

KPX 신재생자료취득장치 연계용 API 정의서

2023. 10. 4.



제·개정 이력

차수	개정일자	개 정 내 용 (개정요지)	작성자
0	2023.03.03	신재생자료취득장치 설치 가이드에서 API 정의서 분리	최종헌
1	2023.06.05	제주 시범사업(VPP) 구분자 추가	최종헌
2	2023.10.04	제주 시범사업(VPP) API 항목 구체화, HTTPS 통신방식 적용 등	최형욱

목 차

1. 신재생자료취득장치 연계 API 개요	1
1) 목 적	1
2) 연계 세부사항	1
2. API Specification	2
1) Analog API	2
2) Control API	4
3) On/Off Status API	5

1. API 정의 개요

1) 목 적

- 본 문서는 전력거래소 전력시장운영규칙 별표 13에 따른 신재생자료취득장치를 연계하기 위해 현장 로컬 EMS 혹은 제주 시범사업 사업자의 로컬 서버가 제공해야하는 API에 대하여 설명함 <개정 2023.10.4.>

2) 연계 세부사항

- 신재생자료취득장치에서 제공해야하는 데이터 항목 및 데이터에 대한 설명은 전력거래소 전력시장운영규칙 별표 13을 따름
- 신재생자료취득장치에서 로컬 EMS 혹은 로컬 서버로 HTTPS 기반의 API를 호출할 수 있도록 네트워크가 구성되어 있어야 함 <개정 2023.10.4.>
- 신재생자료취득장치 연동 전에 로컬 EMS 혹은 로컬 서버에 연결된 장치 ID (device ID)를 신재생자료취득장치 공급사에 제공하여야 하고, 신재생자료취득장치는 해당 장치 ID를 기반으로 API를 호출함
- 재생에너지 입찰제도 VPP자원의 경우 장치 ID (device ID)를 사전에 전력거래소에서 부여하고, 해당 ID로 API를 호출함 <신설 2023.10.4.>

2. API Specification

1) Analog API

○ Analog API 정의서 <개정 2023.10.4>

I/F 명	/kpx/ems/analog		
I/F 설명	사업소 EMS의 Analog 정보를 생성 및 전달한다.		
송신 시스템	사업소 EMS	데이터 유형	JSON
수신 시스템	전력거래소	전송방법	HTTPS
사용법	GET /kpx/ems/analog?did={DEVICE_ID}&isVpp=True False		
결과	<pre>{ "did": "string", "timestamp": 1599661015531, "localtime": 20200909231655, "operation": 0, "alertLevel": 0, "faultDesc": "string", "activePower": 0, "reactivePower": 0, "maxActivePower": 0, "targetActivePower": 0, "essCActivePower": 0, "essDActivePower": 0, "essReactivePower": 0, "essMaxActivePower": 0, "essMinActivePower": 0, "essSoc": 0, "irradiation": 0, "temperature": 0, "windDirection": 0, "windSpeed": 0, "numOperatingTurbine": 0, "lastTargetActivePowerRecvDate": 0, "lastTargetActivePowerReqDate": 0, "activePowerBySource": {"WT": 1000, "PV": 1000, "FC": 1000, "ESS": 10000}, "activePowerByDL": {"123": 1000, "456": 1000, ..., "999": 10000}, "activePowerByBus": {"123": 1000, "456": 1000, ..., "999": 10000} }</pre>		

○ 세부항목 <개정 2023.10.4>

이름	데이터유형	한글이름	설명
did	string	장치아이디	
timestamp	timestamp	절대시간	timestamp in millisecond
localtime	number	로컬시간	YYYYMMDDhhmmss
operation	number	설비상태	0: stop, 1: run, 2:trip
activePower	number	송전단 유효전력	(W)
reactivePower	number	송전단 무효전력	지상진상: +/- (VAr)
maxActivePower	number	송전단 최대 가능 출력	출력을 낼 수 있는 유효전력 최대값 (W)
targetActivePower	number	송전단 목표 출력	목표 유효 출력 (W)
essCActivePower	number	ESS 충전 유효전력	(W)
essDActivePower	number	ESS 방전 유효전력	(W)
essReactivePower	number	ESS 무효전력	지상진상: +/- (VAr)
essMaxActivePower	number	ESS Limit High	(W)
essMinActivePower	number	ESS Limit Low	(W)
essSoc	number	SOC	(%)
irradiation	number	일사량	(W/m ²)
temperature	number	온도	(°C)
windDirection	number	풍향	(°)
windSpeed	number	풍속	(m/s: meter/sec)
numOperatingTurbine	number	운영중인 터빈 수	(n/a)
lastTargetActivePowerRecvDate	number	목표출력 최신 수신 시간	YYYYMMDDHH24miss /kpx/ems/control에 전달된 값을 전달받은 시간
lastTargetActivePowerReqDate	number	목표출력 최신 요청 포함된 요청 시간	YYYYMMDDHH24miss /kpx/ems/control에 전달한 requestAt 값을 전달
activePowerBySource	json	발전원별 실시간 유효전력	(isVpp 가 True일 때) { "PV": 태양광 유효전력, "WT": 풍력 유효전력, "FC": 연료전지 유효전력, "ESS": ESS 유효전력}
activePowerByDL	json	D/L별 실시간 유효전력	(isVpp 가 True일 때) { "DLid1": 유효전력1, "DLid2": 유효전력2, ... "DLidN": 유효전력N }
activePowerByBus	json	모선별 실시간 유효전력	(isVpp 가 True일 때) { "모선id1": 유효전력1, "모선id2": 유효전력2, ... "모선idN": 유효전력N }

2) Control API

○ Control API 정의서

I/F 명	/kpx/ems/control		
I/F 설명	사업소 EMS에 Control 신호 (목표출력 제어지시) 신호를 전달한다.		
송신 시스템	전력거래소	데이터 유형	JSON
수신 시스템	사업소 EMS	전송방법	HTTPS
사용법	<p>○ 출력 Limit 제어가 가능한 경우</p> <p>POST /kpx/ems/control</p> <pre>{ "did": "0", "controlMode": "limit" "targetPower": 1234 // 최대 허용 출력값 (kW) "requestAt": 20200603050000 // YYYYMMDDHH24miss, "isVpp": True False }</pre> <p>○ 인버터별 on/off 제어가 가능한 경우 <삭제 2023.10.4></p>		
결과	<p>○ 출력 Limit 제어에 대한 응답</p> <p>POST /kpx/ems/control</p> <pre>{ "request": { "did": "0", "controlMode": "limit" "targetPower": 1234 // 최대 허용 출력값 (kW) "requestAt": 20200603050000 // YYYYMMDDHH24miss, "isVpp": True False }, "result": "success" "fail" "unsupported" }</pre> <p>○ 인버터별 on/off 제어가 가능한 경우 <삭제 2023.10.4></p>		

3) Status API

○ Status API 정의서(선택사항)

I/F 명	/kpx/ems/status		
I/F 설명	사업소 EMS에 설비별 on/off 상태를 전달해 준다.		
송신 시스템	전력거래소	데이터 유형	JSON
수신 시스템	사업소 EMS/시스템	전송방법	HTTPS
사용법	○ 설비별 on/off 상태를 요청 한다. GET /kpx/ems/status?did={DEVICE_ID}		
결과	○ 설비별 on/off 상태를 전달해 준다. GET /kpx/ems/status?did={DEVICE_ID} { "inv01": 0, // inv01 off "inv02": 0, // inv02 off "inv03": 1 // inv03 on }		