재구성형 주변장치유닛을 사용한 유무선 LED 조명 통신

Wired/Wireless LED Lighting Communication Using Reconfigurable Peripheral Unit

유 세 훈*, 공 정 철**, 김 기 철**

Sehoon Yoo*, Jungchul Gong**, Kichul Kim**

Abstract

In this paper, a reconfigurable peripheral unit for LED lighting communication is presented. Embedded lighting devices require various communication protocols. Usually, serial communication protocols and lighting control communication protocols such as DALI, DMX512, UART, SPI, IrDA, *etc.* are used in lighting devices. When the requirements of communication protocols are satisfied with separate IPs, the cost and the power consumption can considerably increase. We propose a reconfigurable communication peripheral unit which uses analysis of signal formats of the protocols. The gate count of the reconfigurable peripheral unit uses only 57% of the gate count of the separate implementation. Also, in this paper, a mapping table based DALI-ZigBee interfacing method for flexible lighting network configurations is proposed. Using this method, various DALI-ZigBee network systems can be easily set up. An LED lighting system platform is implemented to verify the operation of the DALI-ZigBee interfacing method. The reconfigurable peripheral unit and the DALI-ZigBee interfacing method can be efficiently used to implement various wired/wireless lighting communication systems.

요 약

본 논문에서는 LED 조명 통신을 위한 재구성형 주변장치유닛을 제안한다. 조명시스템을 위한 임베디드 장치에서는 다양한 통신 프로토콜이 요구된다. 이러한 통신 프로토콜로는 UART, SPI, IrDA 등의 직렬 통신, DALI나 DMX512와 같은 조명제어 통신이 있다. 다양한 통신 프로토콜 요구사항을 각 IP로 개별 구현하여 만족시킬 경우 비용 및 전력 효율이 떨어질 수 있다. 본 논문에서는 각 프로토콜의 신호 형식을 분석하여 UART, SPI, IrDA, DALI, DMX512의 통신 기능을 통합하는 방법을 제안한다. 제안한 재구성형 주변장치유닛은 개별 구현 방법에 비해 57% 감소된 양의 게이트수를 사용한다. 또한 본 논문에서는 조명 네트워크를 유연하게 구현하기 위해 사용될 수 있는 매핑테이블 기반의 DALI-ZigBee 인터페이싱 방법을 제안한다. 이 방법을 사용하면 DALI와 ZigBee를 혼용한 다양한 구성의 조명 네트워크 시스템을 효율적으로 구현할 수 있다. LED 조명시스템 플랫폼을 구성하여 유무선 조명 통신 네트워크의 동작을 검증하였다. 본 논문에서 제안한 재구성형 주변장치유닛과 DALI-ZigBee 인터페이싱 방법은 유무선 조명시스템을 효율적으로 구현하는 데에 사용될 수 있다.

Key words : Lighting Communication, Reconfigurable Peripheral, DALI, DMX512, Serial Communication.

* Dept. of Electrical and Computer Engineering,
University of Seoul

** CDS Power IC Group, Samsung Electro-
Mechanics Co. Ltd.
yugdoog@uos.ac.kr, 02-6490-5690

★ Corresponding author

Manuscript received Oct. 18, 2013; revised Oct. 28,
2013; accepted Oct. 28, 2013

1. 서론

LED를 조명에 적용할 경우, 에너지 고효율, 장기간의 사용 가능 수명, 빠른 동작 속도, 제어의 용이성, 고품질 색 재현 등의 장점이 있기 때문에 기존의 조명 장치에 비해 LED 조명시스템에 대한 수요가 점점

증가하고 있다. 건물에서 LED 조명 장치를 이용해 다양한 조명제어를 수행하기 위해서는 이를 효율적으로 관리하기 위한 디지털 조명제어 네트워크가 필요하다. 유명한 조명 통신 프로토콜로는 DALI나 DMX512가 있다. 이러한 조명 제어 기술을 지원하기 위해, 다양한 조명제어장치들이 개발되었다[1-4]. 대부분의 LED 조명제어장치는 보통의 SoC에서 가지고 있는 CPU, 메모리, 직렬통신 인터페이스와 LED를 모니터링하고 제어하기 위해 ADC, DAC, PWM 등의 주변장치를 가지고 있다. LED 조명제어장치를 다양한 환경의 조명시스템에 연결되어 사용하기 위해서는 다양한 주변장치유닛을 가지고 있는 것이 좋다. 하지만 많은 주변장치유닛을 하나의 조명제어장치에 모두 포함 시킬 경우, 지원하는 주변장치의 수가 증가함에 따라 비용과 전력소모가 크게 증가할 수 있다. LED 조명제어장치는 보통 임베디드 시스템으로 구성되기 때문에 비용과 전력소모의 증가는 효율적인 조명시스템 구현에 큰 문제가 될 수 있다. 이러한 문제를 해결하기 위해 [1]와 [4]에서는 재구성형 주변장치 구조를 제안하였다. 이 연구들에 나타나 있는 주변장치유닛은 높은 재구성 능력을 가지기 때문에 빠른 프로토타이핑에 강하다는 장점이 있다. 반면 실제 조명시스템에 사용될 때 조명시스템이 세팅되면 몇 가지의 주변장치 기능만 사용되는 경우가 많기 때문에 하드웨어 사용률(utilization)이 크게 떨어질 수 있다. 따라서 재구성이 가능하면서도 하드웨어 사용률이 크게 떨어지지 않는 구조에 대한 연구가 필요하다. [5]에서는 이를 위해 DMX512, UART, IrDA를 통합하는 구조를 보였다. 하지만 잘 알려진 조명 프로토콜인 DALI를 지원하지 않고 많이 사용되는 유선 직렬통신 중에 UART만 지원하기 때문에 실제 조명장치에 적용시킬 경우 추가적인 주변장치유닛을 필요로 할 수 있다.

고품질 조명제어시스템을 구성함에 있어 중요한 점으로 네트워크 구성의 유연성이 있다. DALI나 DMX512는 기본적으로 유선 조명네트워크를 사용한다. 건물에 조명시스템을 설치할 때 그 구조에 따라 조명제어라인을 구축하는 것은 큰 시간적/공간적 비용이 들 수 있다. 또한 조명장치의 배치 구조를 바꿀 경우 이에 대한 비용은 더욱 커질 수 있다. 따라서 무선 네트워크 기반의 조명시스템에 대한 관심이 커지고 있다. 무선 조명시스템을 구현하기 위한 솔루션들이 개발되어 있다[6-7]. 유무선 조명시스템을 구현하기 위해 [6]과 [7]에서는 마이크로프로세서 기반으로 제품을 개발하였다. DALI의 경우 고품질 조명제어를 위해 지속적인 모니터링과 타이밍 제어가 필요하다. 특히 DALI 표준[8]에 따르면 밝기변화시간(fade time)과 밝기변화율(fade rate)에 대한 세밀한

제어가 요구되기 때문에 나눗셈이 필요할 수 있다. 또한 LED 색 좌표 재현 및 연색성 제어 등의 기술을 적용시킬 경우 마이크로프로세서에 부담이 되어 처리가 힘들 수 있다. 따라서 유선 시스템과 무선 시스템을 연동시킬 때 연산량이 적은 방법을 사용하는 것이 좋다.

본 논문에서는 LED 조명 통신을 위한 재구성형 주변장치 유닛을 제안한다. 본 논문의 재구성형 주변장치 유닛은 다양한 통신 프로토콜의 신호 유사성을 이용하여 저비용으로 DALI, DMX512, UART, SPI, IrDA의 기능과 각 통신 프로토콜 간의 브리지 기능을 지원한다. 재구성형 주변장치유닛을 사용하면 각 조명장치의 비용이 크게 절감될 수 있다. 또한 본 논문에서는 매핑테이블을 사용한 효율적 DALI-ZigBee 인터페이싱 방법을 제안한다. 이 방법은 처리 데이터 양과 연산량이 적기 때문에 다양한 상황에서 유선조명네트워크와 무선조명네트워크를 저비용으로 연결할 수 있다. 본 논문에서 제안되는 재구성형 주변장치 유닛과 DALI-ZigBee 인터페이싱 방법을 사용하면 유무선 조명시스템을 효율적으로 구축할 수 있다.

본 논문의 나머지 부분은 다음과 같이 구성된다. 2장에서는 재구성형 주변장치유닛에 대해서 설명한다. 3장에서는 DALI-ZigBee 인터페이싱 방법을 보인다. 4장에서는 재구성형 주변장치유닛의 비용 감소에 대해서 보이고, DALI-ZigBee 인터페이싱 방법에 대한 검증 플랫폼과 검증 결과에 대해서 보인다. 마지막으로 5장에서 본 논문의 결론을 맺는다.

II. 재구성형 주변장치유닛

본 장에서는 제안하는 재구성형 주변장치유닛의 구조를 설명한다. 본 논문에서는 각 주변장치의 통신 신호를 분석하여 서로간의 유사성을 찾고 이를 이용하여 다양한 주변장치유닛을 하나의 하드웨어 구조로 지원한다. 2.1절에서는 [5]에서 연구된 DMX512, UART, IrDA의 통합 방법을 소개한다. 2.2절과 2.3절에서는 DMX512, UART, IrDA 통합 구조에 SPI와 DALI를 추가하기 위한 구조를 설명한다.

2.1 DMX512, UART, IrDA의 통합

DMX512의 신호가 그림 1에 나타나 있다. 그림 1에 나타나 있는 것처럼 DMX512 프레임은 Break로 시작된다. 첫 번째 슬롯데이터는 커맨드의 역할을 하고 뒤이어 조명의 디밍값(dimming value)을 연속적으로 보낸다. 보내는 슬롯데이터의 개수는 제어하고자 하는 조명장치의 개수에 따라 결정된다[9]. Break의 최소 지속시간(duration)은 92us로서 이 부분을 제외한

나머지 부분은 일반적인 UART에서 생성가능한 신호이다. 따라서 [5]에서는 Break를 생성하기 위해 수정된 UART 송신기(transmitter)를 사용하였다. DMX512 수신 시, UART 수신기(receiver)를 사용하여 신호를 받는다. 이 때 Break는 2개의 정지 비트(stop bits)가 0인 프레임 오류(frame errors)를 통해 판정한다. 이에 대한 처리는 제어 유닛(control unit)에서 이루어진다.

Fig. 1. DMX512 Signal
그림 1. DMX512 신호

Fig. 2. Comparison between signals of IrDA SIR and UART
그림 2. IrDA SIR 신호와 UART 신호의 비교 [5]

Fig. 3. IrDA SIR reception signal and its latched signal
그림 3. IrDA SIR 수신 신호와 지연된 신호 [5]

Fig. 4. Integrated structure of DMX512, UART, and IrDA
그림 4. DMX512, UART, IrDA의 통합 구조 [5]

그림 2에 IrDA SIR 신호와 UART 신호가 비교되어 있다[10]. 그림에 나타나 있는 것처럼 IrDA의 경우 송신신호와 수신신호가 반전되어 있다. IrDA SIR 신호를 송신하기 위해 [5]에서는 비트의 지속시간 제어를 담당하는 카운터(counter)를 IrDA 부분과 UART 간에 공유하는 방법을 사용하였다. IrDA SIR 신호를 수신하기 위해 그림 3에 나타나 있는 것처럼 IrDA 신호의 falling edge마다 비트 지속시간만큼 지연시킴으로써 그림 2의 UART 신호와 동일한 신호를 만들었다. 이를 통해 UART 수신기를 그대로 사용할 수 있다. 그림 4에 위에서 설명한 내용이 반영된 DMX512, UART, IrDA가 통합 구조를 보였다[5].

2.2 DMX512, UART, IrDA, SPI의 통합

SPI의 신호가 그림 5에 나타나 있다. SPI는 de facto 표준으로서 통일된 국제표준이 존재하지 않는다. 따라서 구현에 따라 전송 속도, 데이터프레임의 비트수, 통신 클럭(SCK)의 CPOL과 CPHA(clock polarity and phase) 등이 다를 수 있다. 본 논문에서 SPI 모듈의 비트수는 8비트이고 지원하는 전송 속도는 UART의 전송 속도를 따랐다. 그림 5에 나타나 있는 것처럼 SPI는 CPOL과 CPHA에 따라 4가지 모드가 있다[11]. 단일 모드의 SPI 모듈은 단순한 시프트레지스터로 구현될 수 있다. 하지만 4가지 모드를 동일한 방법으로 구현할 경우 CPOL에 대응하기 위해 두 가지 클럭 에지(clock edge)에서 동작하기 위한 레지스터 지원과 CPHA에 대응하기 위해 데이터샘플링(data sampling)과 데이터토글링(data toggling)에 대한 제어가 추가적으로 필요하다.

본 논문에서는 추가적인 클럭 에지 레지스터와 추가적인 제어가 필요 없는 SPI 슬레이브 모듈 구조를 제안한다. 본 논문에서 설계한 SPI 슬레이브 모듈의 구조가 그림 6에 나타나 있다. 시프트레지스터의 동작 클럭은 SPI 모듈의 내부 클럭을 사용한다. 대신 데이터의 토글링 시점을 결정하기 위해 SCK의 rising edge와 falling edge의 탐지기를 사용한다. 이 탐지기

Fig. 5. SPI signals depending on CPOL and CPHA
그림 5. CPOL과 CPHA에 따른 SPI 신호

Fig. 6. SPI slave module structure

그림 6. SPI 슬레이브 모듈 구조

의 출력은 시프트레지스터의 시프트 enable 입력으로 사용된다. 그림 5에서 (CPOL,CPHA)의 값이 (0,1) 또는 (1,0)이면 SCK가 rising edge 일 때 데이터가 토글링되고, (0,0) 또는 (1,1)이면 falling edge 일 때 데이터가 토글링된다. 따라서 시프트 enable 신호는 CPOL과 CPHA에 따라 멀티플렉서로 선택되어 사용된다. CPHA가 0인 경우 설정된 SCK 에지마다 시프트 시키면 그림 7(a)에 나타나 있는 것처럼 SCK 8사이클 동안 통신이 제대로 완료된다. 하지만 CPHA가 1인 경우에는 그림 7(b)에 보인 것처럼 첫 번째 비트가 더미비트이기 때문에 마지막 비트가 저장되지 않는 상태로 시프트가 끝나게 된다. 이를 위해 그림 6에 보인 것처럼 SS 신호의 rising edge를 탐지해 이 신호도 시프트 enable 신호로 사용한다. 이 신호는 데이터통신의 종료를 알리는 VALID 신호의 생성 로직을 공유하여 사용된다. 또한 시프트레지스터를 1비트 증가시켜 사용한다. 즉, 시프트는 9번 이루어지게 되며 SPI 통신 결과는 그림 8에 나타나 있는 것처럼 CPHA가 0인 경우 상위 8비트, CPHA가 1인 경우 하위 8비트가 된다.

그림 6의 SPI 구조에서 가장 큰 부분을 차지하는 로직은 시프트레지스터이다. 그림 4의 구조는 UART를 기반으로 설계되었다. UART 같은 직렬 통신 인터페이스 모듈은 1비트씩 통신하기 때문에 보통 시프트레지스터를 내부에 포함하고 있다. 따라서 그림 6

Fig. 7. SPI communication results using SCK

그림 7. SCK를 사용한 SPI 통신 결과

Fig. 8. SPI communication results using SCK and SS

그림 8. SCK와 SS를 사용한 SPI 통신 결과

의 시프트레지스터는 UART의 시프트레지스터를 공유하여 구현될 수 있다. DMX512/UART/IrDA/SPI 통합 구조가 그림 9에 나타나 있다. UART 송신기와 수신기 모두 시프트레지스터를 갖고 있지만 여기서는 송신기의 시프트레지스터를 사용하여 SPI 기능을 통합시켰다. 제안하는 SPI 모듈의 구조는 추가적인 게이트가 필요 없으므로 기존의 제어 로직은 SPI 모드 추가를 제외하고는 수정되지 않는다. 4가지 CPOL/CPHA 모드를 지원하는 SPI의 기능이 저비용으로 설계되었고 그 중 가장 큰 부분을 차지하는 시프트레지스터에 대한 추가 게이트가 없으므로 SPI가 효율적으로 통합된다.

Fig. 9. Integrated structure of DMX512, UART, IrDA, and SPI

그림 9. DMX512, UART, IrDA, SPI의 통합 구조

2.3 DMX512, UART, IrDA, SPI, DALI의 통합

DALI는 DMX512와 같이 잘 알려진 조명 통신 프로토콜로서 전송 속도는 1.2kbps이다. DALI의 신호 형식이 그림 10에 나타나 있다[8]. 그림 10에서 forward frame은 DALI 마스터가 DALI 슬레이브에 명령을 내릴 때의 신호 형식이고 backward frame은 DALI 슬레이브가 DALI 마스터에 응답할 때 사용되는 신호 형식이다. DALI의 각 비트는 가운데에서 rising edge가 있으면 '1'을 나타내고 falling edge가 있으면 '0'을 나타낸다. 정지 비트는 일반적인 NRZ(non-return-to-zero) 부호 형식을 사용한다.

Fig. 10. DALI signals
그림 10. DALI 신호

Fig. 11. 1.2kHz signals of CLK_TX, UART, and DALI
그림 11. 1.2kHz 속도의 CLK_TX, UART, DALI 신호

(a) Simplified state diagram of UART transmitter

(b) Signals for each state

Fig. 12. Relationship between states of UART transmitter and signals of CLK_TX, UART, and DALI
그림 12. UART 송신기의 상태와 CLK_TX, UART, DALI 신호의 연관성

DALI 신호를 생성하기 위해 1.2kHz의 클럭, UART 신호, DALI 신호를 그림 11에 비교하였다. 1.2kHz 클럭은 UART를 1.2kbps로 설정 후 UART 송신기 내부적으로 생성된다. CLK_TX 신호와 UART 신호를 XOR 시킨 신호와 DALI 신호는 시작 비트와 정지 비트를 제외하고 동일하다. 따라서 시작 비트와 정지 비트에 대한 예외 처리가 수행되면 DALI 신호를 생성하기 위해 기존의 UART 송신기를 그대로 사용할 수 있다. XOR된 신호에서 시작 비트는 DALI 신호가 반전된 결과 신호와 동일하다. 정지 비트는 UART 신호와 동일하다. 이 특징을 사용하면 적절한 타이밍에 XOR된 신호, 반전된 XOR 신호, UART 신호 중 하나를 선택하면 DALI 신호를 생성할 수 있다. 그림 12에 UART 송신기의 간략화된 상태도(state diagram)를 나타내고, 각 상태에 대해 신호를 구분하여 나타냈다. 기존의 UART의 경우 시작 비트는 0으로 정지 비트를 1로 설정하면, 부호화 방법은 데이터 비트와 동일하기 때문에 START_BIT 상태와 STOP_BIT 상태가 추가적으로 필요하지 않다. 하지만 다양한 프로토콜의 신호 형식을 지원하기 위해서는 시작 비트와 정지 비트에 대한 개별적인 제어가 필요하기 때문에 본 논문의 UART에서는 이 상태들을 구분하였다. 그림 12(a)에 나타나 있는 UART 송신기의 FSM (finite state machine)에서는 세 개의 카운터를 사용한다. 먼저 CNT1BIT는 1비트의 지속 시간을 유지하기 위한 카운터다. CNT8은 0~7 범위의 카운팅을 수행하는 카운터로서 데이터 비트를 카운팅할 때 사용된다. CNTSTOP은 정지 비트를 카운팅할 때 사용된다. FSM에서 사용된 FIN_CNTXXX 신호는 각 카운터의 카운팅 종료 신호이다. 그림 12(b)에 보인 것처럼 UART 송신기의 각 상태들에 의해 시작 비트 구간, 데이터 비트 구간, 정지 비트 구간, 대기 신호 구간이 구분될 수 있다. 따라서 UART 송신기의 FSM의 상태값을 참조하여 타이밍 제어를 수행하면 DALI 신호를 쉽게 생성할 수 있다. CLK_TX 신호의 진동(oscillation) 시점도 비슷한 방법으로 상태값을 사용해 제어된다.

DALI 신호를 수신할 때, DALI 신호 생성 및 송신과 달리 CLK_TX와 UART 신호를 내부적으로 생성할 수 없다. 따라서 수신되는 DALI 신호만으로 복호를 해야 한다. DALI 신호는 정지 비트를 제외하고 각 비트의 지속시간 중앙에서 에지가 발생한다. 이 에지가 rising edge면 비트 '1'이고 falling edge면 비트 '0'이 된다. 이 특징을 이용하기 위해 그림 13에 DALI 신호의 에지를 세 가지로 분류하였다. DALI-2 신호는 DALI 신호에서 데이터 비트 지속시간의 중간 에지마다 신호값을 1비트 지속시간 동안 유지

Fig. 13. Signal change from DALI to UART

그림 13. DALI에서 UART로의 신호 변경

(latching)한 것이다. 이 때 rising edge에서는 '1'값을 유지하고 falling edge에서는 '0'값을 유지한다. DALI-2 신호와 UART 신호를 비교하면 음영으로 표현된 시작 비트 부분을 제외하고는 동일함을 알 수 있다. 시작 비트 부분을 수정하기 위해 DALI 신호의 첫 번째 falling edge와 rising edge를 사용한다. 즉, 기본 상태인 '1'을 유지하고 있다가 첫 번째 falling edge와 rising edge가 발생하면 '0'값을 1비트 지속시간 동안 유지시키는 것이다. 그 결과 DALI-3 신호와 같은 신호를 얻을 수 있다. 이는 UART 신호와 동일하므로 UART 수신기를 통해 DALI 신호를 복호할 수 있다.

DALI 마스터는 16비트 송신기와 8비트 수신기를 사용하고 DALI 슬레이브는 8비트 송신기와 16비트 수신기를 사용한다. 이를 위해 본 논문의 UART 송신기와 수신기에서는 16비트/8비트 두 가지 비트수를 지원한다. 이는 데이터 비트를 카운팅 하기 위한 카운터의 비트를 1비트 증가시킴으로써 구현된다.

그림 14에 본 논문에서 제안하는 재구성형 주변장치 유닛의 구조가 나타나 있다. DALI TX Selection 유닛과 Edge Latching 유닛은 위에서 설명한 DALI 신호 송수신 방법을 적용시킨 유닛이다. DALI TX Selection 유닛과 Edge Latching 유닛의 동작을 그림 15에 보였다.

제안하는 재구성형 주변장치 유닛은 DALI, DMX512, UART, IrDA SIR, SPI의 송수신기의 기능을 하나의 구조로 지원할 수 있다. 또한 송신기와 수신기가 구분되어 설계되었기 때문에 각 프로토콜 간의 브리지의 기능도 지원한다. 그림 16에 LED 조명 장치의 SoC 구조가 나타나 있다. 기존에는 다양한 프로토콜을 지원하기 위해 개별 모듈을 사용하기 때문

에 해당 프로토콜을 처리하기 위한 모듈의 수가 크게 증가한다. 재구성형 주변장치 유닛을 사용할 경우 모듈의 개수를 크게 줄일 수 있으므로 효율적인 LED 조명장치 구현이 가능하다.

Fig. 14. Proposed reconfigurable peripheral unit

그림 14. 제안된 재구성형 주변장치 유닛

```

Output DALI_tx_selection( TX_UART, CLK_TX, MODE, TX_STATE )
{
    if ( MODE == ( DMX512 || UART || IrDA || SPI ) )
    {
        return TX_UART;
    }
    else // MODE == ( DALI_Master || DALI_Slave )
    {
        if ( TX_STATE == START_BIT )
            return !(CLK_TX xor TX_UART);
        else if ( TX_STATE == DATA_BIT )
            return (CLK_TX xor TX_UART);
        else // TX_STATE == ( STOP_BIT || READY )
            return TX_UART;
    }
}

RX_UART edge_latching( Input, MODE, RX_STATE )
{
    if ( MODE == ( DMX512 || UART || SPI ) )
    {
        return Input;
    }
    else if ( MODE == IrDA )
    {
        if ( falling_edge( Input ) )
            return latch_1bit( Input );
        else
            return Input;
    }
    else // MODE == ( DALI_Master || DALI_Slave )
    {
        if ( RX_STATE == READY )
        {
            if ( first_two_edges() )
            {
                flag = 1;
                return latch_1bit( Input ); // ignore middle edge
            }
            else
                flag = 0;
        }
        else if ( flag == 1 && ( rising_edge( Input ) || falling_edge( Input ) ) )
            return latch_1bit( Input ); // ignore middle edge
        else
            return Input;
    }
}

```

Fig. 15. DALI_TX_Selection unit operation and Edge_Latching unit operation

그림 15. DALI_TX_Selection 유닛과 Edge_Latching 유닛의 동작

Fig. 16. Comparison between LED controller architectures
 그림 16. LED 조명제어장치의 구조 비교

III. DALI-ZigBee 인터페이싱

본 장에서는 유무선 조명 네트워크를 효율적으로 구현하기 위한 DALI-ZigBee 인터페이싱 방법을 설명한다. DALI와 ZigBee를 연결하기 위해서 먼저 3.1절에서 DALI와 ZigBee의 네트워크 토폴로지를 살펴본다. 3.2절에서 존재할 수 있는 유무선 네트워크 형태를 분석하고 DALI-ZigBee 인터페이싱 방법을 설명한다.

Fig. 17. Network topologies of DALI and ZigBee
 그림 17. DALI와 ZigBee의 네트워크 토폴로지

3.1 DALI와 ZigBee의 네트워크 토폴로지

DALI는 그림 17(a)에 나타나 있는 것처럼 두 개의 라인을 사용하는(two-wired) 공유 버스(shared bus) 형태를 갖고 있다. 버스의 마스터는 하나만 존재할 수 있고 슬레이브는 최대 64개까지 존재할 수 있다. 조명시스템에서 명령어(command)를 조명장치에 전송해 조명제어를 수행하는 기기는 마스터로 지정되고 조명장치들은 슬레이브로 지정된다[8]. 그림 17(b)에 나타나 있는 것처럼 ZigBee는 스타(star), 트리(tree), 그물망(mesh) 등 다양한 네트워크 토폴로지를 지원한다. 장치 종류로는 크게 full function device(FFD)와 reduced function device(RFD)로 이루어져 있다. FFD는 라우팅 기능을 지원한다. FFD 중 전체 네트워크 관리 역할을 맡은 장치를 coordinator라 하고 이 네트워크를 PAN(personal area network)라 한다[12].

3.2 DALI-ZigBee 인터페이싱 방법

ZigBee를 사용하여 DALI 기반 조명시스템을 구축할 때, 크게 조명시스템제어를 기존에 개발된 DALI 마스터를 연결하여 사용하는 경우와 ZigBee 모듈에 DALI 마스터의 기능을 포함시키는 경우로 구분할 수 있다. 그림 18에서 왼쪽은 마스터 측의 네트워크 구성을 나타내고 있고 오른쪽은 슬레이브 측의 네트워크 구성을 나타내고 있다. DALI 마스터 측은 실제 조명시스템이 구현될 때 마스터 측은 기존 유선 DALI 마스터를 ZigBee 네트워크에 연결할 경우(그림 18(a)), 기존의 유선 DALI 네트워크를 ZigBee 네트워크에 연결할 경우(그림 18(b)), DALI 마스터의 기능이 ZigBee 기기 내부에 구현될 경우가 있다. 슬레이브들은 필요에 따라 DALI 슬레이브가 ZigBee 기기 내부에 직접 구현하거나, 기존의 DALI 슬레이브 하나를 사용하거나, 기존의 DALI 슬레이브 네트워크에서 중간 라인을 ZigBee로 대체한 형태를 가질 수 있다. 기존의 유선 DALI 네트워크의 경우 마스터가 신호를 주면 모든 조명장치가 그 신호를 직접 받고 ZigBee의 경우 coordinator가 신호를 주면 조명장치들이 FFD의 라우팅을 거쳐 간접적으로 받아야 할 수 있다. 이러한 라우팅 기능은 ZigBee 스택의 네트워크 계층을 구현함으로써 지원될 수 있다. 그림 18에서는 라우팅 과정이 생략되어 있다.

기존의 유선 DALI 마스터를 ZigBee 네트워크로 변경할 때 그림 18(a)에 보인 것처럼 DALI 네트워크와 ZigBee 네트워크를 연결하는 기능을 하는 DALI-ZigBee 변환기(DALI-ZigBee converter)가 필요하다. 기존에 개발된 DALI 마스터가 ZigBee 네트워크의 유무와 관계없이 동작하기 위해서 그림 19와 같은 추상화 계층이 있어야 한다. 이 추상화 계층을 통해

DALI 마스터는 자신에게 연결된 DALI 슬레이브들이 직접 유선으로 연결된 것처럼 동작하면 된다. DALI 슬레이브 측도 마찬가지로 자신이 기존의 유선 DALI 네트워크에 연결된 것처럼 동작하기 위해 추상화 계층이 필요하다. 이 추상화 계층의 역할을 DALI-ZigBee 변환기에서 수행한다. 그림 18(a)의 조명시스템에서 DALI와 ZigBee 간의 인터페이싱 과정은 다음과 같다.

Fig. 18. DALI-ZigBee based Wired/Wireless lighting network
그림 18. DALI-ZigBee 기반 유무선 조명 네트워크

Fig. 19. DALI-ZigBee Abstraction Layers
그림 19. DALI-ZigBee 추상화 계층

[STEP1] DALI 마스터 측의 변환기를 coordinator로 하여 무선 DALI 슬레이브 장치와 DALI 슬레이브 변환기를 이 coordinator의 PAN에 참여(join)시킨다. 그 결과, DALI 마스터 측의 변환기에는 이 PAN에 참여된 ZigBee 기기들의 주소값과 각 ZigBee 기기와 연결된 DALI 슬레이브의 개수가 저장된다. 만일 연결되는 ZigBee 기기가 여러 개의 DALI 슬레이브를 가지고 있는 DALI 슬레이브 측 변환기면 연결된 DALI 슬레이브 개수 만큼의 ZigBee 주소값을 가지고 있다. [STEP2] DALI 마스터는 DALI 표준[8]에 지정되어

있는 초기화 명령어를 DALI 마스터 변환기에 준다. 이 명령어는 각 DALI 슬레이브에 주소값을 임의로 할당하기 위한 명령어로서 여기에서는 DALI 마스터 측 변환기가 DALI 슬레이브처럼 동작한다. 이 명령어의 결과로 DALI 마스터 측 변환기는 전체 조명 네트워크에 있는 DALI 슬레이브 개수만큼의 가상 주소값을 저장하게 된다.

[STEP3] DALI 마스터 측 변환기는 [STEP1]에서 얻은 ZigBee 주소값과 각 ZigBee 기기에 연결된 DALI 슬레이브 개수, [STEP2]에서 얻은 DALI 가상 DALI 주소값을 사용하여 DALI-ZigBee 매핑을 수행한다. 그 결과, DALI 마스터 측 변환기는 가상 DALI 슬레이브 주소값과 ZigBee 기기들의 주소값 간의 변환을 할 수 있는 매핑테이블을 저장하게 된다.

[STEP4] 각 DALI 슬레이브 측 ZigBee 기기에서는 자신에게 연결된 DALI 슬레이브에 주소값을 할당하기 위해 DALI 초기화 명령어를 준다. 이 때 DALI 슬레이브 측 ZigBee 기기는 연결된 DALI 슬레이브들에 대해 DALI 마스터의 역할을 한다. 이 명령어의 결과로 DALI 슬레이브 측 ZigBee 기기는 자신에게 연결된 DALI 슬레이브들의 실제 주소값을 저장한다.

[STEP5] 각 DALI 슬레이브 측 ZigBee 기기에서는 자신이 가지고 있는 실제 DALI 슬레이브 주소값과 ZigBee 주소값을 임의대로 매핑시킨다. 그 결과, DALI 슬레이브 측 ZigBee 기기는 자신의 ZigBee 주소값과 자신에게 연결된 실제 DALI 슬레이브 주소값 사이의 매핑테이블을 저장하게 된다.

위에서 설명한 것처럼, DALI-ZigBee 추상화 계층의 역할을 수행하기 위해 DALI 마스터 측 변환기에서는 DALI 슬레이브의 역할을 수행하고 DALI 슬레이브 측 변환기에서는 DALI 마스터 역할을 수행한다. 즉 논리적으로는 1개의 DALI 네트워크지만 물리적으로는 변환기 개수 만큼의 DALI 네트워크가 있는 것과 같다. 따라서 각 변환기에 있는 물리적 DALI 네트워크 안에서는 기존의 유선 DALI 네트워크와 동일한 동작 수행이 가능하다. DALI-ZigBee 인터페이싱 과정이 끝나면 조명시스템이 그림 20에 나타나 있다. 그림에서 각 ZigBee 기기에는 자신의 상태 및 매핑테이블이 저장되어 있다.

DALI에서 조명을 제어할 때의 명령어는 크게 5가지로 나눌 수 있다.

- 초기화 (슬레이브 주소값 할당)
- 조명 밝기 제어
- 장면(scene) 및 그룹 세팅
- 슬레이브 상태 체크
- 기타 (장치 리셋 또는 특정 레지스터에 값 저장)

여기에서 첫 번째 명령어는 위의 DALI-ZigBee 인터페이싱 과정에서 사용되는 명령어다. 조명 밝기 제어, 장면/그룹 세팅, 기타 명령어는 DALI 마스터에 DALI 슬레이브로 명령을 내리기만 하고 DALI 슬레이브로부터 특별한 응답(response)을 받지 않는다. 슬레이브 상태 체크 명령어는 DALI 마스터가 특정 슬레이브의 상태 정보를 요구하면 슬레이브에서 해당 상태 정보를 응답의 형태로 돌려준다. 그림 21은 응답이 있는 경우의 조명시스템 동작을 표현하고 있다. 응답이 없는 경우의 동작은 그림에서 명령어가 타겟 DALI 슬레이브에 도달할 때까지와 같다. 장면과 그룹을 통한 조명장치 제어는 ZigBee의 그룹 어드레싱(group addressing) 방법으로 처리 가능하므로 여기서는 단일 슬레이브에 대한 직접 어드레싱(direct addressing)을 사용한 조명제어에 대해서만 살펴본다. 그림에서 보는 것과 같이 본 논문에서 제안하는 DALI-ZigBee 인터페이싱 방법을 사용하면 각 ZigBee 기기에 저장된 매핑테이블을 참조함으로써 DALI-ZigBee 연동이 가능하다. 각 ZigBee 기기가 가상의 DALI 슬레이브 또는 가상의 DALI 마스터로 동작해야 하는 경우는 초기화 때 뿐이고 보통의 경우 명령어의 주소 필드만 대체하여 DALI 마스터 명령어를 전달만 하면 되기 때문에 연산량이나 메모리에 저장해야 하는 데이터량의 관점에서 효율적이다.

그림 18(c)에 대한 DALI-ZigBee 인터페이싱은 위에서 설명한 그림 18(a)의 경우와 동일하게 처리 가능하다. 그림 18(b)의 경우 DALI 마스터와 유선으로 연결된 DALI 슬레이브는 기존의 DALI 프로토콜을 따르고, 나머지 DALI 슬레이브들과의 인터페이싱을 그림 18(a)의 경우와 동일한 방법으로 처리하면 된다.

Fig. 21. DALI-ZigBee lighting system operation

그림 21. DALI-ZigBee 조명시스템 동작

IV. 하드웨어 비용 및 검증 실험

본 논문에서 제안한 재구성형 주변장치유닛은 DALI, DMX512, UART, IrDA, SPI 모듈로 재구성하여 사용할 수 있다. 또한 각 모듈 간의 변환기로도 사용될 수 있다. 제안한 재구성형 주변장치유닛과 DALI, DMX512, UART, IrDA, SPI를 개별적으로 구현했을 때의 게이트수를 표 1에 비교하였다. 표에서 보는 것처럼 재구성형 주변장치유닛은 개별 구현 방법에 비해 약 57% 감소된 양의 게이트수를 사용한다.

본 논문에서 제안한 DALI-ZigBee 인터페이싱 방법을 검증하기 위해 그림 22와 같은 검증 플랫폼을 구축하였다. 검증 플랫폼에서 사용된 ZigBee 기기는 삼성 ZBS240이고 FPGA 보드는 Dynalith사의 iNTUITION이다. DALI 마스터는 ZigBee 기기 내부에 구현되었고 호스트 PC에 의해 제어된다. 슬레이브 측은 8개의 DALI 슬레이브가 있는 유선 DALI 네트워크로 이루어져 있다. 이 DALI 네트워크에 DALI-ZigBee 변환기가 연결되어 DALI-ZigBee 인터페이싱을 담당한다. LED 조명제어 테스트를 위해 하우스모델을 구축하여 검증 데모를 진행하였다. 그림

Fig. 20. DALI-ZigBee interfacing result using mapping tables

그림 20. 매핑테이블을 통한 DALI-ZigBee 인터페이싱 결과

23은 구축된 조명시스템 플랫폼을 사용한 검증 데모 결과다. DALI 마스터에 검증 시나리오를 프로그램하여 데모가 진행되었다. 검증 데모 결과 DALI-ZigBee 유무선 통합 조명시스템이 제대로 동작함을 확인하였다.

Fig. 22. DALI-ZigBee Lighting System Platform
그림 22. DALI-ZigBee 조명시스템 플랫폼

Fig. 23. DALI-ZigBee Lighting System Demo

그림 23. DALI-ZigBee 조명시스템 데모

Table 1. Comparison between separate implementation method and proposed integrated implementation method for peripheral units

표 1. 주변장치유닛의 개별 구현 방법과 제안된 통합 구현 방법의 비교

	Separate implementation	Proposed
Gate count	7594	3272

V. 결론

본 논문에서는 효율적인 LED 조명시스템 구축을 위한 재구성형 주변장치유닛과 DALI-ZigBee 인터페이스 방법을 제안하였다. 제안한 재구성형 주변장치유닛은 통신 프로토콜의 신호 분석을 통해 DMX512, DALI, UART, IrDA SIR, SPI의 기능을 저비용으로 통합하였다. 또한 송신기 측과 수신기 측을 구분하여 구현하였기 때문에 각 프로토콜 간의 브리지의 기능도 수행할 수 있다. 재구성형 주변장치유닛은 각 프로토콜을 개별적으로 구현했을 때보다 약 57% 만큼 적은 양의 게이트수를 사용한다. 따라서 재구성형 주변장치유닛을 사용하면 각각의 LED 조명제어장치의 구현 비용을 크게 줄일 수 있다. 제안한 DALI-ZigBee 인터페이스 방법은 매핑테이블을 사용하여 DALI와 ZigBee의 유무선 네트워크를 구축할 수 있다. 이 방법을 사용하면 적은 데이터 처리량과 연산량 만으로 다양한 상황에서 DALI와 ZigBee 간 네트워크 인터페이스가 가능하기 때문에 유무선 조명네트워크를 저비용으로 구축할 수 있다. DALI-ZigBee 인터페이스 방법은 LED 유무선 조명시스템 플랫폼을 구축하여 동작을 검증하였다. 제안하는 재구성형 주변장치유닛과 DALI-ZigBee 인터페이스 방법은 유무선 혼합 조명시스템을 구현하는 데에 효율적으로 사용될 수 있다.

References

- [1] Cypress Semiconductor, "PowerPSoC intelligent LED driver," <http://www.cypress.com>, April 2012.
- [2] NXP Semiconductors, "Single-chip 16-bit/32-bit microcontrollers: 8kB/16kB/32kB flash with ISP/IAP, fast ports and 10-bit ADC," <http://www.nxp.com>, June 2009.

- [3] STMicroelectronics, "8-bit MCU with single voltage flash memory, data EEPROM, ADC, timers, SPI, DALI," <http://www.st.com>, Nov. 2004.
- [4] M. K. Saxena, E. Bhatnagar and S. N. Kulkarni, "Reconfigurable architecture for IP peripherals," International Conference on Signals and Electronic Systems, Poland, pp.347-350, Sep. 2010.
- [5] S. Yoo, S. Song, C. Park, J. Gong, and K. Kim, "Multi-function unit for LED lighting," International SoC Conference (ISOCC), pp. 447-450, Nov. 2012.
- [6] Orama, "Wi-Fi DALI Integrated LED Lighting Control System," <http://www.oramainc.com>, 2013.
- [7] Wi-Light, "Wi-Light: a wireless light controller family," <http://www.wi-light.com>, Dec. 2010.
- [8] IEC, "Digital addressable lighting interface - Part 102: General requirements : Control gear," International Electrotechnical Commission, June 2009.
- [9] ESTA, "American national standard E1.11-2008 entertainment technology, USITT DMX512-A: asynchronous serial digital data transmission standard for controlling lighting equipment and accessories," Entertainment Services and Technology Association, April 2008.
- [10] IrDA, "Serial infrared physical layer specification version 1.4," Infrared Data Association, May 2001.
- [11] Motorola, "SPI block guide V04.01," <http://www.freescale.com>, July 2004.
- [12] ZigBee Alliance, "ZigBee specification," <http://www.zigbee.org>, Jan. 2008.

BIOGRAPHY

Sehoon Yoo (Student Member)

2006 : BS degree in Electrical and Computer Engineering, University of Seoul.
 2008 : MS degree in Electrical and Computer Engineering, University of Seoul.
 2008~ : Working toward the PhD degree in Electrical and Computer Engineering, University of Seoul.

Jungchul Gong (Member)

2000 : BS degree in Electrical and Material Engineering, Kwangwoon University.
 2010 : MS degree in Electronics and Computer Engineering, Hanyang University.
 2000~ : Senior Engineer, CDS Power IC Group, Samsung Electro-Mechanics Co. Ltd.

Kichul Kim (Member)

1982 : BS degree in Electrical Engineering, Seoul National University.
 1984 : MS degree in Electrical Engineering, Seoul National University.
 1991 : PhD degree in Electrical Engineering, University of Southern California.
 1984~1994 : Senior Research Engineer, Electronics and Telecommunications Research Institute (ETRI).
 1994~ : Professor, in Electrical and Computer Engineering, University of Seoul.