Journal of Digital Contents Society Vol. 13 No. 3 Sep. 2012(pp. 353-358)
http://dx.doi.org/10.9728/dcs.2012.13.3.353

텍사스 인스트루먼트의 CC2530 상에 IEEE 802.15.4 프로토콜 스택 구현

김병순*

요 약

무선 센서 네트워크는 일반적으로 엔드 장치에 연결된 센서가 측정한 데이터를 코디네이터에게로 전달하는 자원이 한정된 장치들로 구성된다. 무선 센서 네트워크에서 가장 많이 사용되는 매체 접근 제어 프로토콜은 IEEE 802.15.4 표준인데, 이것은 적정한 데이터 전송률을 가지면서 저전력, 저비용의 무선 통신을 지원하기 위해 표준으로 도입되었다. 이 논문에서는 텍사스 인스트루먼트의 CC2530 상에서 IEEE 802.15.4 프로토콜 스택과 UART 인터페이스를 구현하는 방법에 관하여 논의한다. 프레임 캡처를 통해 IEEE 802.15.4 프로토콜이 구현된 보드에서 UART를 통한 데이터 전송이 올바르게 동작됨을 보인다.

Implementation of IEEE 802.15.4 Protocol Stack on the Texas Instrument CC2530

Byungsoon Kim*

Abstract

Wireless sensor networks consist of resource constraint devices which typically send data measured by sensors attached to the end devices towards a coordinator. The most often used medium access control protocol used in wireless sensor networks is the standard IEEE 802.15.4, where it is wireless standard introduced for low power, low cost wireless communication with moderate data rates. In this paper, we present implementation of both the IEEE 802.15.4 protocol stack and UART interface on the TI CC2530. We show that data via UART of our implemented IEEE 802.15.4 board are transmitted correctly in terms of captured frames.

KeyWords: IEEE 802.15.4, TIMAC, CC2530

1. 서 론

무선 센서 네트워크는 물리적 또는 환경적 조건을 모니터링 하기 위해 센서를 사용하는 독자적인 장치들로 구성된 무선 네트워크이다. 무선 센서 네트워크는 제한적인 자원을 갖는 장치들로 구성되며, 이러한 장치들은 센서에 의해 측정된 데이터를 싱크 노드(sink node)로 전달하는 역할을 한다[1]. 이러한 장치들은 일반적으로 작고, 배터리로 동작하며 제한적인 계산 능력을 갖는다.

무선 센서 네트워크는 수질, 토양 또는 기후 측정을 수집하기 위해 장기간 배포 솔루션이 요구되는 환경 모니터링과 같은 응용에 이상적이다. 전력망, 가로등, 도시 상수와 같은 공익설비를 위해 무선 센서는 에너지 사용을 줄이고 리소스를 더욱 잘 활용하기 위해 시스템 상태 데이터를 수집하는 저가형 방식을 제공한다[2].

무선 센서 네트워크에서 사용하는 대표적 물리 매체 프로토콜은 IEEE 802.15.4 표준이다[3]. 이것은 무선에서 매체에 대한 접근을 제어하는 프로토콜로서 매체에 대한 공정한 접근과 충돌없는 데이터 전송을 보장한다.

※ 제일저자(First Author): 김병순
접수일:2012년 08월 07일, 수정일:2012년 09월 04일
완료일:2012년 09월 07일
* 안동대학교 정보과학교육과
bsgim@andong.ac.kr

본 논문에서는 텍사스 인스트루먼트의 CC2530 마이크로 컨트롤러상에서 IEEE 802.15.4 프로토콜 스택 구현과 상위 계층을 위한 UART(universal asynchronous receiver transmitter) 인터페이스 구현과 관련하여 논의한다.

이 논문의 구성은 다음과 같다. 먼저 IEEE 802.15.4 프로토콜과 CC2530 에 대한 기술을 간략히 살펴보고, 3장은 텍사스 인스트루먼트에서 IEEE 802.15.4 MAC 명세서를 구현한 TIMAC 구조를 살펴보며, 4장은 구현에 대하여 설명한다. 5장은 실험 결과에 대하여 설명하고, 마지막으로 결론은 6장에서 맺는다.

2. IEEE 802.15.4, CC2530 및 관련연구

IEEE 802.15.4 표준은 장치들 사이에서 저 비용과 저 속도 통신 제공을 목적으로 무선 개인 영역 통신망 (Personal Area Network)을 위한 프로토콜로서 개발되었으며 물리층과 MAC 부계층으로 구성된다.

물리층은 868MHz, 902MHz, 2405 MHz인 3개의 ISM(Industrial, Science and Medical) 대역을 정의하며, 868MHz 대역은 20Kbps 대역폭의 한 개 채널, 902MHz는 각각 40 Kbps 대역폭인 10개의 채널, 2405MHz는 각각 250Kbps인 16 개 채널로 구성된다. 868MHz와 902MHz 대역은 유럽과 북미에서 지역적으로 사용할 때 적합하며, 2405 MHz는 전세계에서 지역적으로 사용하기에 적합하다[4]. 물리층의 기능은 라디오 트랜시버 열기와 닫기, 채널 선택, 채널 에너지 발견, 프레임 신호 전송 및 수신 등이 있다.

MAC 부계층은 상위층에게 데이터 서비스와 관리 서비스를 제공하며, 비콘 관리, 채널 접근 제어, 보장된 시간 슬롯(guaranted time slot) 관리, 프레임 검사, 프레임 전송 및 응답, 장치 연결 및 해제 등의 기능을 수행한다. 그리고 CSMA/CA 프로토콜을 사용하여 라디오 채널에 대한 접근을 제어한다[4].

텍사스 인스트루먼트(Texas Instrument)의 CC2530[5]은 8051과 호환되는 마이크로 컨트롤러 유닛(Micro Controller Unit)을 갖는 RF 트랜시버(radio frequency transceiver)로서 2.4GHz의 ISM 대역에서 동작한다. DSSS(Direct-Sequence Spread Spectrum)를 사용하여 16개의 데이터 채널을 제공하며 데이터 전송률은 최대 250 Kbps이다. 8051 MCU는 최대 256K 플래쉬 메모리와 8K RAM을 제공하고, 인터페이스는 21개의 GPIO (General-Purpose I/O), 두 개의 USART(Universal Synchronous Asynchronous Receiver/Transmitter), 한 개의 ADC(Analog-to-Digital Converter)를 포함한다. 그리고 보안을 위해 AES-128 암호 기법을 제공한다.

Wauthy[6]는 TI CC2430상에 IEEE 802.15.4-2006 프로토콜을 구현하였고. Bhat[7]는 FPGA상에 IEEE 802.15.4을 구현하였다. 또한 Ko[8]는 비콘을 사용하는 네트워크 상에서 IEEE 802.15.4를 구현하였다. 하지만 우리는 새롭게 출시된 TI CC2530 마이크로 컨트롤러에서 IEEE 802.15.4-2007을 통한 UART 인터페이스 기능을 구현 하고자 한다.

3. TIMAC 구조

TIMAC[9, 10]은 텍사스 인스트루먼트에서 IEEE 802.15.4 MAC 명세서를 구현한 것으로, TIMAC-CC2530-1.4.0.exe 프로그램을 설치하면 필요한 디렉토리 구조를 생성하고 모든 소프트웨어와 문서 파일들을 로드한다. TIMAC 라이브러리와 샘플 응용 프로젝트들은 IAR Embedded Workbench(EW8051) [11] 소프트웨어 개발 툴과 함께 사용된다. TIMAC-CC2530-1.4.0은 EW8051 8.10 버전에서 빌드되었고 테스트되었기 때문에 이 버전의 툴을 구해서 설치하도록 한다. 그리고 빌드후 생성된 .hex 파일을 보드로 다운로드하기 위해 텍사스 인스트루먼트의 SmartRF Flash Programmer 툴을 설치한다.

(그림 1) TIMAC 구조

(그림 1)은 TIMAC의 소프트웨어 구조를 나타낸 것이다. MAC 응용층은 사용자가 802.15.4 MAC 연산을 테스트하고 검정할 수 있도록 기본적인 프로토콜과 기능들을 포함하며, 프로그램의 main() 함수가 위치한다. 그리고 장치의 유형 설정, 네트워크 가입 및 형성, 연결 설정 및 해제, 장치 스캐닝, 데이터 송·수신 등의 기능을 수행한다[9].

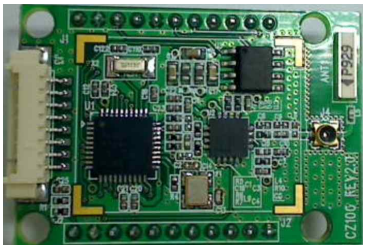
OSAL(operating system abstraction layer)은 처리 환경의 구체적인 것으로부터 TI 스택 소프트웨어를 보호하기 위해 사용된다. 그리고 OSAL 라이브러리는 메시지 관리, 태스크 동기화, 타이머 관리, 인터럽트 관리, 태스크 관리, 메모리 관리, 전원 관리, 비휘발성 메모리, 간단한 비휘발성 메모리, OSAL 클록 시스템 등의 API를 제공한다[12].

MAC층은 IEEE 802.15.4 MAC 소프트웨어에 대한 API(application programming interface)를 제공하는데 이 함수들은 메시지 전달, 직접 실행, 콜백 등 3가지의 인터페이스 메커니즘을 사용한다. 첫째, 메시지 전달 함수 호출 API들의 기능은 OSAL 메시지를 MAC 이벤트 핸들러에게 전송하는 방법으로 MAC에게 메시지 전달 인터페이스를 제공한다. 둘째, 직접 실행 함수 호출 API 함수들은 MAC 연산들을 직접 실행하는 것으로 MAC 계층의 데이터를 직접 접근하기도 한다. 마지막으로 콜백(Callback) 함수들은 응용 프로그램에서 구현이 되어야 하고 MAC 계층에서 응용으로 이벤트와 데이터를 전달하고자 할 때 사용한다[3, 13].

HAL(hardware abstraction layer)는 GPIO와 같은 하드웨어 서비스에 대한 플랫폼 독립적인 인터페이스를 제공한다. HAL 라이브러리는 타이머, GPIO, UART, ADC(analog to digitial conversion) 등의 접근에 대한 API를 제공한다 [14].

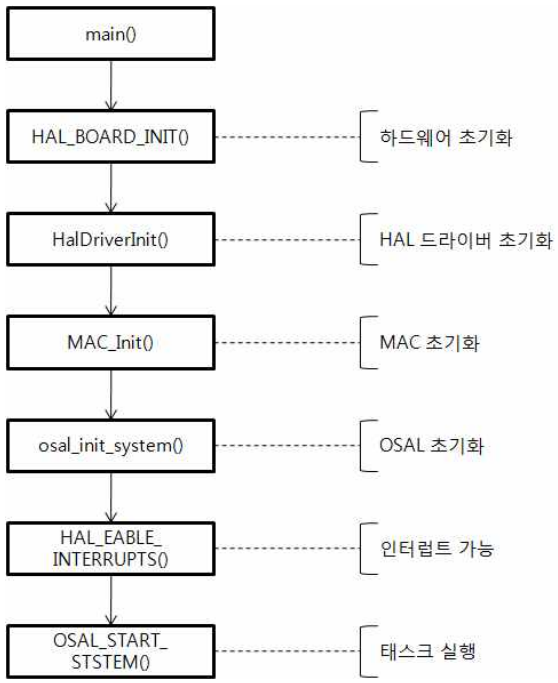
4. 구현

IEEE 802.15.4를 구현할 대상은 씨알지테크놀로지의 CM-Z100 모델로서 하드웨어는 (그림 2)와 같다.



(그림 2) CM-Z100

CM-Z100은 2.4GHz의 무선 전송거리를 확장하기 위해 텍사스 인스트루먼트의 CC2591 칩을 사용하였고, 다른 모듈과의 인터페이스를 위해 RS232인 8핀 커넥트를 제공한다.



(그림 3) TIMAC 함수 다이어그램

(그림 3)은 TIMAC의 함수 호출 과정을 나

타낸 것이다. 먼저 하드웨어를 초기화한 후 HAL 드라이버를 초기화한다. 그리고 MAC과 OSAL을 차례대로 초기화한 후 등록된 태스크들을 반복 실행한다. OSAL_START_SYSTEM() 함수는 tasksArr[] 배열에 등록된 태스크들을 차례대로 반복 실행하므로 새로운 태스크를 생성할 때는 이 배열에 등록해야 한다.

응용 프로그램이 UART 인터페이스를 통해 RF 모듈로 데이터를 전달할 수 있도록 UART 인터페이스를 구현한다. 이를 위해 우리는 UART 인터페이스를 초기화하는 함수와 데이터가 UART 인터페이스에 전달되었을 때 호출되는 콜백 함수를 각각 정의해야 한다. UART 인터페이스 초기화는 응용 프로그램의 함수 중에서 이름이 _Init()으로 끝나는 초기화 함수에서 (그림 4)와 같이 UART 인터페이스를 초기화하는 Serial_Init()함수를 호출하도록 한다. Serial_Init() 함수는 UART 포트의 전송률, 송·수신시 버퍼 크기, 흐름제어 여부, 콜백 함수 이름 등을 설정한다.

```
void Serial_Init(void)
{
  halUARTCfg_t uartConfig;
  // UART의 전송률, 흐름제어 여부, 버퍼 크기,
  // 데이터 입력시 호출하는 함수 이름 지정
  uartConfig.configured = TRUE;
  uartConfig.baudRate = HAL_UART_BR_38400;
  uartConfig.flowControl = FALSE;
  uartConfig.flowControlThreshold =
                        SERIAL_APP_THRESH;
  uartConfig.rx.maxBufSize =
                        SERIAL_APP_RX_SZ;
  uartConfig.tx.maxBufSize =
                        SERIAL_APP_TX_SZ;
  uartConfig.idleTimeout = SERIAL_APP_IDLE;
  uartConfig.intEnable = TRUE;
  uartConfig.callBackFunc = Serial_CallBack;
  HalUARTOpen(SERIAL_APP_PORT,
                &uartConfig);
}
```

(그림 4) UART 초기화 함수

```
void Serial_CallBack(uint8 port, uint8 event)
{
  // UART에 데이터 도착
  if ((event & (HAL_UART_RX_FULL |
      HAL_UART_RX_ABOUT_FULL |
      HAL_UART_RX_TIMEOUT)) &&
      !SerialApp_TxLen)   {
      Serial_Send();
   }
}

void Serial_Send(void)
{
 // 데이터 읽음
 if (!SerialApp_TxLen &&
     (SerialApp_TxLen=
       HalUARTRead(SERIAL_APP_PORT,
       SerialApp_TxBuf,
              SERIAL_APP_TX_MAX))) {
   SerialApp_TxBuf[SerialApp_TxLen] ='₩0';
 }
 // 전송할 데이터 있으면 MAC 계층에게 전송
 // 요청
 if (SerialApp_TxLen) {
      if (msa_IsStarted) {
        if (msa_State == MSA_IDLE_STATE)
          msa_State = MSA_SEND_STATE;
          osal_start_timerEx(MSA_TaskId,
                MSA_SEND_EVENT, 100);
      }
   }
}
```

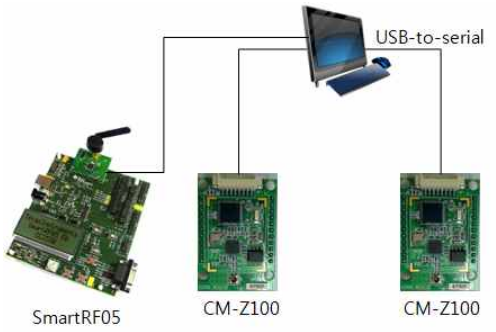
(그림 5) UART 콜백 함수

데이터가 UART 인터페이스에게 도착하면 (그림 5)의 콜백함수인 Serial_CallBack() 함수가 호출되며, UART 포트로부터 데이터를 읽은 후 osal_start_timerEx() 함수를 사용하여 MAC 계층에게 데이터 전송을 요구한다.

프로토콜 스택을 쉽게 구현하기 위해 구현할 대상의 하드웨어와 가장 유사한 하드웨어 모델을 지정한다. 이를 위해 우리는 CC2530과 CC2591로 구성된 HAL_PA_LNA을 IAR embedded workbench의 컴파일 옵션에서 선택하였다. 그리고 UART를 위해 HAL_UART, SERIAL_APP_PORT, HAL_UART_ISR 옵션을 추가로 지정한다.

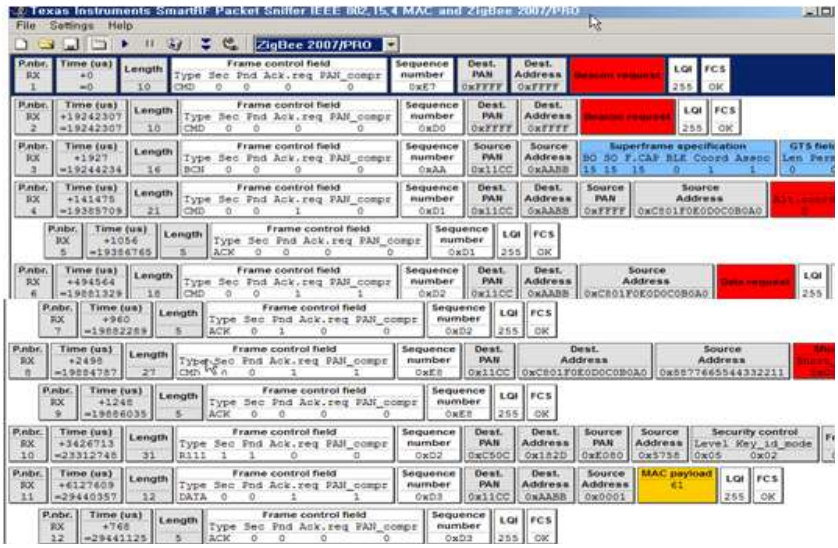
5. 실험 결과

구현된 프로토콜의 동작을 확인하기 위해 컴퓨터 본체에 두 개의 장치를 USB-to-serial 인터페이스를 사용하여 (그림 6)과 같이 연결하였다. 그리고 무선으로 전송되는 프레임을 캡처하기 위해 TI Packet Sniffer 툴이 동작하는 SmartRF05 보드를 추가로 연결하였다.



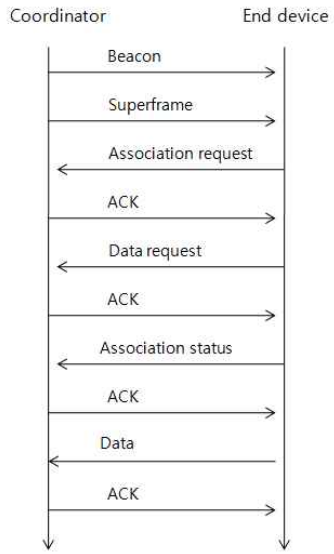
(그림 6) 실험 환경

프로그램을 빌드한 후 생성된 .hex 파일을 SmartRF Flash prgrammer 툴을 이용하여 RF 모듈의 플래시 메모리로 다운로드후 실행하였다. 가장 먼저 실행되는 모듈이 코디네이터(coordinator)로 동작되며 나중에 실행되는 모듈이 엔드 장치(end device)로 동작된다.



(그림 7) 캡처된 패킷

(그림 7)은 TI Packet Sniffer를 이용하여 두 장치 사이에서 전송되는 프레임들을 캡쳐한 것이고, (그림 8)은 캡쳐한 프레임들의 흐름을 다이어그램으로 나타낸 것이다. 코디네이터 장치가 비콘 프레임을 주기적으로 보내며, 엔드 장치가 비콘 프레임을 수신하면 연결 요청을 한다. 두 장치 사이에 데이터 전송을 위한 연결이 이루어진 후 엔드 장치는 코디네이터에게 데이터를 전송하며, 전송된 프레임에 대해 확인 응답인 ACK 프레임을 수신한다.



(그림 8) 프레임 흐름도

6. 결 론

이 논문은 텍사스 인스트루먼트의 CC2530 마이크로 컨트롤러상에 IEEE 802.15.4 프로토콜 스택구현과 응용 프로그램의 인터페이스를 위해 UART 인터페이스를 구현하는 방안에 대하여 살펴보았다.

구현한 하드웨어를 사용하여 코디네이터와 엔드 디바이스로 설정한 후 패킷 캡쳐 툴을 사용하여 연결 설정과 엔드 장치가 코디네이터에게 데이터를 전송함을 보였다.

참 고 문 헌

[1] Thomas Basmer, Henry Schomann, Steffen Peter, "Implementation Analysis of the IEEE 802.15.4 MAC for Wireless Sensor Networks," International Conference on Selected Topics in Mobile and Wireless Networking, 2011.

[2] 김광현, "무선 센서 네트워크란?", 계장 기술, 2009년 11월.

[3] IEEE Part 15.4: Wireless Meduim Access Control (MAC) and Physical Layer (PHY) Specification for Low-Rate Wireless Personal Area Networks , IEEE Std., Rev. IEEE Std 802.15.4-2006 and IEEE Std 802.15.4-2007a, 2007.

[4] Chienyuan Liu, "The design of home care assistant system by the Zigbee technology," Life Science Journal, Vol.6, No.2, 2009.

[5] "A True System-on-Chip Solution for 2.4GHz IEEE 802.15.4 and Zigbee Applications", Texas Instruments, Tech. Rep., 2011.

[6] Jean-Francois Wauthy and Laurent Schumacher, "Implementation of an IEEE 802.15.4-2006 Protocol Stack on the Texas Instrument CC2430," Proceedings of the 7th ACM workshop on Performance evaluation of wireless ad hoc, sensor, and ubiquitous networks, 2010.

[7] Naagesh S. Bhat, "Design and Implementation of IEEE 802.15.4 Mac Protocol on FPGA," Proceedings on Innovative conference on Embedded Systems, Mobile Communications and Computing, September 2011.

[8] Li-chun Ko, Young-chih Liu, Hua-wei Fang, "Design and Implementation of IEEE 802.15.4 Becon-enabled Network Devices," Proceedings of the 4th annual IEEE international conference on Pervasive Computing and Communications Workshops, 2006.

[9] "MAC Sample Application Software Design," Texas Instruments, Tech.. Rep., 2011.

[10] "802.15.4 MAC Application Programming Interface, " Texas Instruments, Tech. Rep., 2011.

[11] IAR Embedded Workbench for 8051 (www.iar.com/ew8051).

[12] "OS Abstraction Layer Application Programming Interface," Texas Instruments, Tech.. Rep., 2011.

[13] "802.15.4 MAC Application Programming Interface, " Texas Instruments, Tech.. Rep., 2011.

[14] "HAL Drivers Application Programming Interface," Texas Instruments, Tech.. Rep., 2011.

김 병 순

1993년 : 서강대학교 컴퓨터과(공학석사)
2003년 : 경북대학교 컴퓨터공학과(공학박사)
2003년~현재 : 안동대학교 정보과학교육과 부교수
관심분야 : DTN, USN, 멀티캐스팅