

무선 센서 인터페이스 모듈과 NCAP 구조의 구현

학생회원 오 세 문*, 금 민 하**, 김 동 혁*, 종신회원 김 진 상*, 정회원 조 원 경*

Implementation of the Wireless Transducer Interface Module and NCAP architecture

Se-moon Oh, Min-ha Keum, Dong-hyeok Kim *Student Members*,
Jin-sang Kim *Lifelong Member*, Won-kyung Cho *Regular Member*

요 약

본 논문은 최근 표준화가 완성된 IEEE 1451.5 표준을 적용한 네트워크 접속가능한 응용프로세서(NCAP: network capable application processor)와 무선 트랜듀서 접속 모듈(WTIM:wireless transducer interface module)의 구현에 대한 연구이다. PC와 무선통신 모듈로 구성되는 NCAP부와, FPGA, 센서 보드, 무선 통신모듈로 구성되는 WTIM 부로 구성된다. NCAP부의 구현에는 C++ 언어가 사용되었고 WTIM 구현에는 FPGA를 이용하여 Verilog-HDL이 사용되었으며 NCAP과 WTIM과의 무선통신은 Zigbee를 이용하였으며 Zigbee의 기능구현을 위하여 nesC를 이용하였다. 본 논문에서는 NCAP과 WTIM은 IEEE 1451.0와 IEEE 1451.5 표준을 통하여 서로 통신하도록 구현하였으며 표준에 근거하여 두 가지 실험을 실시하였다. 실험을 통하여 제안된 구조가 IEEE 1451.5 표준의 기능적인 부분을 효과적으로 수행하는 것을 검증하였다.

Key Words : WTIM, NCAP, FPGA, IEEE 1451, Zigbee

ABSTRACT

This paper presents an implementation of the Network Capable Application Processor (NCAP) and the Wireless Transducer Interface Module (WTIM) architectures based on the new IEEE P1451.5 standard. Proposed architecture is implemented using a computer for NCAP, an FPGA board, a sensor board and two radio modules, which communicate through the ZigBee wireless communication technology between the NCAP and the WTIM based on the IEEE 1451.0 and the IEEE 1451.5 interfaces. In this paper, two experiments has been done to verify operations of proposed architecture. From the experimental results, we verify that the proposed architecture performs the wireless sensor communication functions efficiently.

1. 서 론

무선 센서 네트워크의 활용 범위는 홈 네트워크 시스템에서부터 의학 분야에 까지 매우 광범위하게 이용되고 있다. 이러한 요구에 맞추어 다양한 센서들이 개발되고 있으며 센서를 이용한 신체 신호 계

측 및 각종 산업 현장에서의 시스템에 이르기까지 다양한 분야에서 무선 센서 네트워크가 활용되고 있다.

센서 네트워크가 널리 활용됨에 따라 센서간의 호환성을 위하여 IEEE에서 센서 네트워크에 관한 표준을 제정해 왔는데, IEEE 1451은 센서 네트워크 표준

※ 본 연구는 한국학술진흥재단(KRF-2006-521-D00337), ITRC(IITA-2008-C1090-0801-002)의 지원으로 수행되었습니다.

* 경희대학교 전자·전파공학과 VLSI&CSA Lab (neogoril@vlsi.khu.ac.kr), ** 삼성전자

논문번호 : KICS2008-02-098, 접수일자 : 2008년 2월 26일, 최종논문접수일자 : 2008년 10월 17일

으로서 센서와 네트워크 환경 간의 통신 인터페이스를 정의하며 그 중 IEEE 1451.5는 무선 통신 프로토콜과 인터페이스를 정의하고 있다. 최근 IEEE 1451.5의 구현에 관한 연구가 진행되고 있다. UML(Unified Modeling Language)틀과 자바 프로그래밍을 이용한 1451.0과 1451.5 표준 구현 [1], 모듈 방식의 하드웨어와 소프트웨어의 구조에 대해 1451.5 기반 고속 스마트 센서의 TIM (Transducer Interface Module) 개발 [2], 1451.5 기반 무선 Ad-hoc 센서 네트워크 구현 [3] 등이 대표적인 연구이다.

본 논문의 목적은 IEEE 1451.5를 기반으로 하여 WTIM과 NCAP의 상세 구현구조를 제안하여 이를 검증하여 향후 WTIM의 집적회로를 설계하는데 이용하고자 하는 것이다. 본 논문에서는 이를 위하여 정보를 수집하는 센서부와 수집된 정보를 무선네트워크를 이용하여 전송하는 TIM, 전송받은 데이터를 표시 및 처리하거나 데이터에 따라 필요한 작업을 명령하는 등의 어플리케이션이 있는 NCAP부의 상세 구현구조를 각각 센서보드, FPGA, PC를 이용하여 구현한다.

본 논문의 II장에서는 무선 센서 네트워크 인터페이스의 표준에 대해 설명하고 III장에서는 제안된 NCAP과 WTIM의 구조를 설명하고 IV장에서 실험결과, V장에서 결론을 맺는다.

II. 무선 센서 네트워크 인터페이스 표준

IEEE 1451 표준은 센서 네트워크를 실행하는데 필요한 기본적인 핵심내용을 정의한다 [4]. 예를 들면 센서 데이터를 읽어오는 절차와 이와 반대로 액추에이터에 쓰기 명령을 통하여 이를 구동하는데 필요한 인터페이스를 정의하고 유무선 인터페이스 종류와 각 인터페이스 간의 프로토콜에 대해 정의하고 있다. IEEE 1451의 목적은 센서 데이터를 읽어오는데 필요한 인터페이스에 독립적인 공통적인 인터페이스를 정의하는 것이다. IEEE 1451.0는 다른 1451 표준과는 달리 인터페이스 종류에 대한 표준을 설명하는 것이 아니라 모든 1451시리즈에 공통적으로 필요한 명령어 셋 들을 기술하며 1451.x의 물리적 계층을 통해 통신할 수 있게 하는 프로토콜과 Transducer Electronic Data Sheet(TEDS)의 포맷을 정의하고 있다 [5, 6]. IEEE 1451.x는 6개의 표준으로 나뉘는데 1451.1부터 1451.5까지는 물리적 계층의 인터페이스에 따른 표준으로 정의되고 있다. 그림 1은 IEEE 1451 표준을 간략하게 요약

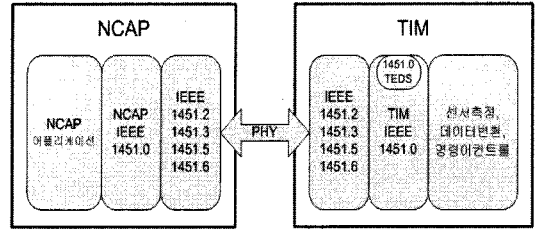


그림 1. IEEE 1451 표준

한 것이다. IEEE 1451.2는 지점 대 지점간의 통신에 대한 프로토콜을 정의하고 있으며 IEEE 1451.3은 버스를 이용한 멀티-드롭 형태의 프로토콜을 정의하며 IEEE 1451.4는 아날로그와 디지털 신호를 혼합한 형태의 모드를 지원하는 프로토콜을 정의하고 있으며 IEEE 1451.5는 무선 인터페이스를 지원하는 프로토콜을 정의하고 있다. IEEE 1451.5의 경우 무선 통신을 이용한 인터페이스의 표준을 기술하며, WiFi, Bluetooth, Zigbee 등의 무선 프로토콜이 사용될 수 있다.

IEEE 1451의 구조는 크게 NCAP과 TIM으로 구성된다. NCAP은 TIM의 데이터를 1451.x의 인터페이스를 통해 전달받아 데이터를 저장하고 분석하는 등의 기능을 하며, 네트워크 기능을 할 수 있는 PC 나 PDA 등의 장비를 이용하여 구현할 수 있다. TIM은 여러 개의 센서에서 취합한 데이터를 NCAP으로 전달하거나 NCAP에서 받은 명령을 액추에이터 측으로 전달하여 동작하게 하는 변환기의 역할을 하며 단순히 센서와 NCAP간의 매개체의 역할 뿐만 아니라 스마트 센서로서의 역할을 담당하며 수요가 많아 향후 집적회로로 구현하는 것이 유리하다.

TIM이 스마트 센서로서의 역할을 할 수 있게 하는 것은 TEDS 때문이라고 할 수 있다. TEDS는 센서의 각종 타이밍 정보를 포함 측정범위, 센서의 종류, 센서의 개수 및 단위 등 데이터 시트를 통해 얻을 수 있는 모든 정보를 저장하고 있으며 1451 표준에 따르면 TEDS 정보를 우선적으로 읽어 들임으로써 어떤 종류의 센서가 연결되든지 TEDS를 통해 자동적으로 NCAP이 인식할 수 있으므로 스마트 센서의 중요한 역할을 한다고 할 수 있다. TIM이 1451.5의 무선인터페이스를 갖게 되면 이를 WTIM이라고 한다.

III. 제안된 NCAP과 WTIM의 구조

그림 2는 NCAP과 WTIM의 간단한 기능적인 구조를 보여주고 있다. 본 논문에서는 NCAP의 구조

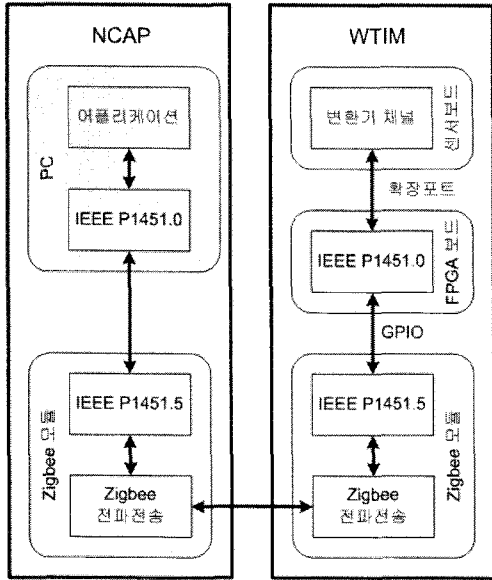


그림 2. NCAP 과 WTIM의 기능적인 구조

는 PC에 C++언어로 구현하고 Zigbee 통신 모듈과는 USB 포트로 연결된다. WTIM의 구조는 센서 보드와 컨트롤러 역할을 하는 FPGA가 외부 확장 포트로 연결되어 있고 통신 모듈과는 GPIO를 이용하여 연결되어 있다. 무선 통신을 위한 프로토콜로는 Zigbee 프로토콜을 이용한다.

3.1 NCAP의 구조

NCAP은 NCAP 응용부, IEEE 1451.0과 IEEE 1451.5로 구성된다. 1451.0은 NCAP 응용부와 1451.5 사이의 인터페이스이며 1451.5는 WTIM과의 무선 인터페이스 기능을 담당한다. NCAP 응용부는 데이터를 읽기 위한 명령을 1451.5를 통해 WTIM으로 전달하거나 WTIM 측에서 전송되어 온 데이터를 처리하여 화면에 보여주는 역할을 한다. 1451.0에서는 응용부에서 온 명령들을 1451.5에서 전송 시 사용하는 패킷의 포맷에 맞게 만들어 주는 역할을 하며 WTIM에서 읽어 온 데이터들을 분석해서 유용한 데이터만 NCAP 응용부로 전달하는 역할을 수행한다.

그림 3은 NCAP 응용부의 상세구조이다. Open device부터 버퍼를 쓰고 읽는 버튼과 채널 1,2를 모두 읽거나 따로 읽는 명령을 주는 버튼, TEDS를 읽어오는 버튼과 각종 메시지 창이 있다. Open device를 통하여 PC와 무선 통신 모듈간의 USB를 통한 통신을 시작하며 TEDS를 읽어오는 버튼과 센서 데이터를 읽는 버튼을 실행시켜 실제로 쓰기 버

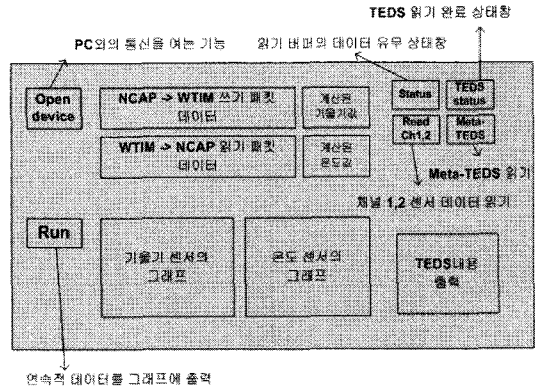


그림 3. NCAP 응용부

퍼와 읽기 버퍼에 저장된 데이터를 확인할 수 있을 뿐 아니라 TEDS에 저장된 데이터를 이용한 보정된 값을 확인할 수 있으며 이를 그래프로 연속적으로 보여주는 기능이 포함된다.

IEEE 1451.5는 Zigbee 방식을 이용하며, Zigbee 스택은 TinyOS를 이용하여 구현된다 [7]. TinyOS에서 제공되는 Tosbase 프로그램은 UART로 들어온 데이터는 전파로 보내고 전파로 받은 데이터는 UART로 내보내는 역할을 하는 프로그램이다. 약속된 패킷의 포맷에 맞추어 데이터를 USB를 통해 직렬전송하면 무선으로 송신되어 고유의 Group ID가 동일한 WTIM에서 수신하여 명령을 처리하게 되고 무선으로 들어온 데이터는 패킷 그대로 1451.0로 전달되어 유용한 데이터를 추출해낼 수 있게 하였다.

무선 통신 모듈 (Zigbee)에 있는 FT232RL 칩은 USB를 통한 입력을 UART로의 데이터와 같게 처리할 수 있게 해주는 기능을 해주므로 직렬 통신 응용부를 쉽게 처리할 수 있다. 직렬 통신은 패리티 비트, 스톱 비트, 데이터 비트, baud rate 등을 설정해 줘야 하며 각각의 값은 0, 1, 8, 57600 으로 설정하였다. WTIM으로부터 센서 및 TEDS 데이터를 읽기 위해서는 NCAP에서 WTIM으로 읽기 명령을 수행해야 한다. 이 때 1451.0에서 1451.5 무선 통신 모듈에게 약속된 포맷의 패킷 형태로 명령을 전달해야 한다. 1451.0은 무선 통신 모듈에게 약속된 포맷의 패킷 형태에 따라 표 1과 같이 패킷의 구성요소에 맞게 데이터를 재배열하여 1451.5로 보내며 Zigbee 모듈내의 CC2420 칩은 USB로 받은 데이터를 무선으로 전송하게 된다. 패킷의 전송방법은 그림 4에 나타나 있다.

nesC로 구현된 Tosbase 프로그램은 UART로 들어온 데이터를 전파로 전달하며 Radio로 받은 데이

표 1. 패킷의 구성요소

패킷의 구성요소							
SYNC_BYTE		PACKET TY PE		PAYLOAD D ATA		SYNC_BYT E	
0		1		2 ... n-1		n	
패킷의 구성요소							
Address	Message Type	Group ID	Data L ength	Data	CRC		
0	1	2	3	4	5 ... n-2	n-1	n
패킷의 내용							
SYNC_BYTE		시작과 끝을 알리는 바이트					
PACKET TYPE		확인 비트가 필요없는 타입					
Address		브로드캐스트 : FF FF					
Message type		임의의 타입					
Group ID		동일한 그룹 : 5D로 통일					
Data length		20 바이트					
Data		명령 또는 데이터					
CRC		값의 진위성을 판단					

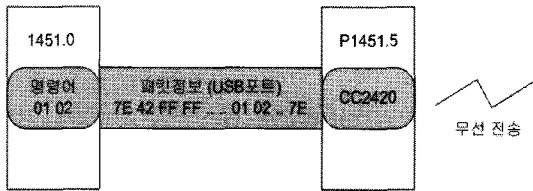


그림 4. 패킷의 전송방법

터는 UART로 전달해 주는 프로그램이다. NCAP에 연결된 Zigbee 모듈은 Tosbase가 프로그램된 것으로 NCAP이 USB포트를 통해 명령을 전달하면 전파로 전달하여 WTIM의 송수신 모듈로 수신하여 FPGA로 명령을 전달하는 절차를 거친다. 이때 전달되는 전파 방식을 브로드캐스팅이라고 하며 WTIM은 고유한 Group ID 값을 가지고 자신의 데이터가 맞는지 알아낸다. NCAP과 WTIM의 Group ID는 동일하여 유용한 값으로 인식할 수 있으며, 이는 전파송수신 모듈에 프로그램 되어 있다. 패킷의 내용 중 데이터의 길이, 데이터의 내용, CRC 등은 입력 데이터에 따라 변경될 수 있는 값이다. 그 외의 sync_byte, packet_type 등은 1451.0에서 부가된다. 패킷의 구성요소에 관한 내용은 표 1과 같다.

NCAP 응용부는 TEDS 데이터를 읽기 위해 먼저 TEDS를 읽겠다는 명령을 보낸 후에 정해진 시간만큼 기다리게 된다. 만일 정해진 시간 동안 데이터가 도착하지 않는다면 시간을 초과했다는 메시지와 함께 TEDS 읽기 동작을 멈추게 된다. 반대로 정해진 시간이 되기 전에 데이터를 수신하게 되면 응용부 상단 메시지 창에 "there is Rxbuffer"라는 메시지가 나타나게 되는데 이는 수신 버퍼에 데이터가 들어

와 있음을 알리는 신호로서 버퍼를 읽는 동작을 수행해도 좋다는 의미이다. 버퍼를 읽어 오게 되면 1451.0에서 패킷을 분해하게 되는데 데이터로서 유용한 부분만 추출하여 NCAP 응용부로 전달하여 TEDS 정보를 저장하게 된다. TEDS 정보를 읽어오는 동작은 NCAP 응용부에서 TEDS 데이터를 요청하고 TEDS 데이터의 마지막인지를 검사하고 기다리기를 반복하여 이루어지는데 데이터 수신은 끝은 TEDS의 가장 마지막 데이터인 checksum값이 채워지느냐에 따라서 TEDS 데이터 수신은 종료될 수 있다. 그러므로 checksum 값이 어떠한 데이터가 저장되게 되면 NCAP 응용부는 더 이상 TEDS 데이터를 기다리지 않고 상단의 메시지창에 "TEDS read done"이라는 메시지를 나타내며 TEDS 데이터를 읽는 동작을 종료한다.

TEDS 데이터는 NCAP에게 많은 유용한 정보를 제공한다. Upper/lower range의 경우 입력받은 센서 데이터의 최고/최저 값을 알려줌으로써 특별히 센서 값을 보정해 주기 위한 식이 없는 경우에 입력받은 센서 데이터를 선형적이라고 생각하므로 ADC를 통해 받은 값을 데이터시트를 이용하지 않더라도 우리가 알아볼 수 있는 값으로 변환하기 위한 중요한 변수의 역할을 한다. Physical unit의 경우는 단위 정보를 가지고 있는 것으로써 섭씨(화씨), 킬로그램 등의 단위를 알아내어 디스플레이 할 수 있다. 또한 Identification TEDS의 경우에는 각종 센서의 설명들이 문자열 형태로 저장되어 있어서 몇 개의 센서가 연결되어 있는지, 어떤 종류의 센서인지 등의 설명을 사용자가 쉽게 알아볼 수 있도록 디스플레이 해 주기 위한 것이다. 사용자가 이런 TEDS의 정보를 문자열 형태로 보기 위해서는 "TEDS view"버튼을 누르면 Identification TEDS에서 읽어 온 설명들이 하단 메시지 박스에 출력되게 된다. "TEDS

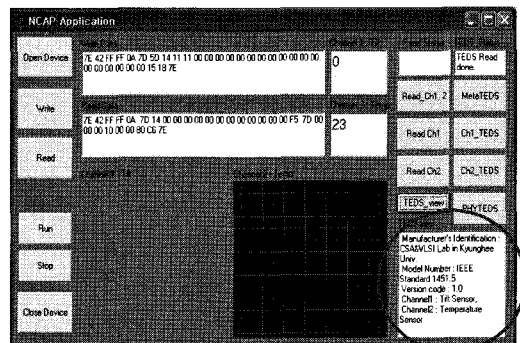


그림 5. NCAP 응용부의 TEDS 내용 출력화면

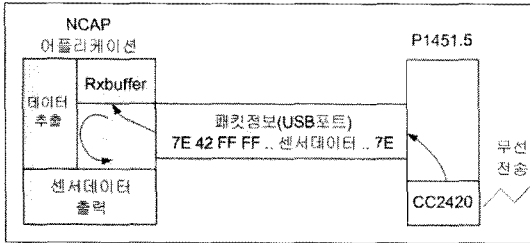


그림 6. NCAP이 센서 데이터를 읽어오는 과정

view”의 결과는 그림 5에 나타나있다.

센서 데이터를 읽어오기 위해서 NCAP 응용부는 센서 read 명령을 WTIM에게 전달해야 한다. 전달 체계는 NCAP 응용부에서 명령을 발생시키면 1451.0에서 패킷을 완성하여 1451.5로 전달하고 이를 무선으로 전파하면 WTIM의 1451.5에서 동일한 Group ID 인지를 판단하여 동일하면 수신하여 처리하게 된다. TEDS 읽어오는 절차와 동일하게 처리하며 NCAP의 입장에서 보면 읽기 위한 명령을 전달하고 수신되는 버퍼를 기다리다가 버퍼에 데이터가 채워지면 읽어서 유용한 데이터를 추출한 후에 정확한 값을 계산하여 디스플레이 해준다.

그림 6은 센서 데이터를 읽어오는 과정의 한 예이다. 그림 4의 패킷 전송방식에 따라 NCAP 응용부가 센서를 읽겠다는 명령을 WTIM에게 전달하면 WTIM은 센서모듈에 센서 데이터를 요청하고, 받은 센서 데이터를 다시 NCAP에 전달하게 된다. 전달된 패킷 정보를 NCAP 무선통신모듈이 수신하여 USB를 통해 PC로 전달되며 NCAP 응용부는 수신된 버퍼가 있음을 확인하여 패킷의 정보 중 필요한 센서 데이터만 추출하여 TEDS에 따라 사용자가 요구한 센서 값을 계산하여 화면에 출력하게 된다.

3.2 WTIM의 구조

WTIM은 크게 무선통신, 센서 데이터 처리 및 제어, 센서 및 센서접속 기능을 수행해야 하는데, 그림 7과 같이 위의 기능을 각각 무선통신모듈, FPGA 보드, 센서보드가 담당한다. 무선통신 모듈은 NCAP으로부터 데이터를 받아 FPGA 보드로 전송하거나, FPGA 보드로부터 데이터를 받아 NCAP으로 전송하는 역할을 한다. FPGA 보드의 제어 모듈은 NCAP의 명령에 따라 TEDS 또는 TC 센서의 데이터를 읽어서 전송하는 역할을 한다. 센서보드는 (본 논문에서는 온도센서와 기울기센서를 사용하였음) 센서로부터 감지된 신호를 디지털로 변환하여 FPGA 보드로 전송하는 역할을 한다.

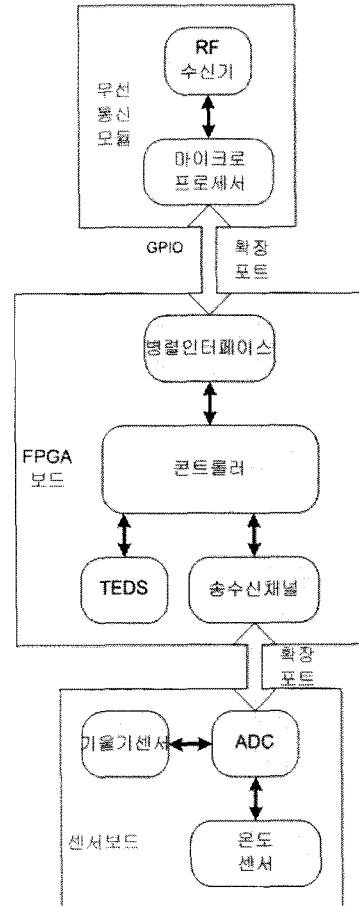


그림 7. WTIM의 블록도

무선통신 모듈은 NCAP으로부터 받은 패킷에서 데이터영역 부분을 추출하여 FPGA 보드로 명령을 전송하거나, FPGA 보드로부터 받은 데이터를 패킷의 데이터영역부분에 삽입하여 NCAP으로 패킷을 전송하는 기능을 수행한다. 이때 FPGA 보드와의 통신을 위해서 Atmega128 마이크로프로세서의 GPIO 포트를 이용한다.

FPGA 모듈은 Verilog를 이용하여 설계되었으며 IEEE 1451.0의 기능이 구현된 부분이다. 사용된 FPGA 보드는 Altera DE2 개발 보드이며, Altera Cyclone 2 FPGA가 포함되어있다. NCAP으로부터 받은 명령을 해석하고 그에 해당하는 센서 또는 TEDS의 데이터를 전송하는 부분이다.

FPGA 보드의 컨트롤러 블록은 그림 8과 같이 상태기계로 구현되었으며 각각의 상태에 따른 동작은 다음과 같다. INIT상태에서는 컨트롤 블록의 전체적인 초기화 동작이 수행되고, 그 다음 Register Module 상태에서는 통신 모듈이 초기화 동작을 수

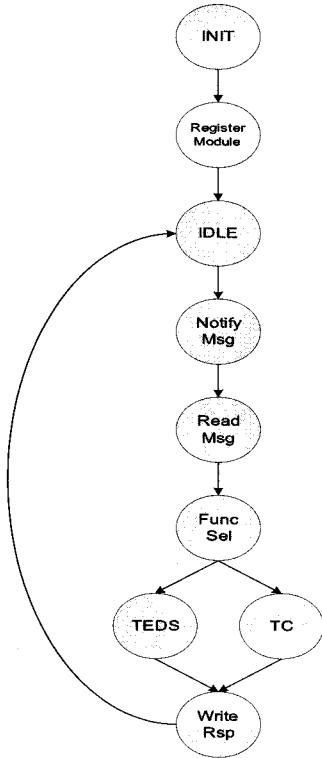


그림 8. WTIM의 상태도

행하고 통신 가능한 대기 상태인지를 확인하게 된다. 그 후 IDLE상태에서는 NCAP으로부터의 명령을 받기 위해 대기하고 있는 상태이다. NCAP으로부터 명령을 받게 되면 통신 모듈로부터 notify 신호가 전달되고 read msg 상태로 전환된다. Read msg 상태에서는 무선통신 모듈로부터 명령 데이터를 받게 된다. 그 후, Func Sel 상태에서는 명령을 해석하고 그 명령의 종류에 따라 다음 상태를 결정하게 된다. TEDS 상태와 TC 상태에서는 TEDS나 센서의 데이터를 얻기 위해 해당 블록을 동작 상태로 전환시키게 된다. 이때 센서 모듈로부터 센서 데이터를 얻기 위해서는 ADC의 특성상 일정시간의 샘플링 시간이 필요하다. Write Rsp상태는 NCAP에서 원하는 TEDS 데이터나 센서 데이터를 통신 부분을 통해 보내는 동작을 수행하게 된다. 데이터의 모든 전송이 끝나게 되면 다시 IDLE상태로 전환되어 NCAP으로부터의 다음 명령을 기다리게 된다.

TEDS 블록은 WTIM 전체의 정보를 담고 있는 Meta-TEDS, 채널1과 채널2의 센서에 대한 정보를 담고 있는 Channel_1-TEDS, Channel_2-TEDS, 통신 부분의 정보를 담고 있는 PHY-TEDS로 구성된다. 디바이더 블록을 이용하여 처리 부분의 동작 속

도를 조절함으로써 WTIM 전체의 동작속도를 제어할 수 있게 된다.

본 논문에서 구현한 센서 모듈은 온도 센서와 기울기 센서 그리고 ADC로 구성되어있다. 채널 1이 기울기 센서, 채널 2가 온도 센서에 해당한다.

IV. 실험결과

제안된 구조의 검증하기 위해 WTIM에 연결되어 있는 표준에 기반을 두어 TEDS 정보를 읽는 과정과 두 개의 센서 데이터를 읽는 과정을 실험하였다. NCAP과 WTIM의 초기화 과정, TEDS데이터를 읽는 과정, TC센서 데이터를 읽는 과정이 포함되어 있다. WTIM의 구현 시스템은 그림 9와 같다.

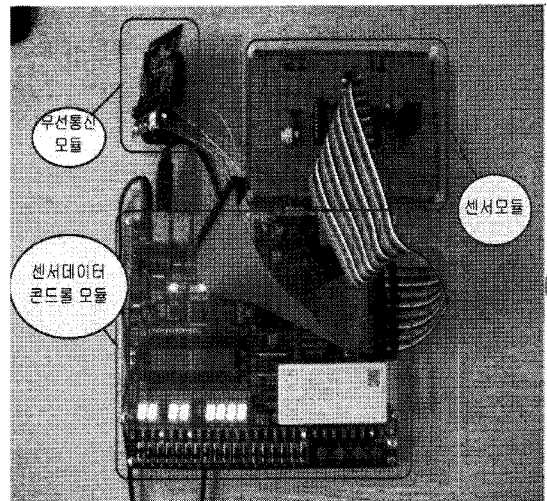


그림 9. WTIM의 구현

4.1 read TEDS의 실험 결과

그림 10의 단계에 따라 NCAP 응용부에서는 Meta TEDS 함수를 실행시켜서 명령을 1451.0에서 패킷에 실어 통신 모듈로 전송하게 되고, 통신 모듈에서는 패킷을 받아 WTIM으로 보낸다. WTIM의 통신 모듈은 NCAP으로부터 전송된 패킷을 받으면 처리 모듈로 notify Msg 신호를 보내게 되고, 그 후 처리 모듈은 read msg 함수를 실행시켜서 통신 모듈로부터 명령을 읽게 된다. 처리 모듈의 컨트롤 블록에서 이 명령을 해석한 후 meta TEDS를 읽어서 통신 부분으로 모두 전송하게 된다. 통신 부분에서는 이 데이터를 20byte 단위로 NCAP으로 보내게 된다. NCAP의 통신 부분에서는 데이터를 받은 후 NCAP 응용부로 notify rsp 신호를 보내게 된다.

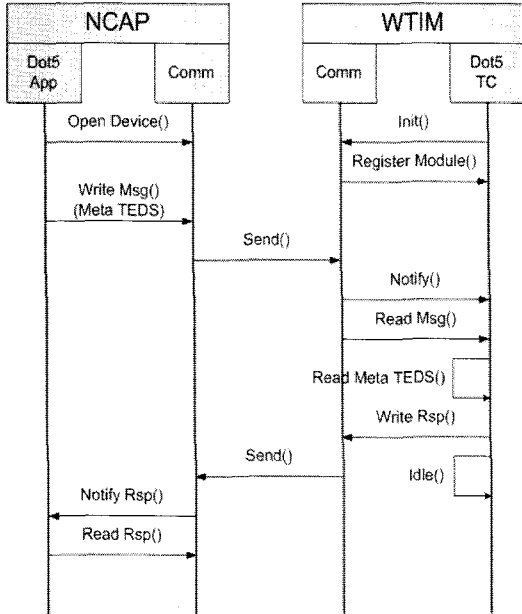


그림 10. Read TEDS 명령어 수행 과정

이때 NCAP 응용부에서는 read status 창을 통해 notify rsp 신호를 확인 할 수 있고 Read 버튼을 누름으로 함수를 실행시켜 데이터를 읽게 된다. 패킷의 전체 내용을 확인하는 창과 센서의 데이터를 확인하는 창이 있으므로 전송 및 수신되는 데이터의 값을 확인할 수 있다.

그 후 NCAP의 통신 모듈은 다시 명령을 보내게 되고 WTIM의 통신 모듈은 다음 데이터를 보내게 되는 과정을 자동적으로 반복하게 되는데 이는 TEDS의 값을 저장하는 구조체의 마지막 값인 checksum에 데이터가 채워지게 되면 더 이상 반복하지 않으며 완전한 meta TEDS 데이터를 얻게 된다. meta TEDS에는 meta-identification TEDS를 포함하였으며 NCAP 응용부에서는 TEDS view 함수를 실행 시켜서 meta-identification TEDS 데이터를 텍스트 형태로 화면에 출력할 수 있다(그림 5 참조).

4.2 Read TC의 동작 흐름 및 결과

NCAP에서는 WTIM으로 tc1센서 데이터와 tc2센서 데이터를 읽기 위한 명령을 보낸다. WTIM에서는 명령을 해석한 후 센서 모듈을 동작 가능한 상태로 만들고 tc1센서 데이터를 읽어서 통신 모듈로 전송을 한다. 그 다음 tc2센서 데이터를 읽어서 통신 모듈로 전송을 한다. 통신 모듈은 두 채널의 센서 데이터를 패킷에 담아 NCAP으로 전송한다. NCAP에서는 데이터를 받아 NCAP 응용부의

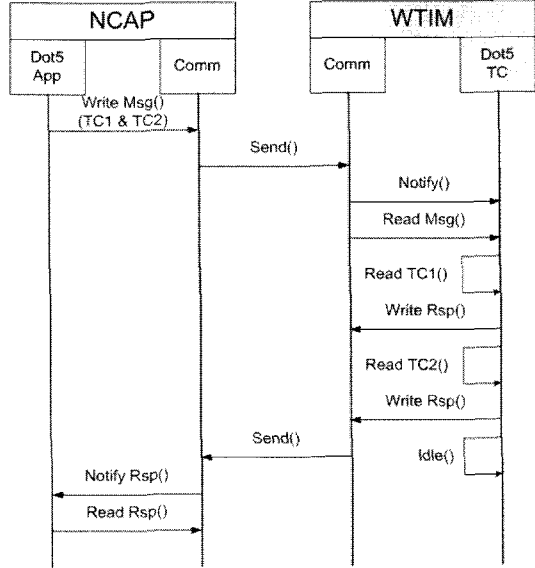


그림 11. Read TC 명령어 수행 과정

channel 1 창과 channel 2창을 통해 받은 데이터를 숫자로 표시하게 된다. 그러나 연속적으로 센서 데이터를 얻고 싶다면 NCAP 응용부에서 run 함수를 실행시켜서 위와 같은 동작을 연속적으로 반복하여 데이터를 얻을 수 있다. Read TC 명령어 수행 과정은 그림 11에 나타나 있다.

그림 12는 연속적으로 센서 데이터를 얻는 모습을 보여준다. tc1센서 데이터와 tc2센서 데이터를 그래프를 통하여 확인 할 수 있다. 그림 13을 보면, 실제로 전송된 데이터의 패킷을 살펴볼 수 있다. TEDS info 창에 TEDS 데이터 패킷을 받고 난 후 센서의 분석된 정보를 확인하고, WTIM에서 센서의 정보를 데이터 페이로드에 실어서 보내게 되면, 데이터의 보정, 단위 환산을 통해 그림과 같이 출력됨을 확인할 수 있다.

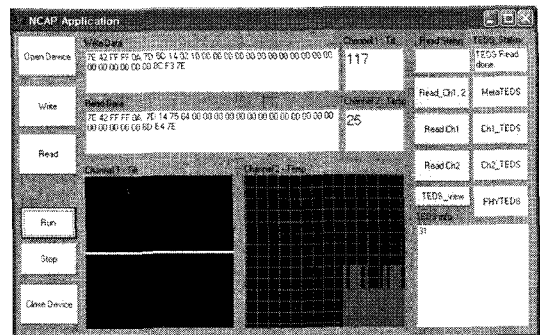


그림 12. 연속적인 센서 데이터 출력 화면

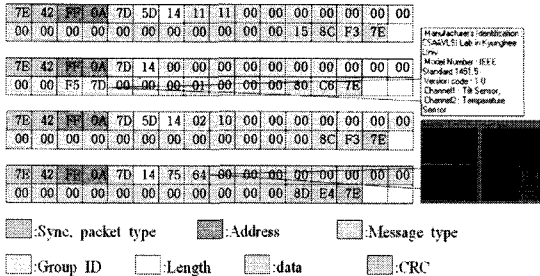


그림 13. packet payload의 전송

V. 결 론

이 논문에서는 NCAP과 WTIM의 구조를 제안하였다. NCAP은 TEDS와 센서 값을 요청하는 명령을 전달하여 각각 센서에 대한 상세한 정보를 얻거나 센서의 값을 TEDS로부터 얻은 데이터시트 자료를 이용하여 계산함으로써 사용자가 식별 가능한 단위 및 문자로 표현해주는 역할을 하는 응용부와 IEEE 1451.5 표준을 따른 무선통신 모듈로 구성되었다. 무선통신 모듈은 Zigbee 방식을 이용하여 정해진 패킷에 전달하고자 하는 명령 및 데이터의 송수신을 담당한다. WTIM의 구조는 센서부와 NCAP의 명령을 받아 디코딩하고 센서부에 데이터를 요청하고 메모리에 저장되어 있는 TEDS 데이터를 NCAP으로 전송하는 일을 하는 제어부와 NCAP과의 무선통신을 위한 무선통신 모듈로 구성된다. 본 논문에서 제안하는 구현구조는 IEEE 1451.0와 IEEE 1451.5의 주요 기능들이 포함되었으며 실험을 통하여 기능을 검증하였다. 본 연구를 바탕으로 향후, WTIM의 기능을 하나의 집적회로로 설계하는 연구를 진행할 예정이다.

참 고 문 헌

- [1] Song, E.Y. and Kang Lee, "An implementation of the proposed IEEE 1451.0 and 1451.5 standards", Sensors Application Symposium, 2006, pp.72-77
- [2] Sweetser, D., Sweetser, V. and Nemeth-Johannes, J., "A modular approach to IEEE-1451.5 wireless sensor development", Sensors application Symposium, 2006, pp.82-87
- [3] Jungil Heo, Jaehwan Kim, Jungho Seo, Suyoung Lim, Junhong Ahn, Junsoo Ahn and Woosik Kim, "A Study on the Implementation

of a wireless Ad-hoc Sensor Network based on the IEEE 1451.5", Advanced Communication Technology, The 9th International Conference on, Volume 1, 12-14 Feb. 2007, pp.419-422

- [4] <http://ieee1451.nist.gov/>
- [5] IEEE 1451.5/D10.8, Draft Standard for a Smart Transducer Interface for Sensors and Actuators - Wireless Communication Protocols and Transducer Electronic Data Sheet (TEDS) Formats.
- [6] IEEE 1451.0/D8.00, Draft Standard for a Smart Transducer Interface for Sensors and Actuators - Common Functions, Communication Protocols, and Transducer Electronic Data Sheet (TEDS) Formats
- [7] <http://www.tinyos.net/tinyos-1.x/doc/tutorial/>

오 세 문 (Se-moon Oh)

학생회원

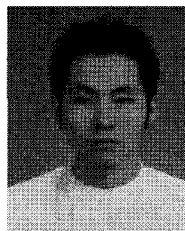
2007년 2월 경희대학교 전자정보학부 학사 졸업
 2007년 3월~현재 경희대학교 전자.전파 공학과 석사과정
 <관심분야> Nanometer VLSI 설계



김 민 하 (Min-ha Keum)

학생회원

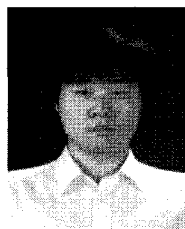
2007년 2월 경희대학교 전자정보학부 학사 졸업
 2008년 8월 경희대학교 전자공학과 석사졸업
 2008년 9월~현재 삼성전자 <관심분야> 무선통신 SoC설계



김 동 혁 (Dong-hyeok Kim)

학생회원

2007년 2월 경희대학교 전자정보학부 학사 졸업
 2007년 3월~현재 경희대학교 전자.전파 공학과 석사과정
 <관심분야> 무선통신 SoC설계



김진상 (Jin-sang Kim)

중신회원



1985년/1987년 경희대학교 전자
공학과 학사/석사

2000년 미국 콜로라도 주립대
전자공학박사 졸업.

2001년 8월 KT 연구소

2001년 8월~현재 경희대학교
전자정보학부 부교수

<관심분야> 무선통신 SoC설계

조원경 (Won-kyung Cho)

정회원



1986년 2월 한양대학교 전자공
학과 공학박사졸업

1980년 3월~현재 경희대학교
전자정보학부 정교수

<관심분야> 컴퓨터 시스템 구
조, VLSI 설계