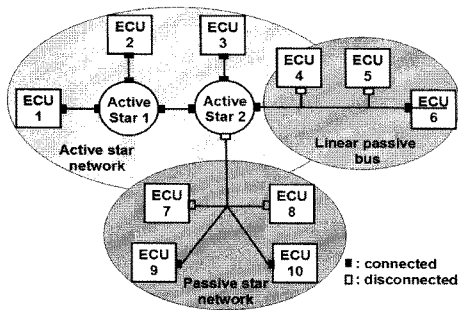


그림 7. Star를 이용한 FlexRay 네트워크 구성  
Fig. 7. FlexRay Network using star.

모든 노드사이의 통신은 point to point 방식으로 수행한다. FlexRay 시스템에서는 여러 노드사이의 통신제어를 위해 star를 사용하고 송신노드의 각 슬롯에 해당된 프레임 메시지는 star를 거쳐 각 수신노드로 전송된다.

FlexRay star에는 active star와 passive star 등 두 가지 종류의 star가 있다. Active star는 상용 transceiver 칩을 응용하여 설계하며 최대 10 Mbps의 빠르고 정확한 데이터 전송을 할 때 사용하므로 FlexRay 통신 시스템에는 주로 active star를 사용한다. Passive star는 상용 transceiver 칩을 사용하지 않고 FlexRay 시스템의 버스를 수동으로 차단하는 방법으로 주로 1Mbps 미만의 데이터를 연결할 때 사용할 수 있다.

그림 7은 star를 이용한 FlexRay 네트워크 구성을 보여준다. ECU 1, ECU 2, ECU 3, ECU 6는 active star 네트워크로 구성되었고 active star1 과 active star 2는 버스에 연결되어 있으며 모든 ECU는 active star에 연결되어 있다. ECU와 ECU 사이의 통신은 미리 정해진 통신 스케줄에 따라 active star를 제어하는 방식으로 통신을 수행한다. ECU 7, ECU 8, ECU 9, ECU 10은 passive star 네트워크로 구성되었으며 passive star는 실제로 ECU 7과 ECU 8 사이의 교차점에 존재한다. 이러한 active star와 passive star를 제어하기 위한 controller는 FlexRay 네트워크를 구성할 때 미리 설정한 스케줄에 따라 제어하므로 별도의 controller가 필요하다.

### 1. FlexRay에서 쓰이는 Active Star

FlexRay 용 active star는 상용 transceiver 칩을 이용하여 구현한다. CAN의 transceiver 칩은 데이터가 입력되면 즉시 BP (Bus Plus)와 BM (Bus Minus)으로

바로 증폭되어 출력하지만 FlexRay 용 transceiver 칩은 크게 low power mode와 normal mode로 나눈다. Low power mode는 standby, sleep, go to sleep, power on standby 등 4가지 모드로 나눈고, normal mode는 normal 모드와 receive only 모드로 나눈다. 데이터를 전송하지 않을 때에는 transceiver 칩 상태를 low power 모드로 세팅하여야하고 데이터를 전송할 때에는 normal mode로 세팅하여야 한다.

이러한 FlexRay 용 트랜시버 칩을 응용하기 위하여 송신하려고 하는 시그널(TxD)의 이벤트 시그널(TxEN)과 기타 제어 시그널을 이용하여 별도의 controller를 설계하여 제어한다.

### 2. Active Star를 이용한 여러 노드 사이의 통신

FlexRay 통신 시스템에서 여러 노드사이의 통신을 수행하려면 반드시 star가 필요하다. 본 연구에서는 기존에 상용 transceiver를 이용하여 설계한 bus driver를 active star로 대체하여 여러 노드사이의 통신을 수행하였다. Active star를 제어하기 위하여 별도의 controller를 구현하였으며 active star를 제어하여 하나의 송신노드에서 두 개의 연속된 프레임을 출력하고 두 개의 수신노드에서 각각 프레임 정보를 받는 실험을 하였다.

그림 8은 송신하려고 하는 두 개의 프레임이 있으며 slot 1에는 frame ID가 '01'인 프레임이 있고 slot 2에는 frame ID가 '10'인 프레임이 있음을 표시한다. 송신노드 Tx에서 이러한 두 개의 프레임을 시리얼로 송신하고 수신노드 Rx\_1에서는 'frame 01' 만 수신하고 수신노드 Rx\_2에서는 'frame 10' 만 수신한다고 가정한다. FlexRay 통신은 각 노드에서 통신을 수행하기 전에 미리 네트워크를 구성하고 파라미터를 선언하는 등 스케줄링을 수행한다.

그림 9의 FlexConfig 소프트웨어에서 미리 각 노드에 수행에 필요한 시그널과 파라미터 값을 FlexCard를 통하여 각 노드에 입력한다. Tx 노드는 송신노드의 스케줄에 맞게 작업을 수행하고, Rx1 노드는 송신노드에서 전송된 첫 번째 slot에서 입력되는 메시지를 수신하

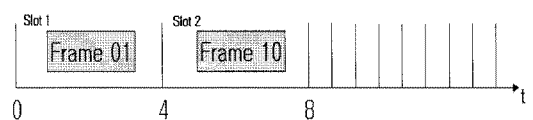


그림 8. Slot에 할당된 프레임의 구조  
Fig. 8. Allocated Frame structure by slot.

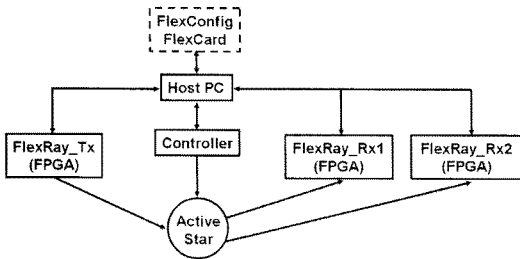


그림 9. 테스트 환경의 블록 다이어그램  
Fig. 9. Block diagram of test environment.

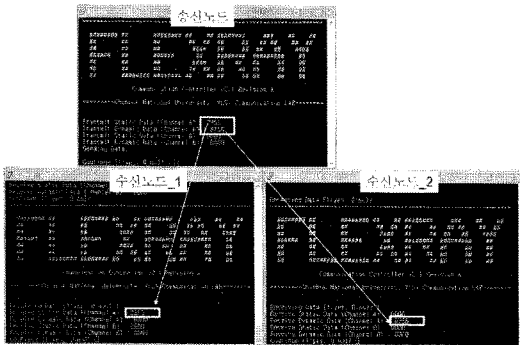


그림 10. Host PC의 프레임 입출력 화면  
Fig. 10. Frame in/out picture of Host PC.

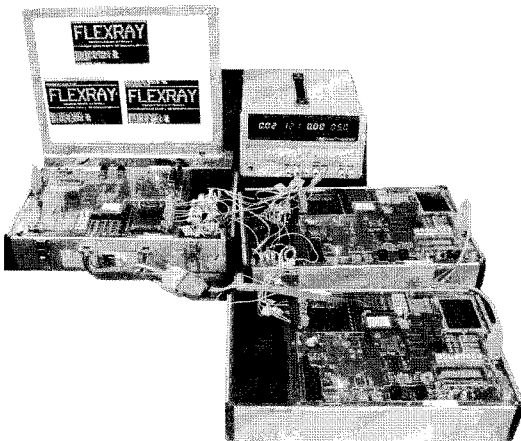


그림 11. FlexRay active star 네트워크의 실제 테스트 환경  
Fig. 11. Actual test environment of FlexRay active star network.

기위해 스케줄에 맞게 작업을 수행한다. Rx2 노드도 Rx1과 유사하게 작업을 수행한다. 또한 transceiver chip controller는 송신노드의 첫 번째 slot에서 입력되는 메시지를 전송 하기위해 윗부분의 두 transceiver칩을 활성화 시켜 첫 번째 slot에 할당된 frame ID '01'의

프레임이 Rx1 노드로 입력되어 통신을 수행하고 끝나면 비활성화 시켜 통신을 정지시킨다. 두 번째 slot에서 메시지가 입력될 때에는 아랫부분의 두 transceiver 칩을 활성화 시켜 두 번째 slot에 할당된 frame ID '10'의 프레임이 Rx2 노드로 입력되어 통신을 수행하고 끝나면 비활성화 시켜 통신을 정지시킨다.

그림 10은 Host PC의 프레임 입출력 화면을 보여준다. 송신노드에서 static 부분에 2353, dynamic 부분에 8756의 데이터가 입력되면 수신노드\_1에는 2353만 수신되고 수신노드\_2에는 8756만 수신됨을 보인다. 그림 11은 실제 테스트 환경을 보여준다.

### V. 결 론

본 논문에서는 차세대 차량 및 로봇 내 네트워크 시스템에서 많은 사용이 예상되는 FlexRay의 구현 및 active star를 이용한 FlexRay 통신 네트워크 구성에 관해 소개하였다.

FlexRay 통신 시스템을 하드웨어로 설계하기 위하여 SDL 소스를 기반으로 하여 Verilog HDL을 사용해 FlexRay 통신 시스템을 설계하였다. 설계한 시스템은 Synopsys Design Compiler로 Magna/Hynix 0.18 um cell library를 이용하여 합성하였다. 합성한 결과 조합 회로는 약 199만 개의 게이트로 구성되며 14 mW의 전력을 소모하였다.

또한 active star를 이용한 FlexRay 통신 네트워크를 구축하기 위하여 상용 Transceiver 칩을 이용하였다. 실험에서 하나의 송신노드와 두 개의 수신노드를 연결하여 각각의 프레임 ID가 다른 두 개의 메시지를 송신노드에서 송신하고 수신노드에서 이에 상응하는 메시지를 수신하는 것을 확인하였다.

### 참 고 문 헌

- [1] G. Leen and D. Hefferman, "Digital networks in the automotive vehicle," *IEEE Computer and Control Engineering Journal*, vol. 10, no. 6, pp. 257-266, Dec. 1999.
- [2] FlexRay Communications System - Protocol Specification, v2.1 Revision A, FlexRay Consortium, Dec. 2005.
- [3] Dominique Paret. *Multiplexed Networks for Embedded Systems*. John Wiley & Sons, Ltd, 2007.

- [4] ITU-T Z.100, CCITT Specification and Description Language (SDL), 1996.
- [5] SDL-Forum Homepage,  
<http://www.sdl-forum.org>.
- [6] ITU-T Z.100, CCITT, Specification and Description Language (SDL), 1996.

---

 저 자 소 개
 

---



장 인 걸(학생회원)  
2005년 전북대학교 전자공학과  
학사 졸업.  
2007년 전북대학교 정보통신  
공학과 석사 졸업.  
2008년~현재 전북대학교 전자정  
보공학부 박사과정

<주관심분야 : 통신, 신호처리, 반도체>



전 창 하(학생회원)  
2009년 전북대학교 전자공학과  
학사 졸업.  
2009년~현재 전북대학교 전자  
정보공학부 석사과정  
<주관심분야 : 통신, 컴퓨터, 신호  
처리, 반도체>



이 재 경(학생회원)  
2009년 전북대학교 전자공학과  
학사 졸업  
2009년~현재 전북대학교  
전자정보공학부 석사과정  
<주관심분야 : 통신, 컴퓨터, 신호  
처리, 반도체, >



정 진 균(정회원)  
1985년 전북대학교 전자공학  
학사 졸업  
1989년 미국 미네소타 주립대학  
전기공학 석사 졸업  
1991년 미국 미네소타 주립대학  
전기공학 박사 졸업

<주관심분야 : 통신, 컴퓨터, 신호처리, 반도체>