

# WPC KI Protocol의 전력전송단계에서의 Power Scheduling에 관한 연구

임수환, 박용주\*

\*한국전자기술연구원

ish0723@keti.re.kr, \*suede8247@keti.re.kr

## A Study on the Power Scheduling in Power Transfer Phase of the WPC KI Protocol

Lim Su-Hwan, Yong-Ju Park\*

\*Korea Electronics Technology Institute

### 요 약

본 논문은 무선전력위원회(WPC)의 KI 표준에서 정의하는 프로토콜에 대한 논문이다. WPC는 Qi 다음 표준으로 KI를 개발 중에 있다. KI는 2.4kW급 cordless kitchen을 타겟으로 하는 무선전력전송 표준이다. WPC는 KI 표준에서는 안전으로 위해 송수신간의 통신을 하도록 되어있으며, 이는 NFC를 이용하도록 되어 있다. 크게 Idle, Configuration, Connected, Power Transfer의 4개의 단계로 구성되어 있으며, 특히 Power Transfer 단계는 FOD slot, Comm slot, Power Transfer 3가지 상태로 구성되어 있다. 본 논문에서는 이 3가지 상태의 scheduling에 대해 연구하였다.

### I. 서 론

WPC는 Wireless Power Consortium 즉, 무선전력위원회로 무선전력전송과 관련된 표준화 활동이 주된 업무이다. WPC가 개발한 Qi표준은 휴대폰 무선충전을 주축으로 15W이하의 각종 디바이스를 충전하는 표준으로 안정성과 호환성에서 우수성을 선보여 다양한 회사들로부터 각광받으며 휴대폰 무선충전의 주류표준으로 자리했다.

WPC는 이어서 KI라는 2.4kW 대역의 cordless kitchen을 타겟으로 하는 새로운 표준을 연구 중에 있다. KI에서는 송수신간의 통신으로 NFC를 활용하는 프로토콜을 정의하고 있다. 해당 프로토콜은 크게 Idle, Configuration, Connected, Power Transfer의 4개의 단계(Phase)로 구성되어 있으며, 각각 수신단 대기, 기본설계값 전송, 전력관련 설정 협상, 전력전송의 역할을 수행한다.

특히, Power Transfer 단계는 전력전송 도중 FOD slot과 Comm slot으로 금속이물질 검사와 전력조정 및 수신기 존재확인을 수행할 수 있도록 slot을 구성해 두었다. 이 slot에서는 전력 전송이 잠시 중단되며, 각 기능은 송신기 및 수신기가 정의한 최대 시간을 초과해서는 안된다.

위 조건들을 만족시키며 안정적으로 동작하기 위해서는 FOD, Comm, Power Transfer 사이의 Scheduling이 필요하다. 본 논문에서는 Power Transfer 단계에서 Scheduling를 구성하기 위해 만족해야 할 조건과 표준에서 정의하는 방법을 알아보고, 나아가 보다 효율적이고 안정적인 Scheduling에 대한 연구를 수행하였다.

### II. 본론

본 논문에서는 Scheduling에 필요한 조건들을 정리하고 표준에서 정의하는 방법의 문제점을 제기하고 최종적으로는 보다 안정적이고 효율적인 알고리즘을 제안한다. 아래의 모든 기술은 FOD Lag Max가 Comm Lag Max보다 크다는 가정으로 기술한다. (반대의 경우 동일하게 적용됨)

#### 1. Scheduling 조건

KI 표준에 따르면 Scheduling은 FOD Lag Max, Comm Lag Max, min T Power 이 3가지 변수에 의해 결정된다. FOD Lag Max는 FOD slot과 다음 FOD slot 사이의 최대시간을 의미하며, 송신기에 그 값이 정의되어 있어야 한다. Comm Lag Max는 Comm slot과 다음 Comm slot 사이의 최대시간을 의미하며, 수신기에 그 값이 정의되어 Configuration 단계에서 NDEF 메시지 형태로 송신기에 전달해야 한다. min T Power는 전력전송이 이루어져야하는 최소시간을 의미하며 Comm Lag Max와 마찬가지로 수신기에 저장되어 있어 Configuration 단계에 송신기로 그 값이 전달되어야 한다.

Power Transfer 단계에서는 위 조건을 만족하는 1 사이클을 구성하고 이 사이클이 반복되도록 구성한다.

#### 2. Slotted Communication 예시

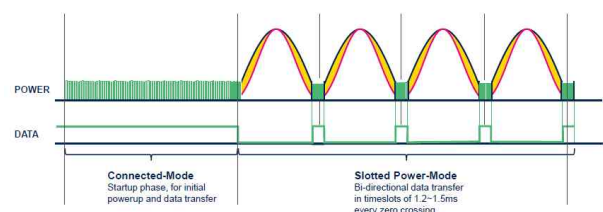


그림 1. Slotted Communication [1]

그림 1에서 보는 것처럼 전력전송 사이의 Slot에서 FOD와 통신이 이루어진다. 일반적으로 일정시간마다 Slot이 생성되는 구조로 설명을 한다.

하지만 Slot이 위와 같이 일정하게 구성된다면 Comm slot에서 FOD slot을 통과하여 Comm slot으로 이어지는 경우, Comm slot과 다음 Comm slot 사이의 시간이 Comm Lag Max를 초과하는 문제가 발생한다. 역으로 Comm Lag Max에 시간을 맞추기 위해 Slot의 주기를 줄이면 전송하는 평균 전력이 감소하는 문제가 발생한다.

### 3. Scheduling 알고리즘 제안

2번 항목에서 보듯, 사이클이 바뀌면서 발생하는 시간초과로 정해진 최대 시간이 초과해버리는 문제가 발생한다. 이 문제를 해결하기 위해 Slot의 주기를 줄이면 전체 전력이 감소하는 문제와 min T Power 값에 의한 제약이 커지는 두 가지 문제가 존재한다. 그러므로 slot을 최소화하면서 정해진 시간을 초과하지 않는 것이 중요하다. 이를 만족하기 위해, FOD slot의 Comm Lag Max의 절반시간만큼 전후로 Comm slot을 수행하고 남은 시간을 균등하게 Comm slot을 배치할 것을 제안하며, 해당 형태가 Comm Lag Max를 초과하지 않으며 가장 min T Power 변수의 제약을 적게 받는 형태임을 주장한다. 해당 주장은 아래의 두 가지 근거에 따른다.

- Comm slot - FOD slot - Comm slot으로 이어지는 경우, Comm Lag Max를 초과하지 않으려면 FOD slot의 전후 시간의 합이 Comm Lag Max보다 작거나 같아야 한다.
- 전후를 동일하게 나눈 상태에서 한 쪽 시간이 커질 경우, 다른 한 쪽은 적은 시간을 할당받으므로 min T Power 변수의 제약을 더 크게 받게 된다.

위의 방식으로 1사이클을 구성하는 것은 FOD Lag Max가 Comm Lag Max의 3배 이상인 경우에만 적용이 가능하다. 두 값의 비율이 1:1이라면 FOD slot들의 가운데 지점에 Comm slot 1개만 배치하는 것이 효율적이다. 비율이 2:1이라면 Comm Slot과 Comm Slot 사이의 간격이 min T Power 값보다 작아지는 경우가 있으므로 3등분하여 배치하는 것이 적절하다.

위의 기술을 토대로 알고리즘을 구성하면 그림 2와 같다.

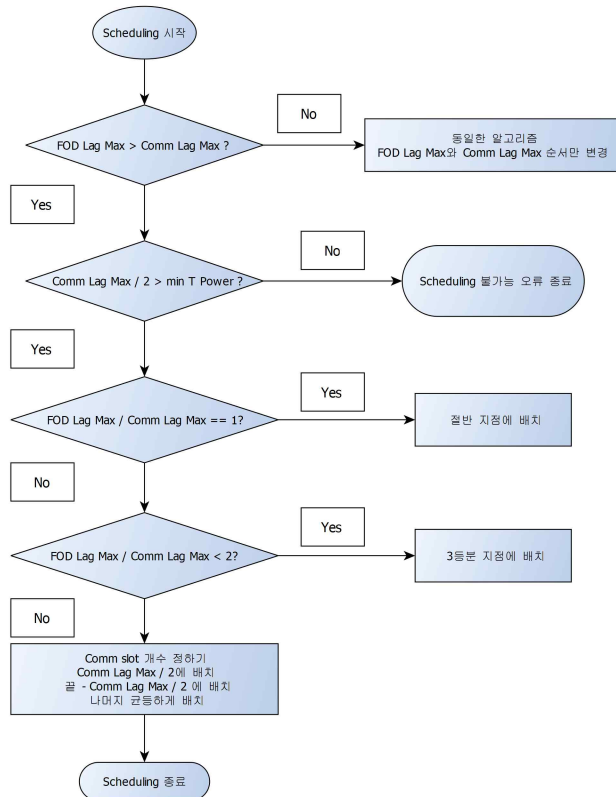


그림 2. Scheduling 알고리즘 제안

### III. 결론

본 논문에서는 WPC KI Protocol의 Power Transfer 과정의 Power Scheduling에 대한 알고리즘을 제안했다. 실제로 구현한 코드는 그림 3과 같고, FOD Lag Max, Comm Lag Max, minTPower를 각각 30/8/1,

15/4/1로 설정하여 실험한 결과는 그림 4에 나타내었다. FOD slot과 Comm slot을 구분하기 위해 FOD slot의 전력을 조금 더 높게 설정했다.

```
static void PowerScheduling(uint8_t maxFodlag, uint8_t maxCommLag, uint8_t minPower){
    uint8_t fodnum;
    uint8_t commnum;
    uint8_t interval;
    int i;

    if(maxFodlag > maxCommLag){
        if(minPower * 2 < maxCommLag){
            commnum = maxFodlag/maxCommLag;
            if(maxFodlag%maxCommLag){
                commnum++;
            }
            InitScheduleBuf(maxFodlag);
            scheduleBuffer[1] = POWER_SCHEDULE_FOD;
            if(commnum == 1){
                scheduleBuffer[maxFodlag/2 + 1] = POWER_SCHEDULE_COMM;
            }else if(commnum == 2){
                scheduleBuffer[maxFodlag/3 + 1] = POWER_SCHEDULE_COMM;
                scheduleBuffer[maxFodlag*2/3 + 1] = POWER_SCHEDULE_COMM;
            }else{
                scheduleBuffer[maxCommLag/2 + 1] = POWER_SCHEDULE_COMM;
                scheduleBuffer[maxFodlag - maxCommLag/2 + 1] = POWER_SCHEDULE_COMM;
                interval = (maxFodlag - maxCommLag) / (commnum-1);
                for(i=0; i<commnum-2; i++){
                    scheduleBuffer[maxCommLag/2 + 1 + interval*(i+1)] = POWER_SCHEDULE_COMM;
                }
            }
        }else{
            DEBUG_PRINTF("minPower is too long or max comm lag is too short\n");
        }
    }else{
        if(minPower * 2 < maxFodlag){
            fodnum = maxCommLag/maxFodlag;
            if(maxCommLag%maxFodlag){
                fodnum++;
            }
            InitScheduleBuf(maxCommLag);
            scheduleBuffer[1] = POWER_SCHEDULE_COMM;
            if(fodnum == 1){
                scheduleBuffer[maxCommLag/2 + 1] = POWER_SCHEDULE_FOD;
            }else if(fodnum == 2){
                scheduleBuffer[maxCommLag/3 + 1] = POWER_SCHEDULE_FOD;
                scheduleBuffer[maxCommLag*2/3 + 1] = POWER_SCHEDULE_FOD;
            }else{
                scheduleBuffer[maxFodlag/2 + 1] = POWER_SCHEDULE_FOD;
                scheduleBuffer[maxCommLag - maxFodlag/2 + 1] = POWER_SCHEDULE_FOD;
                interval = (maxCommLag - maxFodlag) / (fodnum-1);
                for(i=0; i<fodnum-2; i++){
                    scheduleBuffer[maxFodlag/2 + 1 + interval*(i+1)] = POWER_SCHEDULE_FOD;
                }
            }
        }else{
            DEBUG_PRINTF("minPower is too long or max fod lag is too short\n");
        }
    }
}
```

그림 3. 알고리즘 코드로 구현

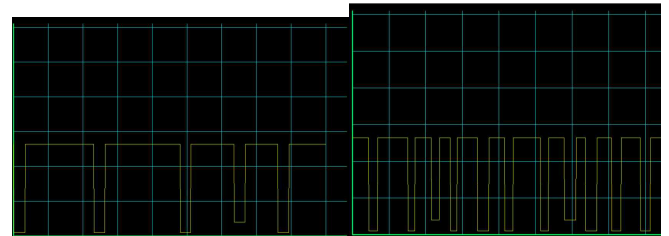


그림 4. 구현 결과

구현한 결과 FOD slot과는 무관하게 Comm slot 사이의 간격이 거의 일정한 것을 확인할 수 있었다.

### ACKNOWLEDGMENT

본 연구는 2021 과학기술정보통신부의 재원으로 정보통신 기획평가원(IITP) 지원을 받아 수행된 연구임 (No. 2021-0-00058, 무선 전력 전송 융합활성화센터 구축 및 운영)

### 참 고 문 헌

- [1] 양태영, "ST brings the cordless kitchen to life", ppt, "20220428\_e4ds\_ST\_webinar\_presentation.pdf".