

<http://dx.doi.org/10.7236/IIBC.2015.15.6.141>

IIBC 2015-6-20

## FPGA를 이용한 음성 신호 감지 시스템 개발

### Development of Voice Signal Detection System using FPGA

김장원\*

Jang-Won Kim \*

**요약** 다양하게 복합된 소리 및 음성신호를 FPGA의 마이크로 입력받아서 신호를 분류하고 분석하여 이상 신호를 감지할 수 있는 많은 시스템이 있으나, 효율적이며 효과적으로 이상 신호를 감지하는 시스템을 구현하는데 있어서는 많은 문제점들을 가지고 있다. 따라서 이 문제를 해결하고 감지율을 높이기 위하여 본 연구에서 제안된 방법에서는 소리 신호가 입력되는 마이크 센서를 사용하여 FIFO(First-in First-out) 구조에 적용하고, 통계학적으로 분산과 변동계수를 적용한 알고리즘을 기반으로 이상 신호를 효과적으로 분류하고, 효율적으로 감지 여부를 출력하는 시스템을 제안하고 구현하였다. 제안된 알고리즘을 적용한 시스템을 통하여 100회 이상의 실험을 반복한 결과 96.3%의 감지율을 보였다.

**Abstract** In order to classify and analyze variously compounded sound and voice signal from FPGA microphone, there are numerous systems to detect abnormality signal, however, they have a lot of problems to implement the abnormality signal detection efficiently and effectively. Therefore, we proposed a method that implements classifying the signal effectively and outputting the detection efficiently based on the algorithm applied FIFO structure (First-in First-out) by using microphone sensor which able to input the sound signal, and statistical variance and coefficient of variation (CV). The result showed 96.3% detection when the experiment was performed more than 100 times with the proposed algorithm applied system.

**Key Words** : FIFO, Variance and Coefficient of variation(CV), Voice signal detection

## 1. 서론

사물인터넷(IoT: Internet of Thing)<sup>[1]</sup>이나 모바일 스마트 시스템의 활성화에 힘입어 인간의 주거환경이나 생활환경 또는 오피스환경에 대한 감시체계에 대하여 관심이 높아지고 있는 추세이다. 그러나 자칫 과도한 사용으로 인하여 사생활 침해와 인권보호의 사각지대에 놓여 있을 수 있다. 이러한 문제를 효과적으로 줄일 수 있고, 해결할 수 있는 방법은 시스템에 입력되는 다양한 소리로부터 음성신호를 먼저 판별한 후 다른 센서들의 순차

적인 작동을 통하여 문제를 해결할 수 있다.

최근 국토교통부는 법무부와 공동으로 ‘실무자를 위한 범죄예방환경설계(CPTED, Crime Prevention Through Environmental Design) 가이드북<sup>[2]</sup>을 발간하였으며 이는 건물·시설 배치를 통해 범죄자가 쉽게 접근할 수 없도록 하거나 범죄 행위의 노출 가능성을 높이는 설계 방식이다. 그리고 범죄를 예방하기 위한 CCTV는 모니터링<sup>[3][4]</sup>에 한정되어 있다. 왜냐하면 CCTV로 음성까지 녹음하는 것은 통신비밀보호법 상으로도 형사처벌 대상인 동시에 감시의 부정적인 면을 가지고 있기 때문

\*중신회원, 가천대학교 전자공학과  
접수일자: 2015년 11월 2일, 수정완료: 2015년 12월 2일  
게재확정일자: 2015년 12월 11일

Received: 2 November, 2015 / Revised: 2 December, 2015 /  
Accepted: 11 December, 2015

\*Corresponding Author: jwkimm@gachon.ac.kr  
Dept. of Electronic Engineering, Gachon University, Korea

이다.

그러나 본 연구에서 제안된 감시 체계 방법은 음성 신호가 감지되면 음성 정보의 노출 없이 그 음성신호를 읽어 컴퓨터로 보내는 역할을 하므로 녹음, 사생활 보호에 위반되지 않는다. 즉, 본 연구에서는 FPGA<sup>[5]</sup>를 사용한 임베디드 기술<sup>[6]</sup>을 활용하여 음성신호 감지 정보체계를 개발하고자 하였으며, 개발된 모듈을 활용하여 음성신호를 인식하고 감지된 음성신호에 따라 결과를 사용자에게 알리는 감시 정보 시스템이다.

## II. 시스템 구현

### 1. FPGA 기반 신호 검증

본 연구는 Verilog-HDL(Hardware Description Language)을 이용하여 FPGA를 구현하고 음성 신호를 받아서 FIFO 방식을 이용 한 데이터 처리와 함께 이를 분석 하고 UDP 통신 방식을 이용하여 사용자에게 감시 정보를 전달하는 체계이다. 보드에 센서를 추가 할 수 있기 때문에 그 융통성을 위하여 FPGA를 이용 하였고 실시간 데이터 분석을 위하여 FIFO 방식을 구현 하였다. 이를 위한 기술을 구현하기 위한 시스템의 블록 다이어그램은 다음 그림 1과 같다.

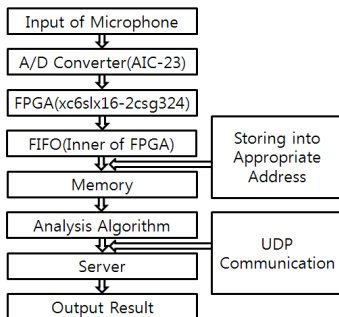


그림 1. 시스템 블록 다이어그램  
Fig. 1. System Block Diagram.

최근 부각되고 있는 음성인식 기술<sup>[7]</sup>을 상용화하기 위한 음성인식 칩, 소프트웨어, 통신용 음성인식 기술을 FPGA를 기반으로 하고 있다. PLD(Programmable Logic Device) 소자는 입출력을 사용자가 정할 수 있고 Verilog-HDL을 이용하여 필요한 설계도 가능한 XC6SLX16-2CSG324를 사용하였다. A/D 변환된 입력신

호를 FPGA로 보내기 위해서 그림 2의 A/D Converter의 데이터시트 분석 후 그림 3과 같이 마이크 데이터를 입력을 받기 위한 비트로 표현해준다.

OS인 Embedded Linux와 FPGA의 연산 속도 차이로 음성 데이터가 실시간으로 입력되는 동안 비교적 느린 출력 함수의 연산이 이루어지면, 그 사이의 음성 데이터에 대한 분석을 놓칠 수 있기 때문에 이를 보완하기 위하여 FPGA에서 FIFO(First-In First-Out)를 구현 하여 FIFO에서 나온 데이터를 배열에 저장 하여 출력하고 분석 하도록 하였다.

Analog Audio Path Control (Address: 0000100)										
BIT	D8	D7	D6	D5	D4	D3	D2	D1	D0	
Function	X	STA1	STA0	STE	DAC	BYP	INSEL	MICM	MICB	
Default	0	0	0	0	1	1	0	1	0	
STA[1:0]	Sidetone attenuation		00 = -6 dB 01 = -9 dB 10 = -12 dB 11 = -15 dB							
STE	Sidetone enable		0 = Disabled 1 = Enabled							
DAC	DAC select		0 = DAC off 1 = DAC selected							
BYP	Bypass		0 = Disabled 1 = Enabled							
INSEL	Input select for ADC		0 = Line 1 = Microphone							
MICM	Microphone mute		0 = Normal 1 = Muted							
MICB	Microphone boost		0=0dB 1 = 20dB							
X	Reserved									

그림 2. A/D Converter의 마이크 입력 모드  
Fig. 2. Microphone Input Mode of A/D Converter.

```

7'b0000100 : begin
    aic_data <= { aic_addr[6:0], 9'b1_0001_0000 };
    send_start <= 1'b1;
    aic_addr[6:0] <= 7'b0000101;
    wait_aic_load <= 1;
end
    
```

그림 3. FPGA Tool에서 마이크 입력 표현  
Fig. 3. Presentations of Microphone Input for FPGA Tool.

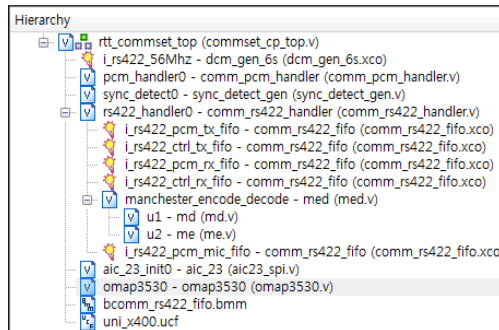


그림 4. FPGA Tool에서 각 소자에 대한 파일  
Fig. 4. Device Files for FPGA Tool.

그림 4에서 보여지듯이 마이크 입력을 해당 레지스터에 저장하기 위해서 mic\_data와 mic\_fifo\_count에 대한 변수를 지정한 후 omap파일에서 해당 어드레스에 그 값을 저장한다.

## 2. FPGA를 이용한 시스템 구현

그림 5는 A/D Converter이며, 그림 6은 타겟보드로 사용한 Xilinx사의 xc6slx16-2csg324이며, Xilinx사의 ISE Ver.14.6에서 컴파일 하였고, 언어는 Verilog-HDL를 사용하였다.

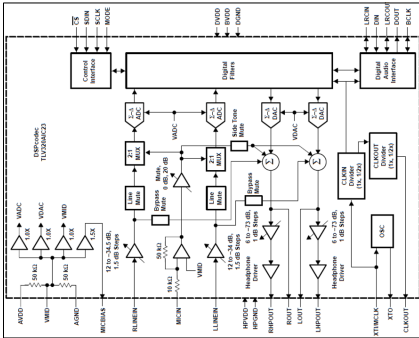


그림 5. A/D Converter의 블록 다이어그램  
 Fig. 5. Block Diagram of A/D Converter.



그림 6. 타겟보드(xc6slx16-2csg324)  
 Fig. 6. Target Board(xc6slx16-2csg324).

## 3. FIFO

타겟보드는 임베디드 리눅스 체제에서 운영 되고 있다. FPGA에서 디지털화 된 음성 신호를 CPU로 보내고, CPU 상에서 분석을 거쳐서 출력 함수를 이용하여 사용자에게 경보를 하는 과정을 거친다. FPGA의 동작 속도가 CPU에서 출력 함수를 거쳐서 출력되는 속도에 비해 월등히 빠르기 때문에 만약 FIFO를 사용하지 않고 FPGA에서 CPU로 데이터를 바로 보내게 된다면 CPU에서 데이터를 분석하고 출력 하는 동안 FPGA로부터 들어오는 데이터는 손실이 될 가능성이 많다. 이는 감지 실패의 가장 치명적인 원인이 될 수 있다. 따라서 FIFO를 사용하면, FPGA에서 데이터를 바로 CPU로 전송하지 않고

원하는 FIFO의 크기만큼 데이터를 쌓은 후 CPU로 전송하는 과정을 반복한다. 분석 및 출력을 함으로써 연산 시간동안의 데이터 손실을 줄여 보다 정확한 결과를 얻어 낼 수 있다.

그림 7에 보여지듯이 FIFO DATA는 32비트의 크기로 설정하고, FIFO COUNT는 4비트의 크기로 설정하여 마이크 입력(pcm\_mic\_fifo)을 구성하였다.

```

comm_rs422_fifo i_rs422_pcm_mic_fifo(
    .rst(!mreset),
    .dout(o32_mic_data),           // Output
    .rd_clk(sys_ref_clk),         // Read Enable
    .rd_en(i_read_mic_en),       // Read FifoCount
    .rd_data_count(o4_mic_fifo_count),
    .din(pcm_tx_data),           // Input
    .wr_clk(pcm_tx_fifo_clk),
    .wr_en(pcm_tx_data_wld),
    .wr_data_count(i)
);
endmodule
    
```

그림 7. FPGA에서 FIFO 매핑 선언  
 Fig. 7. FIFO Mapping Declaration for FPGA.

## 4. UDP 통신을 이용한 정보 전달

구현한 시스템에서는 UDP 통신을 이용하여 리눅스(클라이언트)와 윈도우(서버)간 데이터 정보 교환을 한다. OSI 7 계층의 전송(Transport)계층에 TCP 통신과 함께 속해있는 UDP 통신은 비연결형 통신형이며, TCP 통신과 달리 흐름제어(Flow Control)가 존재하지 않는다. 즉, 송수신하는 데이터의 양은 작으면서 잦은 연결이 필요한 경우에는 UDP 통신이 훨씬 효율적이고 빠르게 동작하기 때문에 본 연구에 적합하여 적용하였다. 그림 8은 UDP의 내부 동작 순서이다.

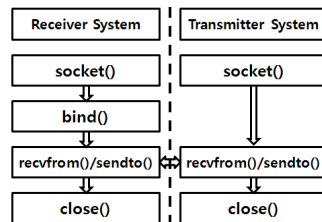


그림 8. UDP 통신 내부 동작 순서  
 Fig. 8. Inner Movement Sequence of UDP Communication.

동작을 할 때 TCP 통신과 달리 read()/write()함수를 사용하지 않고 recvform()과 sendto()함수를 이용하여 자료를 송수신 한다. sendto() 함수를 이용하는 이유는 전송할 목적지를 지정할 수 있기 때문이며, recvform() 함

수를 이용하는 이유는 수신되는 자료 외에도 송신지의 정보를 함께 얻을 수 있기 때문이다. 그러나 UDP 통신의 소켓은 연결 상태를 유지하지 않으므로, 데이터를 전송할 때마다 반드시 목적지의 주소정보(sendto())를 별도로 추가해야 한다.

UDP 데이터는 발신지가 일정하기 않기 때문에 발신지 정보를 얻을 수 있도록 함수가 정의되어 있어야 한다. 즉, recvfrom() 함수는 UDP 패킷에 담겨 있는 발신지 정보를 함께 반환한다. UDP client는 UDP SOCKET을 생성하고, bind() 소켓에 주소 구조체 초기화를 시켜 그림 9와 같이 '주소체계', '포트번호', 'IP주소'를 설정한다. 클라이언트에서 앞서 설명을 하였던 FIFO로 받은 데이터 값을 sendto()와 recvfrom()를 통하여 값을 표현해준다.

```
// Initialization of struct for bind() socket address
serveraddr.sin_family = AF_INET;
serveraddr.sin_port = htons(5000);
serveraddr.sin_addr.s_addr = inet_addr("192.168.0.100");
.
.
.
//Initialization of Winsock
WSADATA wsaData;
WSAStartup(0x0202, &wsaData);
```

그림 9. bind() 소켓 주소 구조체와 원속의 초기화  
Fig. 9. Initializations of Winsock and Struct for bind() Socket Address.

그림 9와 같이 원속(Winsock) 프로그래밍을 할 때에는 반드시 WSAStartup 함수를 호출해서, 프로그램에서 요구하는 윈도우 소켓의 버전을 알리고 해당 버전을 지원하는 라이브러리의 초기화 작업을 진행해야 한다. WSAStartup()는 두 매개 변수가 있다. 먼저 첫 번째 매개 변수는 사용할 소켓의 버전정보를 WORD 형태로 구성해서, wVersionRequested로 전달해야 한다. 위와 같이 0x0202는 master와 slave version이 모두 2이고, 2.2를 뜻하며 인자로 전달을 한다. 두 번째 매개 변수는 &wsaData와 같이 WSADATA 구조체 변수의 주소값을 인자로 전달해야 한다. 그러면 함수 호출이 완료되고 난 다음에 해당 변수에는 초기화된 라이브러리의 정보가 채워진다. 함수 호출을 위해서는 반드시 WSADATA 구조체 변수의 주소 값을 전달해야 한다.

### III. 통계학적 분석 및 실험 결과

A/D Converter(AIC\_23)를 통하여 디지털 신호로 변

환된 음성 신호는 분석을 하여 이상 신호 감지 시 서버에 감지 여부를 송신하여야 한다. 하지만 시스템 주변에서는 작거나 큰 소음이 항상 발생하고 있다. 따라서 작은 소음이 존재하는 환경에서나 큰 소음이 존재하는 환경에서 시간 순으로 들어온 음성신호를 구분해서 분석할 수 있는 알고리즘이 필요하다. 이를 위한 방법 중의 하나인 FFT(Fast Fourier Transform, 고속 푸리에 변환) 기술은 복잡하며 많은 양의 소스 코딩이 필요한 단점이 있으므로, 보다 간단하고 빠른 분산을 이용한 분석 알고리즘을 제안하였다.

#### 1. 분산을 사용한 이상신호 감지 실험

A/D Converter(AIC\_23)에서는 1초 당 4096개의 소리 신호를 디지털 신호로 변환시켜준다. 이 신호는 끊임없이 변화하기 때문에 분석을 하기 위해서는 통계학을 기반으로 한 분산 분석을 적용하였다.

$$S^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2 \quad (1)$$

분산은 자료가 평균으로부터 떨어져있는 정도를 나타내는 수치이며, 지속적으로 4096개로 입력되는 신호의 평균과 분산을 구하여 그 값이 0이 아니라면 특정 신호가 감지되었다고 판단한다. 실험적으로 기계음과 같이 지속적인 소음이 계속되는 환경이 아니라면 분산이 0이 되는 경우는 거의 없다. 따라서 여러 차례의 반복 실험을 통하여 신호의 평균을 여러 단계로 나누어 소음이라 판단되는 신호가 입력되었을 때 분산 값이 평균 단계 각각에 정해진 값보다 클 때 이상신호가 감지되었다고 판단하는 프로그램을 구현하였다. 그리고 정확성을 위하여 여러 환경에서 분산 분석을 하기 위해서는 다중처리를 해야 했기 때문에 간략화 하기 위하여 변동계수를 사용하였다.

#### 2. 변동계수를 사용한 알고리즘

변동 계수(CV : coefficient of variation)는 표준 편차를 산술 평균으로 나눈 것이다. 숫자가 클수록 상대적인 차이가 크다는 것을 의미한다.

$$CV = \frac{\sigma}{x} \quad (2)$$

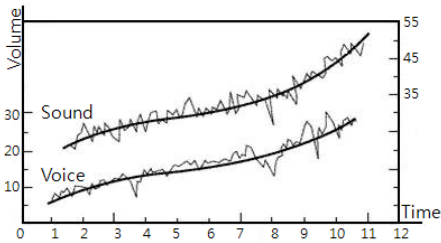


그림 10. CV를 이용한 소음과 음성 분석  
 Fig. 10. Sound and Voice Analysis using CV.

변동 계수를 사용하게 되면서 그림 10과 같이 시간에 따라 변하는 주변 소음과 음성의 크기와 같이 불규칙적인 산포에서도 분산의 크기를 비교할 수 있게 되었다. 또한 변동 계수에 100을 곱한 백분율을 통하여 1초당 받아온 4096개의 신호들은 배열의 평균값의 크기에 관계없이 절대적으로 비교가 가능하다. 따라서 평균의 값에 따라 평균의 단계를 나누어 같은 코드의 반복이 제거되었다.

```

cv = pyojun_pyencha / aver + 100; //변동계수의 백분율.
//변동계수가 25이상일때 평균값에 따라 데이터를 구분하여 전송.
memset(buff, 0, 128);
if(cv > 25){
    if(aver >= 35 && aver < 50){
        buff[0] = 'a';
    }else if(aver >= 50 && aver < 100){
        buff[0] = 'b';
    }else if(aver >= 100 && aver < 150){
        buff[0] = 'c';
    }else if(aver >= 150 && aver < 200){
        buff[0] = 'd';
    }else if(aver >= 200 && aver < 500){
        buff[0] = 'e';
    }else if(aver >= 500 && aver < 1500){
        buff[0] = 'f';
    }else if(aver >= 1500){
        buff[0] = 'g';
    }
}
send(client, buff, 0, 128, 0);
memset(buff, 0, 128);
recv(client, buff, 128, 0);
    
```

그림 11. CV 분석결과와 UDP 통신  
 Fig. 11. UDP Communications of CV Analysis Result.

그림 11의 코드에 나타내었듯이 계산된 변동 계수는 반복적인 실험을 통하여 변동계수가 25이상일 때 이상신호가 감지되었다고 판단하였다. 변동 계수의 이상이 판별되면 각각의 평균의 범위에 따라 특정 신호를 서버 컴퓨터에 송신하게 하였고 이를 통해 서버 컴퓨터는 평균 어느 정도의 음성 신호 상에서 이상신호가 감지되었는지를 판단 할 수 있게 되었다.

### 3. 실험결과

그림 12와 같이 입력 받은 소리의 값이 시간 순으로 출력된 것을 바탕으로 분석하여 이상 신호 표준편차의 변화로 이상신호가 감지되었을 때 서버로 신호 전송을 하게 된다.

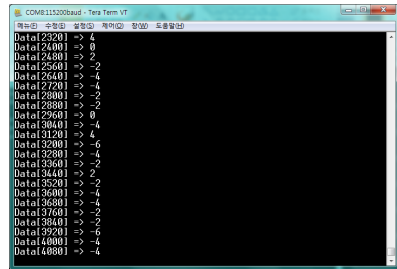


그림 12. 입력된 소리의 출력  
 Fig. 12. Output of Incoming Sound Value.

그림 13과 같이 서버 프로그램 실행 후 클라이언트에 접속하고 이상신호가 감지되면 감지 된 시간과 함께 경보 메시지가 서버에서 출력됨으로써 제안한 시스템의 동작을 실험을 통하여 확인할 수가 있었다.

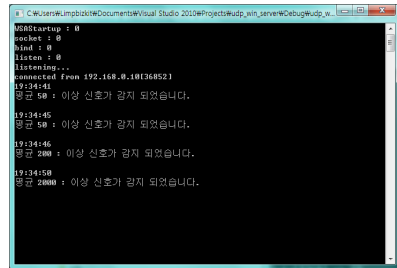


그림 13. 이상 신호 감지 시간과 경보 메시지 출력  
 Fig. 13. Output of Warning Message and Abnormality Signal Detection Time.

## V. 결론

본 연구에서는 Verilog-HDL을 이용하여 FPGA를 구현하고 소리 신호를 받아서 FIFO 방식을 이용 한 데이터 처리와 함께 이를 분석 하고 UDP 통신 방식을 이용하여 사용자에게 이상 유무를 알리는 감시 정보체계를 설계 제작하였다. 구현된 시스템에서 FPGA부분은 마이크 입력을 통해 들어온 소리 신호를 A/D 변환 과정을 거쳐 FIFO를 통해 맵핑된 레지스터에 저장이 되고 4096개의

배열에 저장된다. 이후 이상 신호 감지 여부를 위하여 신호들을 분석한 후 감지 여부에 따라 특정 신호를 서버로 송신 할 수 있도록 하였고, UDP 통신을 통하여 분석 결과를 모듈과 윈도우간 실시간으로 원활하게 주고받는 결과를 볼 수 있도록 하였다.

이러한 제안된 방법으로 구현된 FPGA를 이용한 음성 신호 감지 시스템의 실험결과는 다음과 같다. FPGA를 이용함으로써 감시체계를 필요로 하는 곳에서 직접 상황에 맞는 경보 설정 및 적용이 용이하였다. 또한 입력된 신호를 FIFO 구조에 적용하여 통계학적으로 분산을 이용한 분석 알고리즘을 기반으로 빠르게 이상신호를 감지할 수 있게 하였고, 변동계수를 사용한 알고리즘을 통하여 효과적으로 분석 프로세스를 간략화 할 수 있었다. 제안된 방법을 통하여 100회 이상의 반복실험을 한 결과 96.3%의 감지율을 보였으며, 99% 이상의 감지율을 보이지 못한 이유는 소리와 음성의 경계에 있는 데이터의 오인식에 비롯된 것을 확인 할 수 있었다.

## References

- [1] J.I. Kim, N. H. Kang, "Secure Configuration Scheme of Pre-shared Key for Lightweight Devices in Internet of Things", International Journal of Internet, Broadcasting and Communication, Vol. 15, No. 3, pp.1-6, 2015.
- [2] Ministry of Land Infrastructure and Transport, Ministry of Justice, "CPTED(Crime Prevention Through Environmental Design) Guide Book", auri(Architecture & Urban Research Institute), 2014.
- [3] S.E. Hwang, J.W. Kim, "A Study on the Definition of CCTV Monitoring Duty for K-12 School Safety People", Journal of the Korean Society of Hazard Mitigation, Vol. 15, No. 1, pp.1-8, 2015.
- [4] J.W. Kim, "Smart Fire Image Recognition System using Charge-Coupled Device Camera Image", Journal of Korean Institute of Fire Science and Engineering, Vol. 27, No. 5, pp. 77-82, 2013.
- [5] S.B. Park, J.H. Kim, "Circuit Design and Implementation for Noise Enhancement of Optical Mouse", The Journal of The Institute of Internet, Broadcasting and Communication, Vol. 14, No. 2, pp.135-140, 2014.
- [6] Y.C. Kim, "A Study on Implementation of a PKI(Public Key Infrastructure) Based Embedded Security System for Next-Generation Information Appliances", Institute for Information Technology Advancement, 2004.
- [7] H.S. Hong, S.U. Lee, Y.C. Lee, M.S. Han, "Tangible Stronger Media Voice Recognition Technology for Noise", Korea information processing society review, Vol. 10, No. 1, pp.19-24, 2003

## 저자 소개

### 김 장 원(중신회원)



- 1990년 : 명지대 공과대학 전자공학과 공학사
- 1992년 : 명지대 대학원 전자공학과 공학석사
- 2001년 : 명지대 대학원 전자공학과 공학박사
- 1993년 ~ 현재 : 가천대학교 전자공학과 교수

<주관심분야 : 영상신호처리, 영상이해, 임베디드 시스템, 인터넷 통신