

# 원격 점검 장치 데이터 연동 규격서

[ 원격 점검 장치 ]

Version	1.0.2a
Date	'22.04.15

한국전기안전공사  
원격 점검 추진 센터

## Revision History

[illegible]

## 목 차

1.	문서의 목적 .....	1
2.	용어의 설명 .....	1
3.	연동 규격 .....	4
4.	푸시 메시지 .....	10
4.1.	Last Will .....	11
4.2.	Connection 메시지 .....	11
4.3.	주기 메시지 .....	12
4.4.	이벤트 메시지 .....	12
5.	제어 메시지 .....	13
5.1.	설정 메시지 .....	16
5.2.	조회 메시지 .....	18
5.3.	FOTA 요청 .....	19
5.4.	파형정보 요청 .....	20
5.5.	재시작 .....	21
5.6.	파일전송 .....	22
6.	긴급제어 메시지 .....	23
6.1.	시스템 콘솔 .....	24
6.2.	AT Command .....	25

## 1. 문서의 목적

일반용전기설비에 설치하는 원격점검장치가 한국전기안전공사 관제시스템과 정보전달 규칙을 표준화 하기 위해 통신 메시지에 대한 연동 규격을 정의 함

- (1) 전기안전관리법 제12조의2(관제센터의 설치·운영) 산업통상자원부장관은 원격점검의 효율적 관리를 위하여 원격점검장치와 연결하여 그 측정결과를 전산처리할 수 있는 관제센터를 설치·운영할 수 있음
- (2) 같은 법 제43조(권한의 위임·위탁) 제2항 제3의2에 따라 제12조의2에 따른 관제센터의 설치·운영을 산업통상자원부장관은 한국전기안전공사에 위탁할 수 있음

## 2. 용어의 설명

- (1) **원격점검** : 전기설비의 과전압·과전류 및 누설전류 등을 검출하여 이를 데이터로 수집, 분석 및 전송함으로써 전기설비의 안전 상태 등을 점검하는 것을 말함
- (2) **원격점검장치** : 전기안전 원격점검기능을 갖추고 측정된 정보를 관제센터로 전송하는 장치를 말함(이하 “장치” 라 함)
- (3) **관제시스템** : 원격점검의 효율적 관리를 위하여 원격점검장치와 연결하여 그 측정결과를 전산처리할 수 있는 시스템을 말함(이하 “서버” 라 한다)
- (4) **관제센터** : 원격점검장치의 정보를 수집, 저장, 분석, 알림 등 효율적인 전산처리 및 관리를 위한 운영조직을 말함
- (5) **MQTT** : 사물인터넷(IoT)을 위한 OASIS 표준 메시지 송수신 프로토콜 임. TCP/IP 위에서 동작하지만 매우 가볍고, 양방향 통신과 브로드캐스팅, QoS(서비스 품질) 수준 지정, 영구세션, TLS 암호화를 통한 클라이언트 인증을 지원하는 등 주로 코드 공간이 작게 필요하거나 네트워크 대역폭의 제한되는 통신에 사용하는 프로토콜로 사물인터넷(IoT) 통신에 이상적임

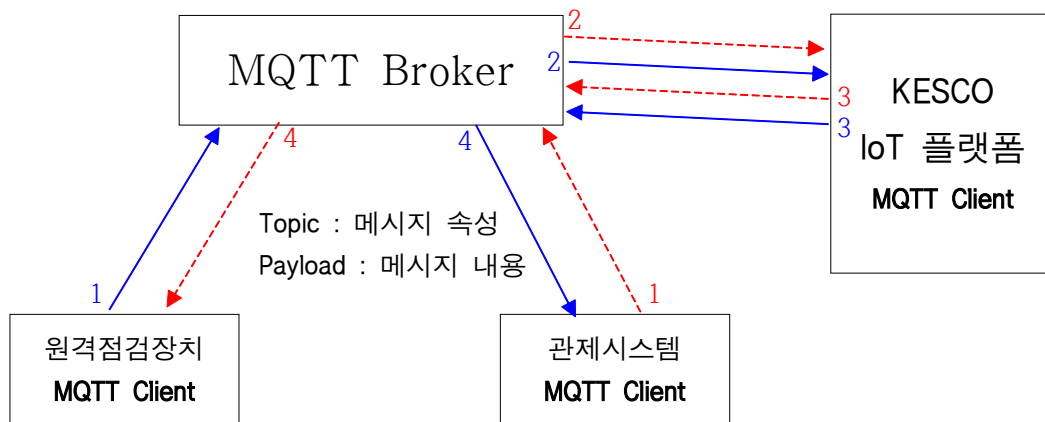
- (6) **OASIS** : OASIS(Organization for the Advancement of Structured Information Standards)는 보안, 사물인터넷, 에너지, 콘텐츠 기술, 긴급 관리 등의 기타 분야를 위한 표준 개발, 컨버전스, 채택을 담당하는 비영리 기관
- (7) **Topic** : MQTT 프로토콜에서 메시지를 발행(Publish) 시 메시지의 종류, 발행될 대상, 메시지 연동 규격 버전 등 메시지 속성을 정의 할 수 있는 메시지 제목에 해당되는 것
- (8) **Payload** : MQTT 프로토콜에서 메시지를 발행(Publish) 시 메시지 내용(Data)에 해당되는 것
- (9) **데이터 프로파일(Data Profile)** : 원격점검장치에 대한 정보를 구성하는 가상의 구조체로서 연동규격 버전 및 측정기능, 장치상태 등의 구조를 정의하고 관제센터에서 각각의 원격점검장치 설정 및 조회 등을 하기위한 경로 정보가 들어있는 구조체
- (10) **JSON** : 속성-값으로 이루어진 데이터의 객체를 전달하기 위한 개방형 데이터 표준 포맷방식으로 자료의 종류에 제한이 없으며 표현방식으로 중괄호와 따옴표를 이용해 데이터를 정리하는 방법 또는 중괄호로 묶여진 데이터 세트는 또 다시 쉼표로 구분하여 확장 할 수 있음
- (11) **인증** : 서로 통신하기 전에 물리적 객체가 접속하려는 아이디어에 해당되는 객체가 맞는지 확인하는 절차로 보안인증서를 사용함
- (12) **ADC** : 아날로그 신호를 디지털 값으로 바꾸는 장치 또는 기능을 말하며 ADC에 기준 전압과 입력신호가 입력되면 ADC 성능에 따라 출력 값이  $\text{입력신호} / \text{기준전압} \times \text{해상도}$  로 출력됨

(13) **MQTT Broker** : MQTT 프로토콜을 사용하기 위해 구축되어 있는 메시지 중계 프로그램으로서 모든 메시지는 MQTT Broker를 통해 목적지로 전달 됨

(14) **FOTA** : FOTA(Firmware Over The Air)원격으로 장치 프로그램을 무선통신을 통해 변경하는 것

### 3. 연동 규격

- (1) 본 문서에서 기술하는 연동규격 프로토콜로 MQTT를 사용함
- (2) 장치 또는 관제시스템의 보안 인증을 위해 TLS 1.3 버전의 인증서를 사용하며 인증서 파일을 송·수신서버로 부터 발급 받아야 함
- (3) SSL을 통해 MQTT Broker에 접속하고 송수신 데이터를 암호·복호화 할 수 있어야 하며 주기적으로 전기안전공사에게 발급 받은 인증서를 원격으로 교체 가능해야 함
- (4) MQTT 프로토콜은 [그림1]과 같은 ACP(Access Control Policy)에 의해 MQTT Broker에서 메시지를 주고 받음



[그림1] 원격점검장치 송·수신 프로그램 MQTT 프로토콜

- (5) 메시지는 Topic과 Payload로 이루어지며, Topic은 메시지 속성을 나타내는 역할을 하고 MQTT Broker에 설정된 ACP와 접속 계정에 따라 메시지 구독(Subscribe) 권한이 설정 됨. 이 문서에서 Topic은 메시지 속성을 구분하는 역할을 하며 기능별 Topic과 Payload 규칙이 정의 되어 있음
- (6) 장치와 관제시스템이 통신하기 위해 [표 1]과 같이 장치 그룹 아이디, 장치 아이디, 접속 토큰 값을 관제시스템에 등록해야 함

[표 1] 장치 등록 정보

이름	설명	비고
Serviceld	장치 그룹 아이디	
deviceld (=IMEI)	장치 아이디	=mqtt user ID, client ID
devicePw	접속 토큰	=mqtt user Password

(7) 장치에서 서버로 이동하는 메시지는 UpLink 메시지로, 반대로 서버에서 장치로 이동하는 메시지는 DownLink 메시지로 지칭함

UpLink 메시지와 DownLink 메시지는 다음과 같은 공통된 부분을 가지고 있음

```
{version}      = (연동 규격 버전)
{UpLink}       = mrm/{serviceld}/{deviceld}/up/{version}
{DownLink}     = mrm/{serviceld}/{deviceld}/down/{version}
```

(8) 관제시스템에 장치가 하나 생성되면 다음과 같이 장치에 대한 정보를 담은 가상의 구조체가 생성됨. 이것을 데이터 프로파일(Data Profile)라고 하며 다음과 같은 JSON 형식을 갖음

```
(연동 규격 버전) = {                                // 본 문서의 버전을 따름
  "val":{                                           // 측정값 저장
    "data":(현재 값),
    "hd":(고조파),
    "af":(알람코드)
  },
  "config":{                                       // 설정값 저장
    "event":[(이벤트 메시지 설정 값)],
    "period":(메시지 주기),
    "server":(서버정보)
  },
  "stat":(장치 상태 정보),                         // 현재 장치 상태 정보를 나타냄
  "wave":{                                         // 옵션 기능 1 : 파형정보 조회
    "trigger":(ADC 트리거 설정),
    "result":(ADC 데이터)
  },
  "watt":(전력량 데이터)                          // 옵션 기능 2 : 전력량 데이터 조회
}
// 표시 : 주석
```

※ 만약 장치에 새로운 기능이 추가된다면 심표와 종괄호를 통해 추가 할 수 있고 추가되는 기능 중 서버에서 수행해야 하는 것이 많고 복잡하다면 데이터 프로파일에서 연동 규격 버전이 변경될 수 있음



(가) 연동 규격 버전은 문서 번호와 같으며, 다음과 같은 규칙으로 표기

[A].[B].[C] <---- 점으로 구분

- [A] : 프로파일 식별자
- [B] : 버전(Major), 같은 기능이지만 한국전기안전공사 송수신 서버 프로그램 추가/변경이 필요할 경우
- [C] : 버전(Minor)

(나) 데이터 프로파일의 상세 내용은 다음과 같음

(현재 값) = {

```
"v" : {
    "c":(전압 rms 현재 값),
    "m":(전압 rms 최대 값),
    "n":(전압 rms 최소 값),
    "a":(전압 rms 평균 값),
},
"c" : {
    "c":(부하전류 rms 현재 값),
    "m":(부하전류 rms 최대 값),
    "n":(부하전류 rms 최소 값),
    "a":(부하전류 rms 평균 값),
    "x":(부하전류 rms 벡터X 평균 값),
    "y":(부하전류 rms 벡터Y 평균 값)
},
"z" : {
    "c":(영상전류 rms 현재 값),
    "m":(영상전류 rms 최대 값),
    "n":(영상전류 rms 최소 값),
    "a":(영상전류 rms 평균 값),
    "x":(영상전류 rms 벡터X 평균 값), //igr
    "y":(영상전류 rms 벡터Y 평균 값) //igc
},
"g" : {                                     //추가 측정요소 선택사항
    ...
},
"t" : {
    "c":(기판 온도 현재 값),
    "m":(기판 온도 최대 값),               //선택사항
    "n":(기판 온도 최소 값),               //선택사항
    "a":(기판 온도 평균 값),
}
}
```

- \* "v" 는 전압, "c" 는 부하전류, "z" 는 누설전류, "g" 추가적인 측정 값
- \* 평균 값은 메시지 전송과 전송 사이 평균
- \* 최대 값은 메시지 전송과 전송 사이 최대 값
- \* 최소 값은 메시지 전송과 전송 사이 최소 값
- \* 상기 '메시지' 에는 (현재 값)이 포함되어야 함

● 추가적인 측정 값 예) 병렬아크 감지

“arc” : { “ret”: 0, “stat”: (text) }

- 추가시 안전공사와 협의 필요

```
(고조파) = {
  “v”:{
    “n”:[(rms),(1차 고조파),(2차 고조파),(3차 고조파),(4차 고조파),(5차 고조파),...]
    “x”:[(rms),(1차 고조파),(2차 고조파),(3차 고조파),(4차 고조파),(5차 고조파),...]
    “y”:[(rms),(1차 고조파),(2차 고조파),(3차 고조파),(4차 고조파),(5차 고조파),...]
  },
  “c”:{
    “n”:[(rms),(1차 고조파),(2차 고조파),(3차 고조파),(4차 고조파),(5차 고조파),...]
    “x”:[(rms),(1차 고조파),(2차 고조파),(3차 고조파),(4차 고조파),(5차 고조파),...]
    “y”:[(rms),(1차 고조파),(2차 고조파),(3차 고조파),(4차 고조파),(5차 고조파),...]
  },
  “z”:{
    “n”:[(rms),(1차 고조파),(2차 고조파),(3차 고조파),(4차 고조파),(5차 고조파),...]
    “x”:[(rms),(1차 고조파),(2차 고조파),(3차 고조파),(4차 고조파),(5차 고조파),...]
    “y”:[(rms),(1차 고조파),(2차 고조파),(3차 고조파),(4차 고조파),(5차 고조파),...]
  },
  “g”:{ //선택사항
    “n”:[(rms),(1차 고조파),(2차 고조파),(3차 고조파),(4차 고조파),(5차 고조파),...]
    “x”:[(rms),(1차 고조파),(2차 고조파),(3차 고조파),(4차 고조파),(5차 고조파),...]
    “y”:[(rms),(1차 고조파),(2차 고조파),(3차 고조파),(4차 고조파),(5차 고조파),...]
  }
}
```

- \* “v” 는 전압, “c” 는 부하전류, “z” 는 누설전류, “g” 추가적인 측정 값
- \* 각 요소들은 배열로 나타내며 배열의 첫 번째는 모든 고조파 성분의 합
- \* “n” 은 “x”, “y” 의 벡터 합을 나타냄

(알람코드)

다음과 같이 계산해서 Hex String 값으로 입력 함

만약, 이벤트 메시지 설정을 8개 이상 설정가능 할 경우엔 Hex String이 뒤로 확장되어 길이가 늘어날 수 있음

이벤트 번호	0	1	2	3	4	5	6	7	bit	HEX
0,1,2,3,4,5 이벤트발생	1	1	1	1	1	1	0	0	1111 1100	FC
정상시	0	0	0	0	0	0	0	0	0000 0000	00

확장시 이벤트 번호	0	1	2	3	4	5	6	7	...	bit	HEX
0,1,2,3,4,5 이벤트발생	1	1	1	1	1	1	0	0	0	1111 1100 0000 0000	FC00
정상시	0	0	0	0	0	0	0	0	0	0000 0000 0000 0000	0000

```
(이벤트 메시지 설정 값) = [
{
  "n":(항목이름=v_a),
  "h":(임계 값 상한),
  "l":(임계 값 하한),
  "t":(지속시간)
},
... 설정 항목별로 나열 됨
]
```

\* 항목이름은 어떤 값을 참조 할 것인지 나타내며 JSON 구조에서 “val” 이하 경로를 나타냄

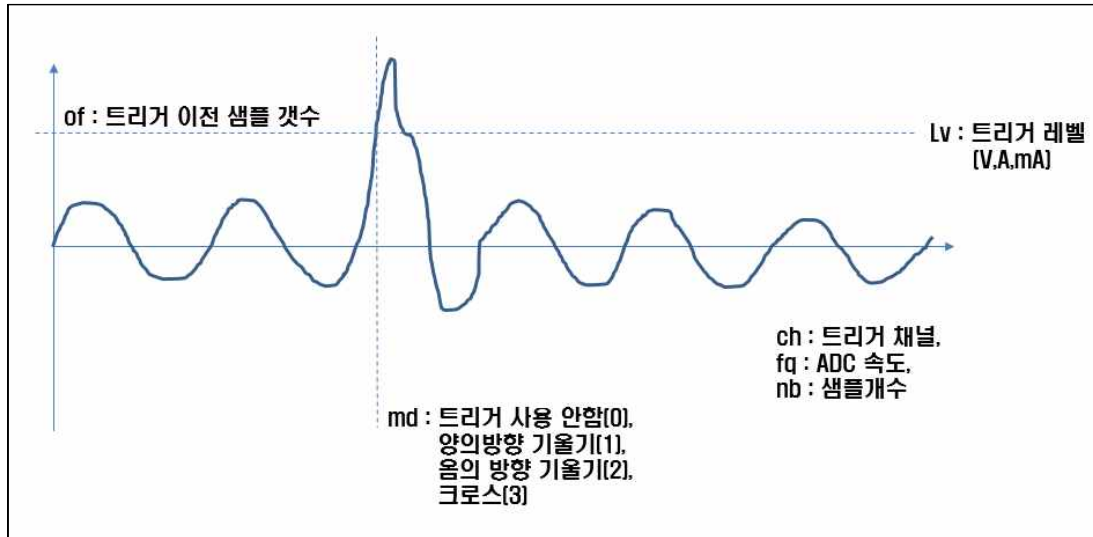
예) 이벤트 판단 조건으로 (현재 값) 항목들 중에서 전압의 rms 평균을 가지고 판단하려 할 경우 “data” 에서 “v”의 “a”이므로 “data/v/a”라고 설정 함

\* 하한이 존재하면 “l”에 값을 입력하고, 하한이 없으면 키 값 “l” 자체를 생략

```
(서버정보) = {
  "host":(서버 주소),
  "port":(서버 포트),
  "sid":(장치 그룹 이름), // ServiceId
  "uid":(장치별 접속 ID), // mqtt user Id, deviceId, IMEI
  "upw":(장치별 접속 PW), // mqtt user Password, devicePw
  "cid":(장치 IMEI), // mqtt client Id, deviceId, IMEI
  "cert":(인증서 cert 파일 경로),
  "key":(인증서 키 파일 경로),
  "ca":(인증서 CA 파일 경로)
}
```

```
(장치 상태 정보) = {
  "gen":{
    "fw_ver" : [(펌웨어 버전)], // 관리해야 할 프로그램이 여러가지면 배열로 나열 함
    "dev_ver" : (하드웨어 버전),
    "maker" : (제조사),
    "bat" : (배터리 충전 상태), // 사용 시 (%) 로 기입 / 미사용 시 생략
  },
  "net":{
    "comp" : (넷사업자_망이름), // 예) KT_EMTC, SKT_CATM1, LG_LTE 등
    "ant" : (안테나 레벨), // 안테나 레벨 ( 0 ~ 5 )
    "band" : (사용 밴드), // 예) B3, B5
  }
}
```

```
(ADC 트리거 설정) = {
  "w":["v", "c", "z", "g"], // 반환 받을 데이터
  "md":(트리거모드; 0/1/2/3), // 0: "disable", 1: "Accend", 2: "Decend", 3: "Both"
  "ch":(트리거채널; 0/1/2/3/...), // 0: "v" , 1: "c" , 2: "z" , 3: "g"
  "lv":(트리거레벨), // % 단위로 설정, 100%면 ADC의 Vref 임
  "of":(트리거오프셋), // nb보다 작아야 함
  "pd":(60Hz 주기수 0/1/2/3/...), // 0: 0.5주기, 1: 1주기, 2: 2주기, 3:....
  "nb":(ADC 샘플 개수) // 통신 메시지로 보낼 총 데이터 샘플 수 ( ~ 200)
}
```



```
(ADC 데이터) = {
    "v": [...],
    "c": [...],
    "z": [...],
    "g": [...]
}
```

\* ADC 데이터는 ADC 트리거 설정이 포함된 파형 요청에 따라 응답 메시지로써 전달 됨.

```
(전력량 데이터) = { // 선택사항
    "kwh" : (적산 소비전력량),
    "kvar" : (적산 무효전력량), // 생략가능
    "kva" : (적산 피상전력량) // 생략 가능
}
```

(9) 메시지 종류는 크게 3가지로 구분함

- (가) 단순히 장치에서 서버로 보내는 메시지인 “푸시메시지”
- (나) 서버에서 장치로 메시지를 보내고 응답을 받는 “제어메시지”
- (다) 제어메시지로 제어 불능상태에서 장치를 복구하기 위한 “긴급메시지”

## 4. 푸시 메시지

푸시메시지에는 Last Will, Connction, 주기적메시지, 이벤트메시지가 있고 메시지 구조는 표2와 같음

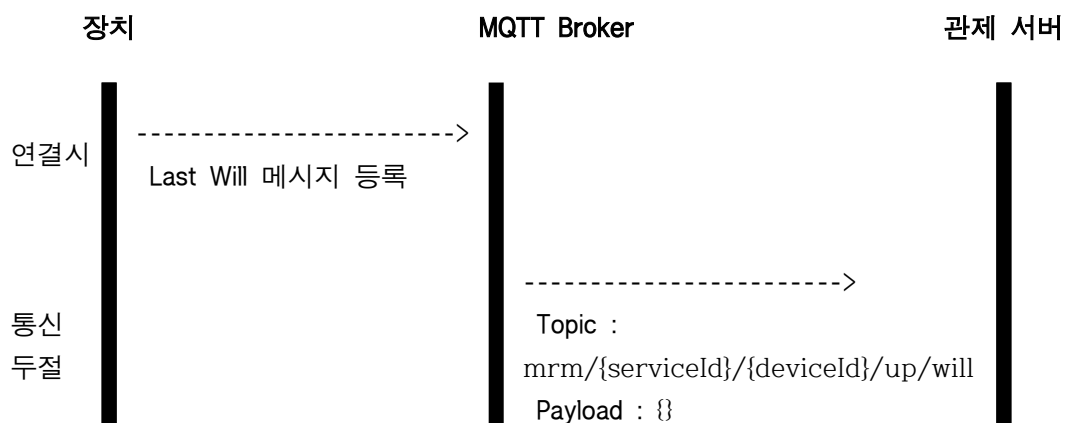
[표2] 푸시 메시지

이름	설명	메시지 구조
Last Will	예상치 못한 통신두절을 알려주는 메시지	Topic : {UpLink}/will Payload : {}
Connection	연결상태를 알려주는 메시지	Topic : {UpLink}/con [연결시] Payload : Connect [종료시] Payload : Disconnect
주기적 메시지	설정된 주기에 따라 주기적으로 상태를 알려주는 보고 메시지	Topic : {UpLink}/p Payload : (현재 값)
이벤트 메시지	이벤트 조건 설정에 따라 이벤트 발생시 발생하는 메시지	Topic : {UpLink}/e Payload : { "data" :(현재 값), "af" :(알람 코드)}

\* {} 표시에 어떤 내용이 들어와도 상관없음

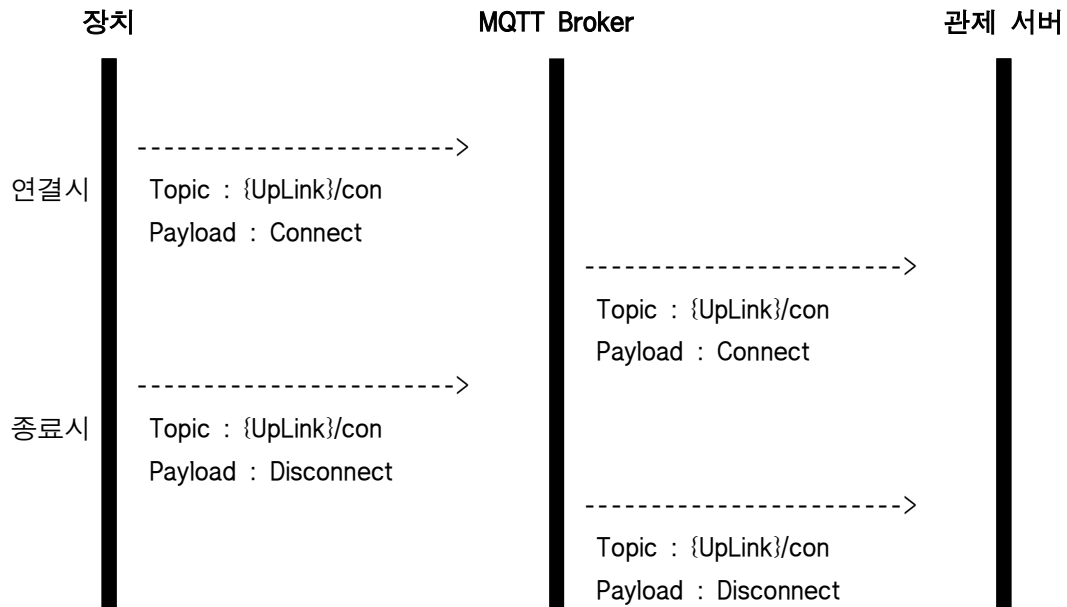
### 4.1. Last Will

장치가 예상치 못하게 통신두절 되는 경우 Last Will 이라는 메시지를 MQTT Broker에 접속할 때 미리 등록할 수 있으며, 예상치 못하게 통신두절 될 경우 아래와 같이 메시지를 송신함



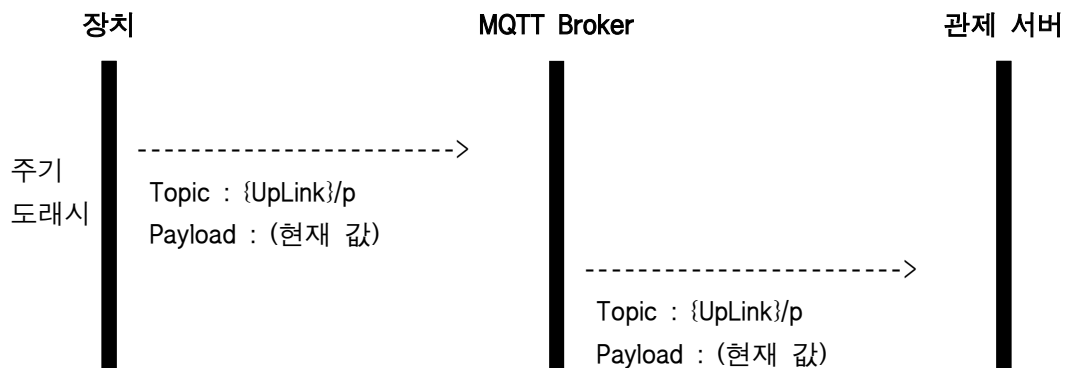
## 4.2. Connection 메시지

장치가 MQTT Broker에 연결될 때, 연결을 종료하기 전 MQTT Broker에서 상태를 알기 위해 메시지를 보냄



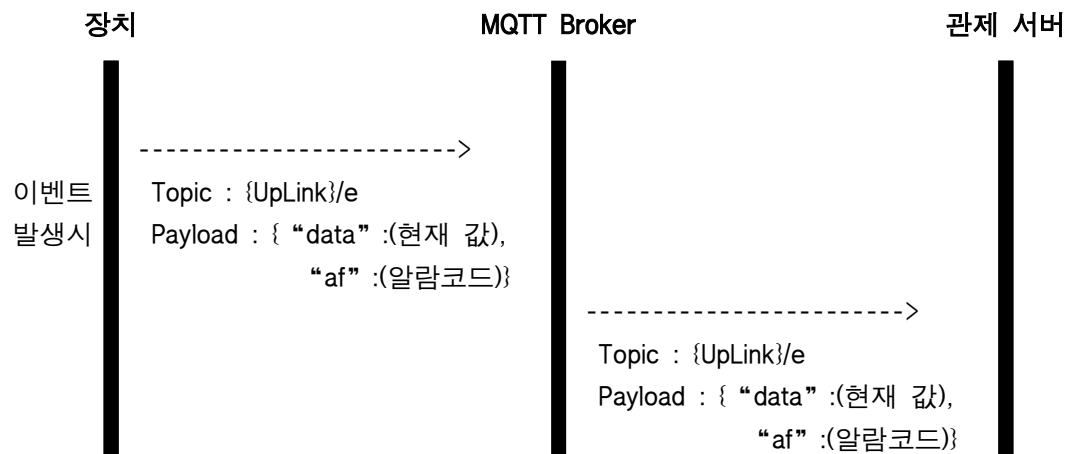
## 4.3. 주기 메시지

관제시스템에서 각 장치별로 상태정보 추이를 확인하기 위해 주기 메시지를 받음. 여기서 메시지 발생 주기는 장치별 설정 값을 참조



#### 4.4. 이벤트 메시지

이벤트 발생시 즉시 관제시스템에서 알고 조치할 수 있도록 각 장치별로 설치환경에 맞게 설정된 이벤트 메시지 발생조건에 따라 이벤트 메시지를 서버로 발송함



## 5. 제어 메시지

관제시스템에서 장치의 상태를 조회하거나 상세한 분석을 위해 또는 알람 설정 변경/추가를 위해 제어 메시지를 송신하고 이에 따른 결과를 응답메시지를 통해 받음

본 연동 규격 버전에서 설명하는 제어 메시지는 다음 표3과 같음

[표3] 제어 메시지

이름	설명	메시지 구조
설정 메시지	장치 설정을 변경하는 메시지	<p>[요청]</p> <p>Topic : {UpLink}/set/{JSON 경로}</p> <p>Payload : (변경 하려는 값)</p> <p>[응답]</p> <p>Topic : {UpLink}/set/{JSON 경로}</p> <p>Payload : (변경 된 값)</p>
조회 메시지	장치 설정을 조회하는 메시지	<p>[요청]</p> <p>Topic : {UpLink}/get/{JSON 경로}</p> <p>Payload : {}</p> <p>[응답]</p> <p>Topic : {UpLink}/get/{JSON 경로}</p> <p>Payload : (조회 된 값)</p>
FOTA 요청	원격으로 내부 프로그램을 변경하는 메시지	<p>[요청]</p> <p>Topic : {DownLink}/fota</p> <p>Payload : {</p> <p>    "url": (firmware url),</p> <p>    "to": (save to),</p> <p>    "ver" : (version),</p> <p>    "size" : (firmware file size),</p> <p>    "psk" : (Pre-Shared key),</p> <p>    "crc" : (CRC)</p> <p>}</p> <p>[응답]</p> <p>Topic : {UpLink}/fota</p> <p>Payload : {</p> <p>    "result" : (결과; success/fail),</p> <p>    "crc" : (CRC)</p> <p>}</p>

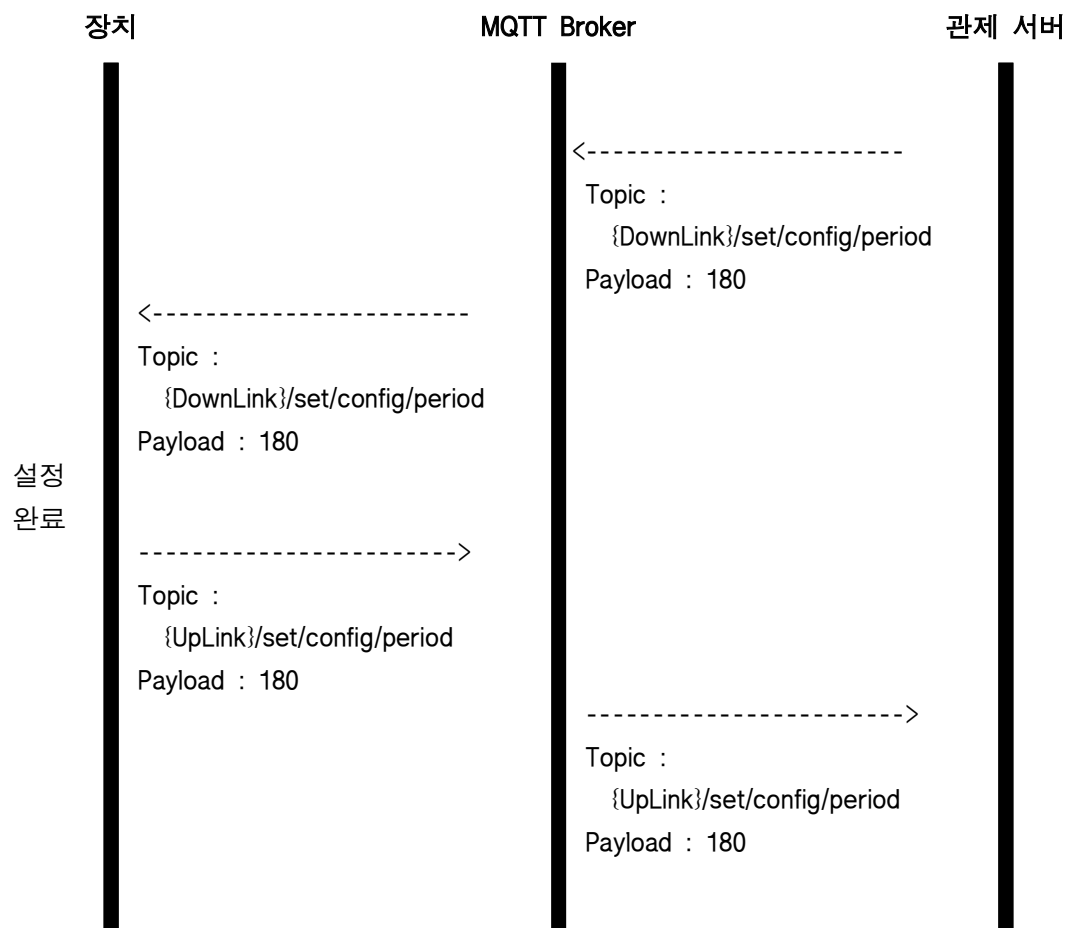


파형정보 요청	주어진 트리거 조건에 따라 파형 정보를 요청 하는 메시지	<p>[요청]</p> <p>Topic : {DownLink}/wave</p> <p>Payload : (ADC 트리거 설정)</p> <p><i>* ch 0: 전압,</i>  <i>1: 부하전류,</i>  <i>2: 영상전류,</i>  <i>3: 추가 측정 요소 // 선택사항</i></p> <p><i>* md 0: 상승,</i>  <i>1: 하강,</i>  <i>2: 상승/하강</i></p> <p><i>* fq 0~... // 숫자가 클수 록 ADC속도가 느려짐</i></p> <p>[응답]</p> <p>Topic : {UpLink}/get/wave</p> <p>Payload : {  “v”: [...], // 요청에 맞게 보낼 데이터  선정해서 보냄  “c”: [...],  “z”: [...],  “g”: [...]  }</p>
재시작	원격으로 장치를 재부팅 시킴	<p>[요청]</p> <p>Topic : {DownLink}/reboot</p> <p>Payload : {reboot 방법}</p> <p><i>*reboot 방법</i>  “sw” : 소프트웨어 리셋  “hw” : 하드웨어 리셋</p> <p>[응답]</p> <p>Topic : {UpLink}/con</p> <p>Payload : Disconnected</p> <p>Topic : {UpLink}/con</p> <p>Payload : Connected</p>

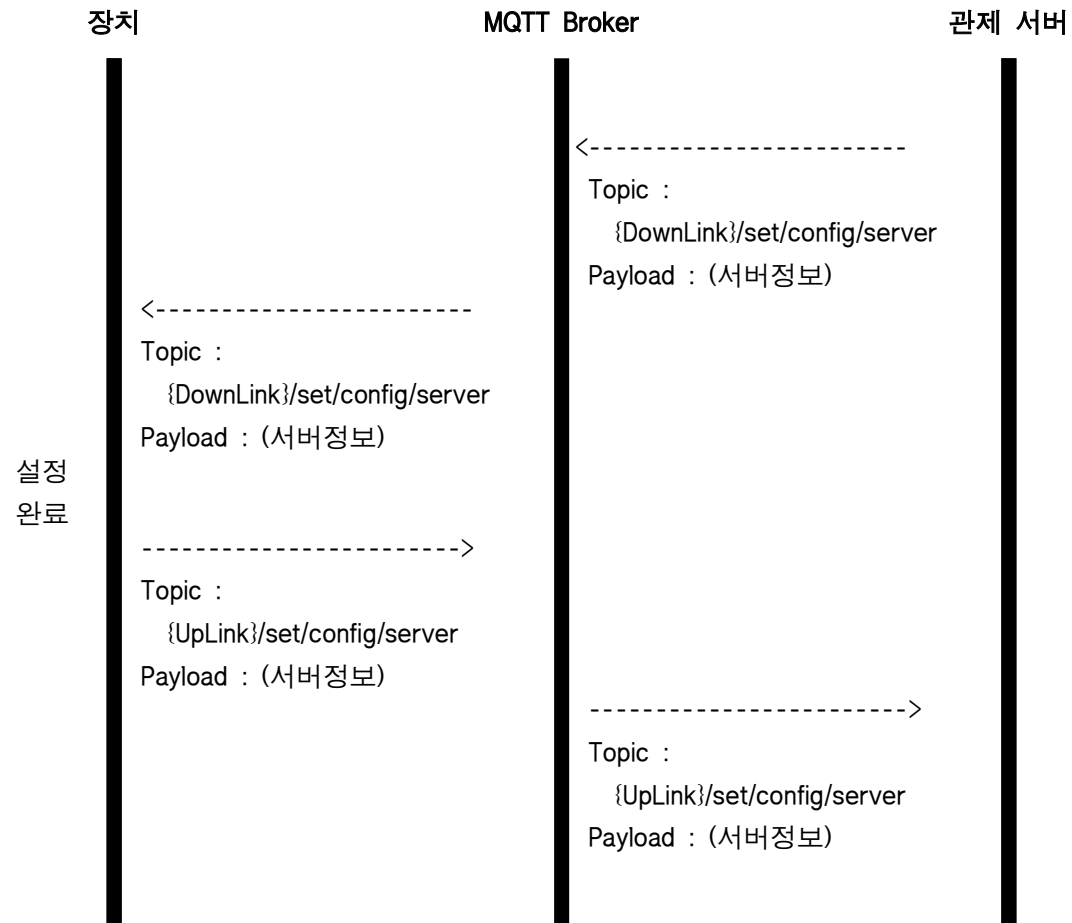
파일 전송	장치에 파일을 전송함 프로그램, 인증서 등	<p>[요청] Topic : {DownLink}/filedown Payload : {   "sz":{"파일전체크기},   "nb":{"사이즈 1000으로 나눈 개수},   "crc":{"합친 파일 CRC},   "to":{"저장경로} }</p> <p>[응답] Topic : {UpLink}/filedown/0 Payload : {}</p> <p>[요청] Topic : {DownLink}/filedown/0 Payload : {Binay datas [최대길이 1000]}</p> <p>...</p> <p>[응답] Topic : {UpLink}/filedown/20 Payload : {}</p> <p>[요청] Topic : {DownLink}/filedown/20 Payload : {Binay datas [최대길이 1000]}</p> <p>다 받았으면,</p> <p>[응답] Topic : {UpLink}/filedown Payload : {   "result" : 0 }</p> <p><i>* result 0: 성공,           1: 실패</i></p>
-------	----------------------------	---

## 5.1. 설정 메시지

장치를 관제시스템에 등록할 때 데이터 프로파일이 생성되는데 바꾸고 싶은 값이 있다면 데이터 프로파일을 참조하여 바꾸려는 값의 JSON 경로를 Topic에 표현. 예를 들어 주기메시지 주기를 180초로 바꾸고 싶다면 아래와 같이 수행함



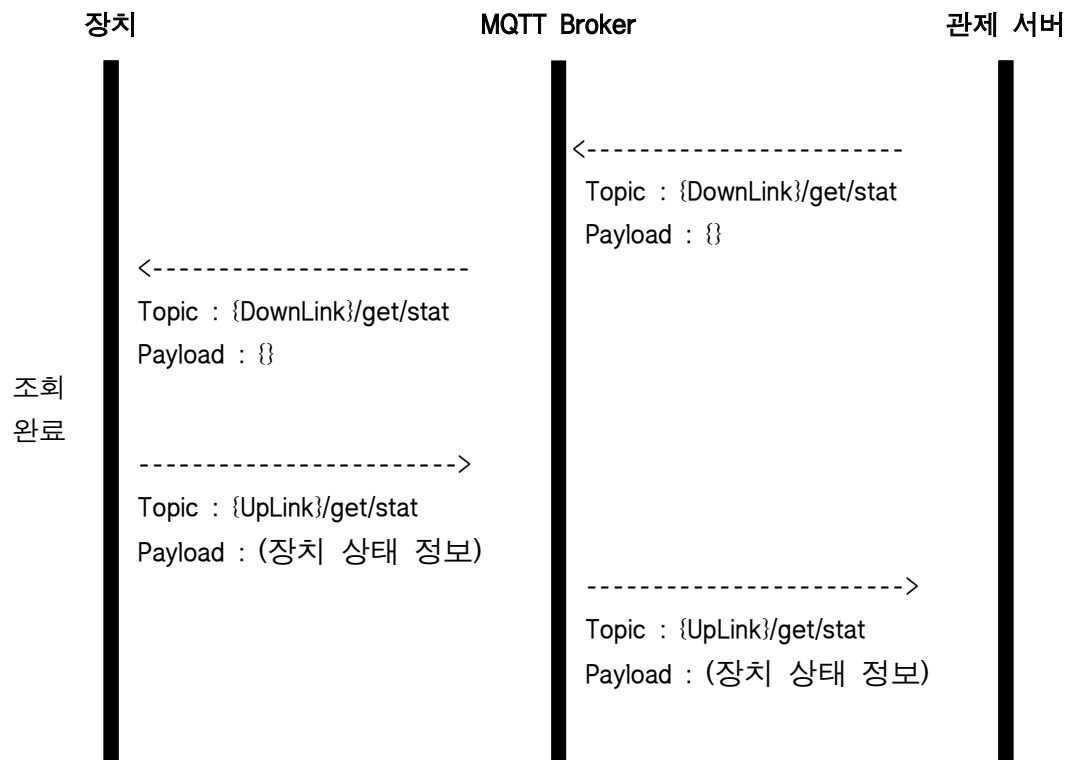
서버 정보와 같이 값들을 한꺼번에 변경하고 싶다면 아래와 같이 Payload  
에 JSON 형식으로 입력함



UpLink로 전송하는 (서버정보)는 설정 완료 후 장치에 적용된 값을 나타냄

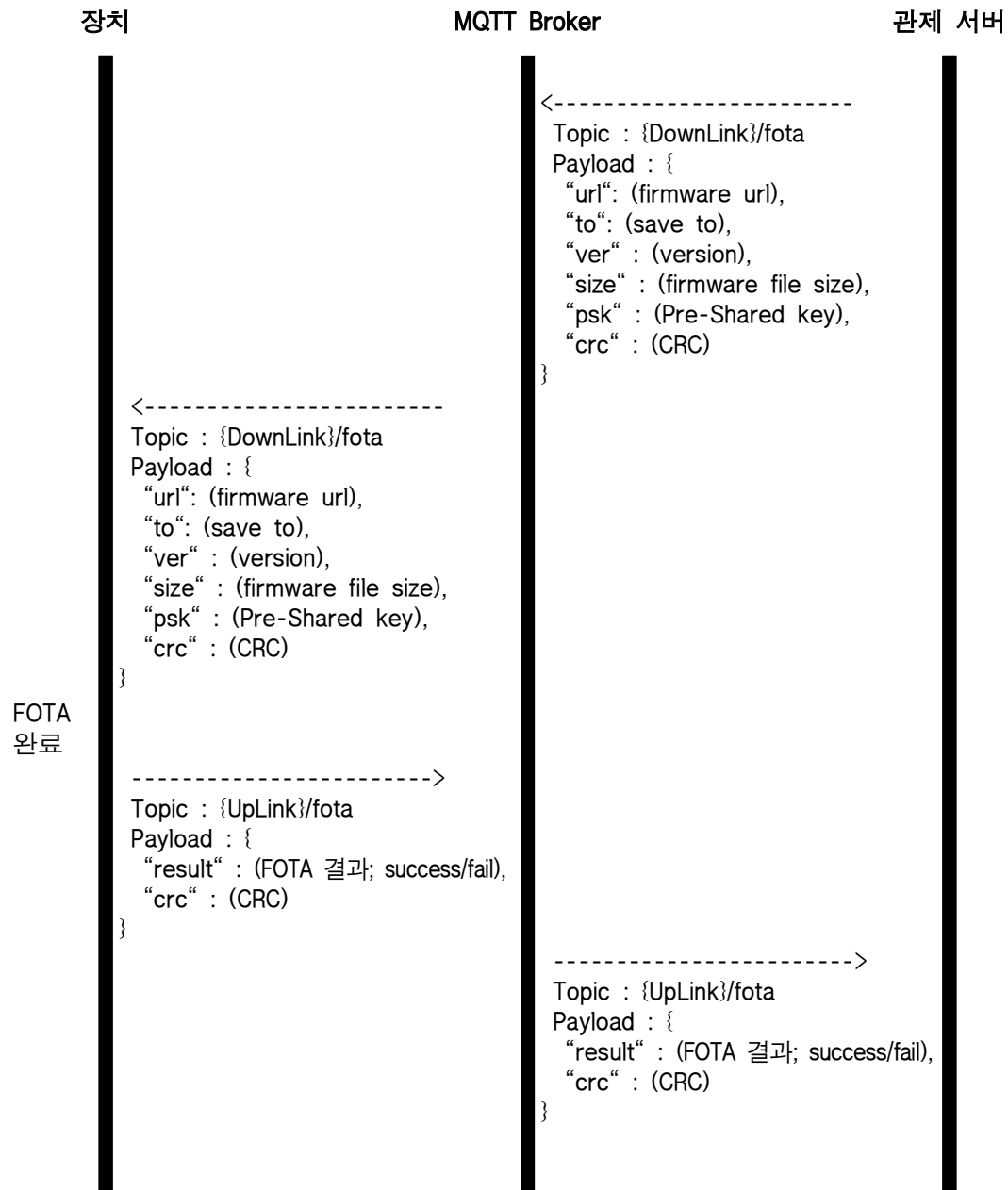
## 5.2. 조회 메시지

관제시스템에서 장치의 상태를 조회하기 위해서 데이터 프로파일에서 원하는 항목의 경로를 Topic에 입력하여 응답 메시지를 통해 상태 값을 받음. 예를 들어서 장치 상태정보를 알고 싶다면 다음과 같이 데이터를 송수신함



### 5.3. FOTA 요청

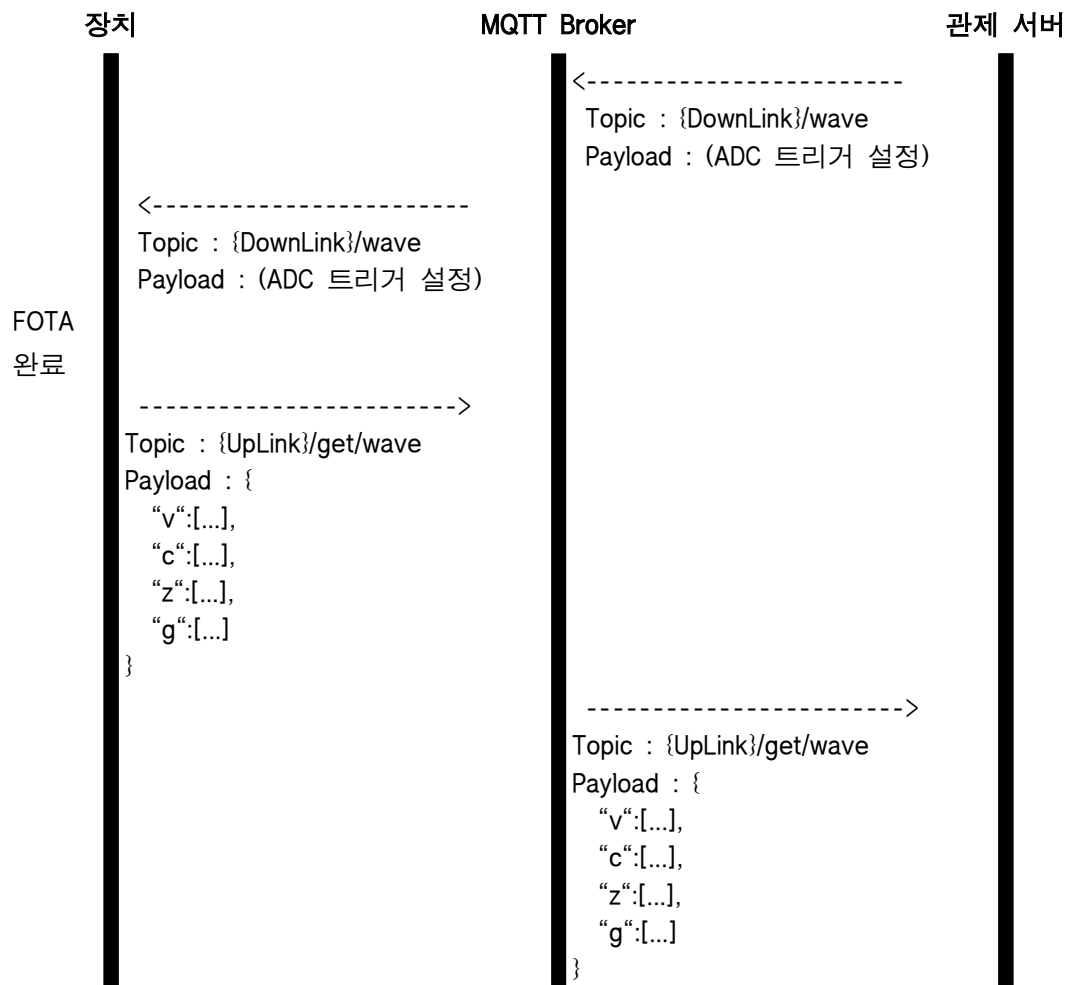
원격으로 현장에 설치된 장치의 내부 프로그램을 변경하고 싶을 때 서버에서 다음과 같은 제어 메시지를 통해 프로그램, 인증서 등 파일을 전송함



전송 받은 파일은 한국전기안전공사에서 알려 줄 암호 알고리즘과 압축 알고리즘, PSK를 통해 블록 암호화 및 압축되어있음  
따라서 원격점검 장치에서 압축해제 후 복호화 해서 (save to) 경로에 저장하고 CRC 검증 결과를 응답 메시지로 서버에 전송해야 함  
{{설정된 주기메시지 주기} + 60} 초 안에 서버로 FOTA 응답 메시지가 오지 않을 경우 Fail로 간주함

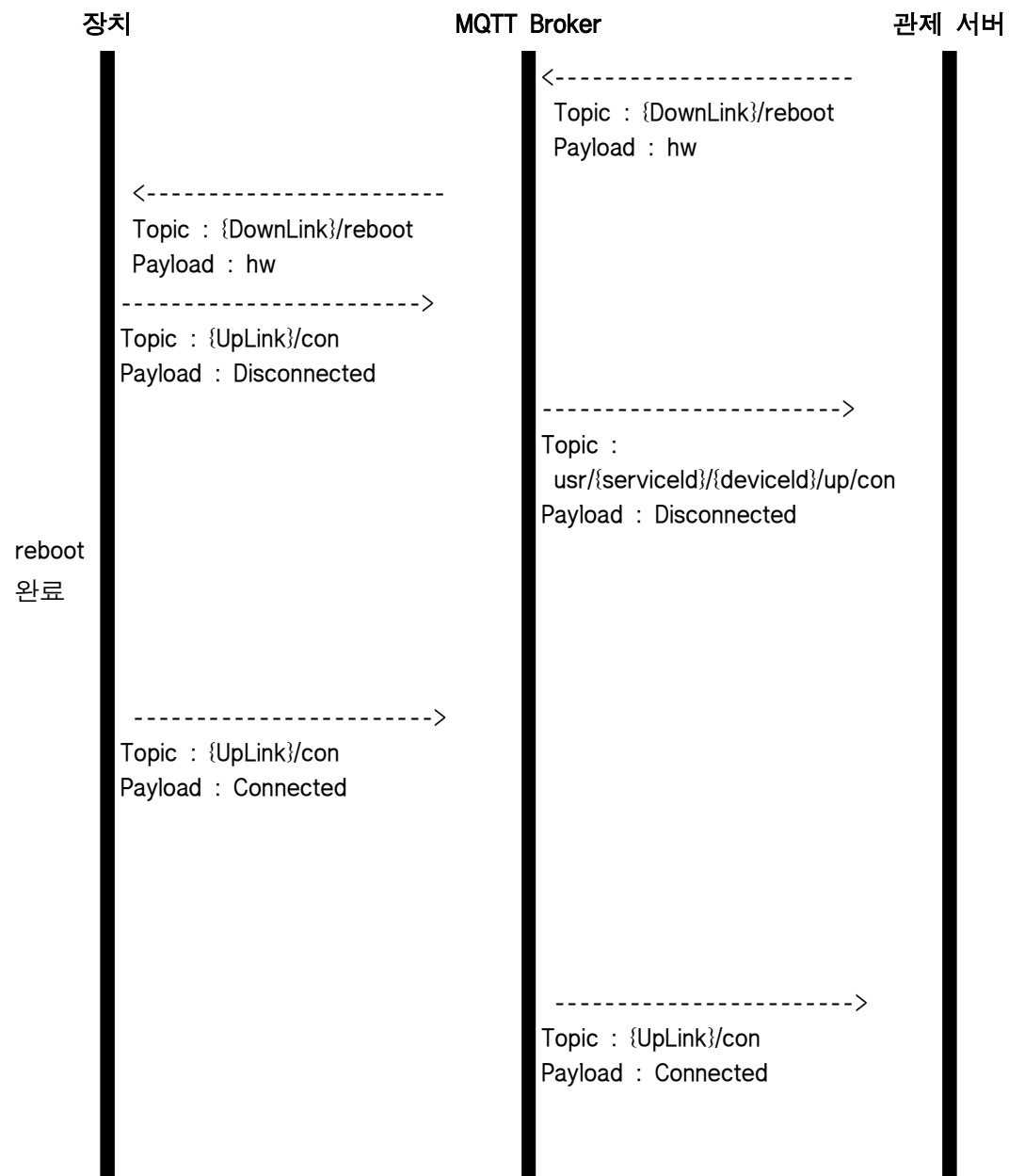
## 5.4. 파형정보 요청

관제시스템에서 사용자가 특정설비의 상태를 분석하기 위해 파형정보를 요청할 수 있음. 이때 트리거 조건을 설정하여 특정 채널의 파형만을 얻을 수도 있고, ADC 속도를 빠르게 조정하여 더 상세한 파형을 취득하거나 느리게 조정하여 전체적인 파형 추이를 볼 수 있음



## 5.5. 재시작

관제시스템에서 원격으로 장치를 재시작 할 수 있어야 함  
요청 메시지에 따라 하드웨어 리셋 또는 소프트웨어 리셋이 수행 됨





## 5.6. 파일 전송

관제시스템에서 장치로 인증서, 프로그램파일을 전송할 수 있어야 함

장치	MQTT Broker	관제 서버
<pre> &lt;----- Topic : {DownLink}/filedown Payload : {   "sz":{"파일전체크기},   "nb":{"사이즈 1000으로 나눈 개수},   "crc":{"합친 파일 CRC},   "to":{"저장경로} } -----&gt; Topic : {UpLink}/filedown/0 Payload : {}  &lt;----- Topic : {DownLink}/filedown/0 Payload : {Binay datas [최대길이 1000]} ... -----&gt; Topic : {UpLink}/filedown/20 Payload : {}  &lt;----- Topic : {DownLink}/filedown/20 Payload : {Binay datas [최대길이 1000]} -----&gt; Topic : {UpLink}/filedown Payload : {   "result" : 0 } </pre>	<pre> &lt;----- Topic : {DownLink}/filedown Payload : {   "sz":{"파일전체크기},   "nb":{"사이즈 1000으로 나눈 개수},   "crc":{"합친 파일 CRC},   "to":{"저장경로} } -----&gt; Topic : {UpLink}/filedown/0 Payload : {}  &lt;----- Topic : {DownLink}/filedown/0 Payload : {Binay datas [최대길이 1000]} ... -----&gt; Topic : {UpLink}/filedown/20 Payload : {}  &lt;----- Topic : {DownLink}/filedown/20 Payload : {Binay datas [최대길이 1000]} -----&gt; Topic : {UpLink}/filedown Payload : {   "result" : 0 } </pre>	
순차적 전송		
파일 전송 완료		

## 6. 긴급제어 메시지

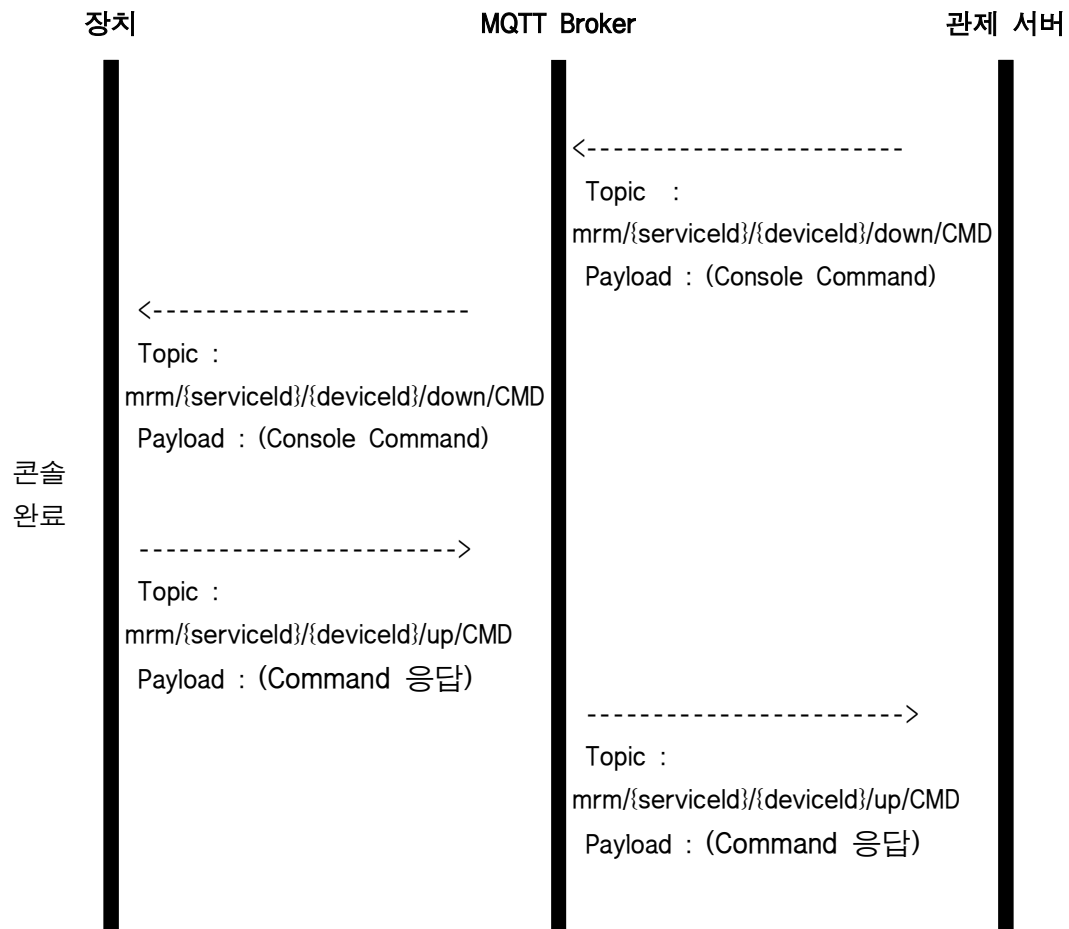
장치에서 메시지가 오지 않거나 제어메시지를 보내도 응답이 없는 상황일 경우 긴급제어 메시지를 통해 상태를 복구 할 수 있음. 긴급제어 메시지에 는 시스템 콘솔과 AT Command가 있음

[표4] 긴급 메시지

이름	설명	메시지 구조
시스템 콘솔	내부 시스템 콘솔	<p>[요청] Topic : mrm/{serviceld}/{deviceld}/down/CMD Payload : (Console Command)</p> <p>[응답] Topic : mrm/{serviceld}/{deviceld}/up/CMD Payload : (Command 응답)</p>
AT Command	통신모듈 직접 제어명령	<p>[요청] Topic : mrm/{serviceld}/{deviceld}/down/AT Payload : (AT Command)</p> <p>[응답] Topic : mrm/{serviceld}/{deviceld}/up/AT Payload : (AT Command 응답)</p>

## 6.1. 시스템 콘솔

장치 내부 OS에 접근하여 기초적인 명령을 내릴 수 있는 시스템 콘솔에 접근하여 관제서버에서 메시지를 보내고 이에 대한 응답을 받을 수 있음



## 6.2. AT Command

장치에서 사용하는 통신모듈이 AT Command를 지원한다면 이 메시지 형식을 갖추어서 AT Command를 실행하고 이에 대한 응답을 받을 수 있음

