**<client variables>**

kp : P gain

ki : I gain

min\_pwm : pwm 최솟값

max\_pwm : pwm 최댓값

self.init\_weight : 자동 급이 전 사료량

self.feed\_weight : 로드셀값(사료무게)

self.target\_weight : 이번 회차 급이 후 목표 잔량(self.init\_weight - self.feeding\_weight)

self.feeding\_cmd : 급이 명령(True/False)

self.feeding\_pace : 급이 속도(kg/min)

self.feeding\_distance : 급이 거리(m) # 메시지에서 받아올 것

self.desired\_weight : PID 제어를 위한 목표 사료 무게(kg)

self.state\_event\_period : state 송신 주기(sec)

self.feeder\_stop() : 급이 중지, feeding mode 'stop'으로 변경

self.set\_feed\_size(size) : 사료 사이즈 재설정

self.ip : server ip addr

self.state\_port : state port

self.cmd\_port : cmd port

self.BUFFER = 10240 : state/cmd data buffer size

self.feeder\_ID : feeder ID

self.weight : 사료잔량(kg)

self.feeding\_size : 사료 사이즈(호)

self.feed\_motor\_pwm : 현재 feed motor pwm(0~100)

self.spread\_motor\_pwm = 0 : 현재 spread motor pwm(0~100)

self.feeding\_mode : feeding mode('stop' 'auto' 'manual')

self.feeder\_event : feeder 내 이벤트 {"remains":self.weight\_event,"motor\_state":self.motor\_event}

<client functions>

set\_feeder\_id(self, id) : feeder ID 설정

set\_feed\_size(self, size) : 사료 사이즈 설정

set\_feeding\_mode(self, mode) : feeding mode 설정

check\_feeding\_amout(self, target\_weight) : 급이 가능 확인

check\_feed\_state(self.weight) : 사료 잔량 확인

<server variables>

self.server\_ip : server ip addr

self.state\_port : state port

self.cmd\_port : cmd port

self.BUFFER : state/cmd data buffer size

## 주요 변수 초기화

self.feeder\_max\_num : 총 급이기 수(10개)

self.info : 모든 급이기 info

self.feeding\_auto\_plan : 모든 급이기 auto\_plan

self.feeder\_socket\_list : 급이기 ID와 client socket 저장 예) {"F-01":socket정보}

self.feeder\_state\_list : 급이기 ID의 연결상태 저장 예) {"F-01":True, "F-02":True, ... , "F-10":False}

<server functions>

- get 함수 -

**<통신 데이터 형식>**

## state ##

# client에서 server로 1초마다 전송(1초 주기) #

state\_msg = {

"timestamp" : time.strftime("%y/%m/%d %H:%M:%S"),

"feeder\_ID" : “F-01” or “F-02” ~~,

''ip\_address" : “0.0.0.0”,

"feed\_size" : 3 or 4 or 5 ~~,

"remains" : 10.0,

"feeding\_motor\_output" : 0,

"spread\_motor\_output" : 0,

"feeding\_mode" : 0,

"event" : {"remains\_state" : “enough feed” or “low feed”,

"motor\_state" : “running” or “stop” or “over current”},

"connectivity" : True or False

}

## cmd ##

# server에서 client로 필요 시 전송(event) #

cmd = {

"type" : "set",

"cmd" : "size",

"value" : 3 or 4 or 5}

cmd = {

"type" : "set",

"cmd" : "mode",

"value" : "auto" or "manual" or "stop"}

cmd = {

"type" : "set",

"cmd" : "id",

"value" : "F-01" or ~~}

# server의 스케줄러에서 스케줄 시간이 되면 송부 #

cmd = {

"type" : "control",

"cmd" : "start",

"value" : {

"feeding\_pace" : 급이속도(kg/min),

"feeding\_distance" : 급이범위(m),

"feeding\_amout" : 급이량(kg)

}

}

# server에서 client로 필요 시 전송(event) #

cmd = {

"type" : "control",

"cmd" : "manual",

"value" : {

"feeding\_pace" : 급이속도(kg/min),

"feeding\_distance" : 급이범위(m),

"feeding\_amout" : 급이량(kg)

}

}

cmd = {

"type" : "control",

"cmd" : "stop",

"value" : {}

}