

极简物控

产品说明书

开放式系统能源储能仿真平台

杭州极简物控科技有限公司

二〇二二年六月

目录

1 平台介绍	1
2 技术特点	2
2.1 更集约：增强型智能融合终端	2
2.2 更通用：事件-边活动网控制组态	2
2.3 更友好：标准化表格/图模配置工具	2
2.4 更高效：高性能计算与加速仿真技术	2
3 产品功能	3
3.1 实时监控	3
3.2 智能控制	3
3.3 优化调度	3
3.4 异常告警	4
3.5 数据报表	4
4 使用说明	5
4.1 入门	5
4.1.1 运行第一个控制应用	5
4.1.2 监控你的控制器运行	7
4.2 设计一个 AOE 网络	9
4.2.1 事件驱动与 AOE 网络	9
4.2.2 AOE 基本表达式	10
4.2.3 AOE 网络触发类型	13
4.2.4 AOE 事件类型	13
4.2.5 AOE 动作类型	14
4.2.6 AOE 网络的设计流程	18
4.3 通讯协议	19
4.4 配置文件编写	22
4.4.1 测点文件配置	23
4.4.2 通道文件配置	27
4.4.3 AOE 策略文件配置	28
4.3.2 AOE 变量定义	29
4.5 界面操作	34

4.5.1 应用界面	34
4.5.2 系统设置界面	35
4.5.3 文件配置界面	36
4.5.4 测点监控界面	38
4.5.5 AOE 监控界面	40
4.6 TwinCAT3 软件	41
4.6.1 从 MATLAB/Simulink 到 TwinCAT3	41
4.6.2 TwinCAT3 仿真设置	42
5 实验案例	45
5.1 实验环境	45
5.2 实验介绍	45
5.3 实验模型	45
5.4 实验原理	46
5.5 实验步骤	47
5.5.1 编写配置文件	47
5.5.2 配置文件存储	52
5.5.3 编写通信程序	52
5.5.4 仿真运行结果	53
5.6 实验总结	54
6 设备参数	55
6.1 智能开关柜	55
6.2 工业 PC	55
6.3 TwinCAT 软件	55
6.4 智能微电网能量管理控制器	55
6.5 智能微电网能量管理控制软件	56
6.6 可编程逻辑控制器	56
6.7 端子	57

1 平台介绍

极简物控的开放式系统能源储能仿真平台，基于智能微电网能量管理控制器实现系统控制功能，支持接入储能系统和光伏、风机等微电网模型，开展各类微电网能量管理实物实验或半实物仿真实验。

平台架构如示意图 1.1 所示。智能开关柜中包含开关断路器与智能电表，负责平台开合、控制并保护用电设备。控制柜中包含工业 PC、智能微电网能量管理控制器、可编程逻辑控制器与端子，负责实现微电网能量管理控制功能，并提供光伏、风机等微网模型。

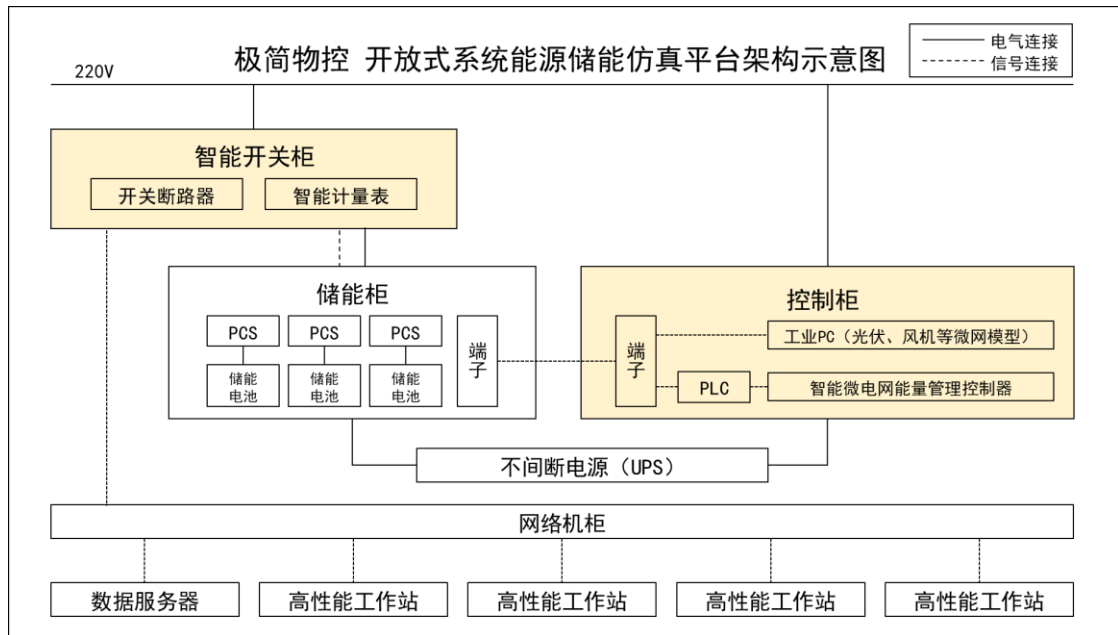


图 1.1 极简物控开放式系统能源储能仿真平台架构示意图

2 技术特点

2.1 更集约：增强型智能融合终端

极简物控所采用智能微电网能量管理控制器是一款增强型智能融合终端，解决了传统分层分级设备庞大、难以布置的问题，可以便捷的嵌入安装在控制柜中，支持数据采集、通用控制、能量管理等功能，功能丰富，可扩展性强，显著降低成本，同时便于维护管理。

2.2 更通用：事件-边活动网控制组态

极简物控首次提出通用化事件-边活动网（AOE 网）控制组态技术，使开放式系统能源储能仿真平台能够适用于不同仿真场景下的微电网实验，通过策略配置即可实现控制功能，避免了大量的重复编程，简化教学流程，便于学生学习、理解控制过程和相关现象。

2.3 更友好：标准化表格/图模配置工具

极简物控的开放式系统能源储能仿真平台配套配备表格/图模配置工具，使得教学人员或学生通过表格配置或图形化界面操作，即可完成控制策略配置，且支持观测实时变化的测点数据图表。

2.4 更高效：高性能计算与加速仿真技术

极简电力的开放式系统能源储能仿真平台采用高性能计算和轻量级数据库技术，相对于同类产品能够显著减少通信次数，释放通信带宽，减少计算资源占用。同时平台基于 TwinCAT3 实现的综合能源分布式硬件加速仿真算法则可以大幅提高仿真速度，避免了出现仿真执行效率低下的问题。

3 产品功能

3.1 实时监控

- 支持 Modbus、MQTT、DLT645、IEC104 协议
- 支持 RJ45、WIFI、RS485、RS232 接口
- 数据一站式接入与转换
- 数据本地存储
- 网页端实时监控数据图表

3.2 智能控制

- 逻辑运算
- 矩阵计算
- 线性方程组求解
- 非线性方程求解
- 混合整数线性规划（稀疏）
- 混合整数线性规划（稠密）
- 非线性规划

3.3 优化调度

- 光伏最大功率点跟踪
- 储能辅助调频
- 储能削峰填谷
- 过零检测
- 线损分析

- 直流法潮流计算
- 交流法潮流计算
- 低压继电保护
- 基于事件-边活动网（AOE 网）控制组态的其他自定义优化调度策略

3.4 异常告警

- 失败响应
- 系统告警

3.5 数据报表

- 数据日志
- 自定义报表

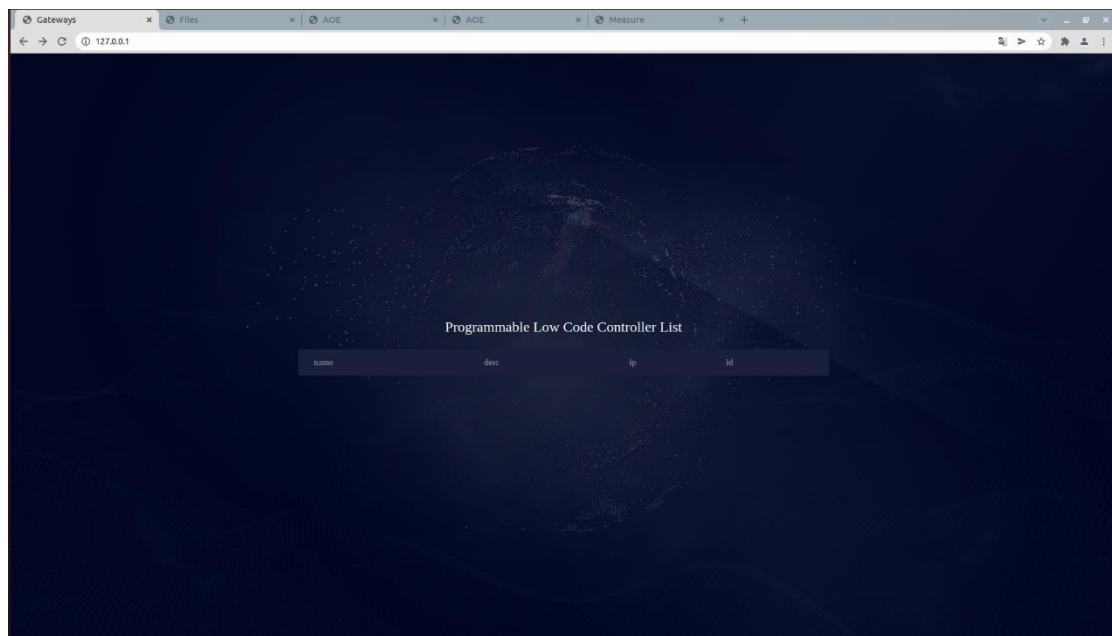
4 使用说明

4.1 入门

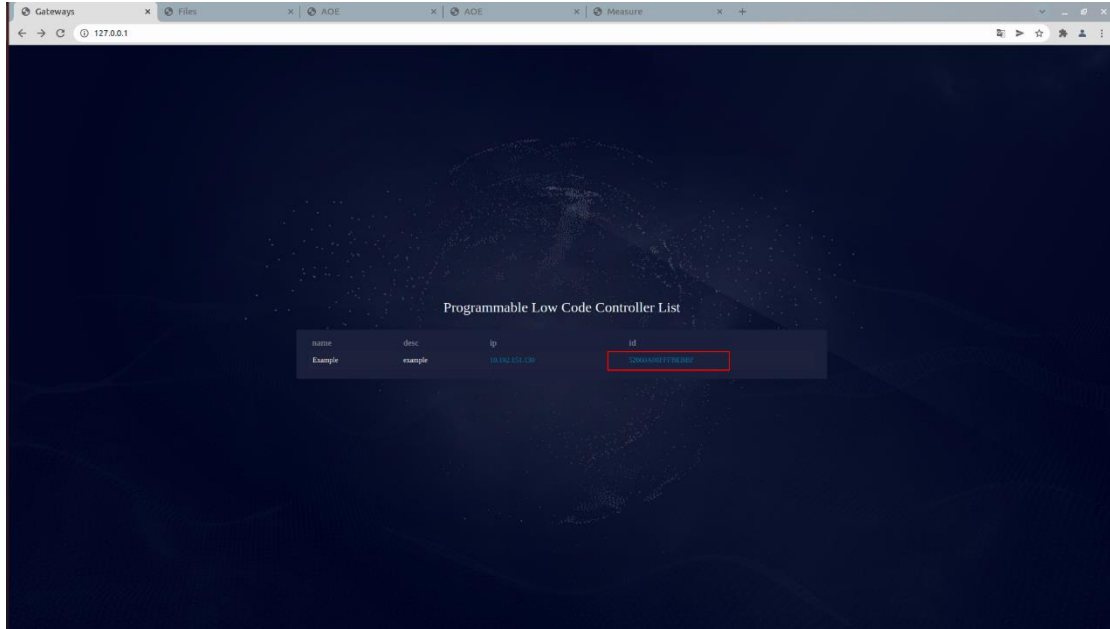
4.1.1 运行第一个控制应用

将智能微电网能量管理控制器与监控设备置于同一局域网内。

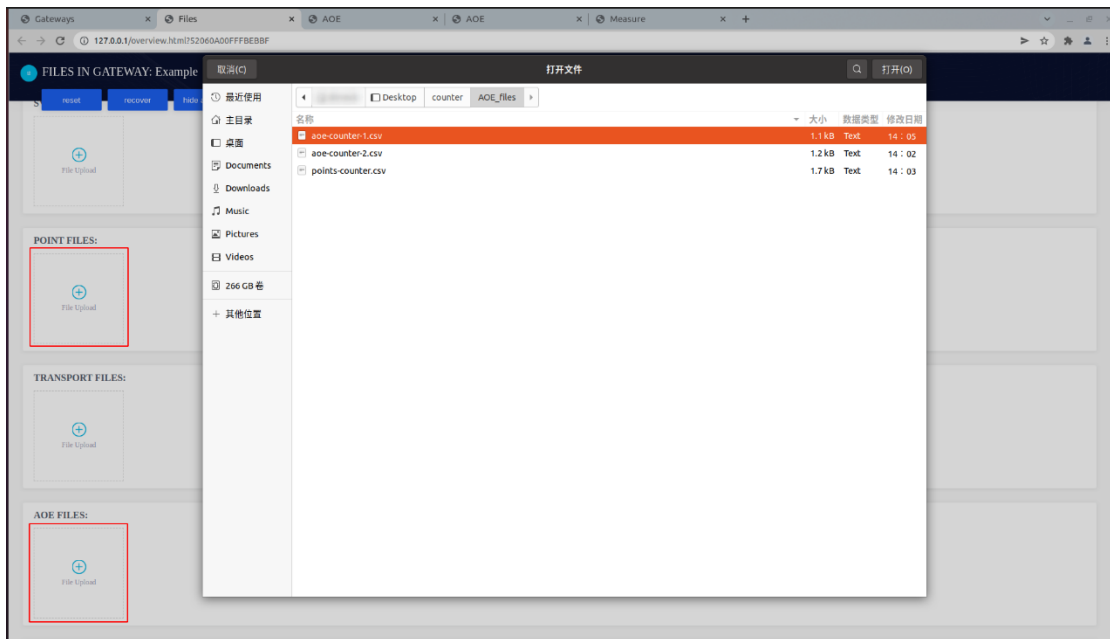
在浏览器中输入智能微电网能量管理控制器的 IP 地址，您会看到以下界面，请稍等片刻以供加载相关服务。



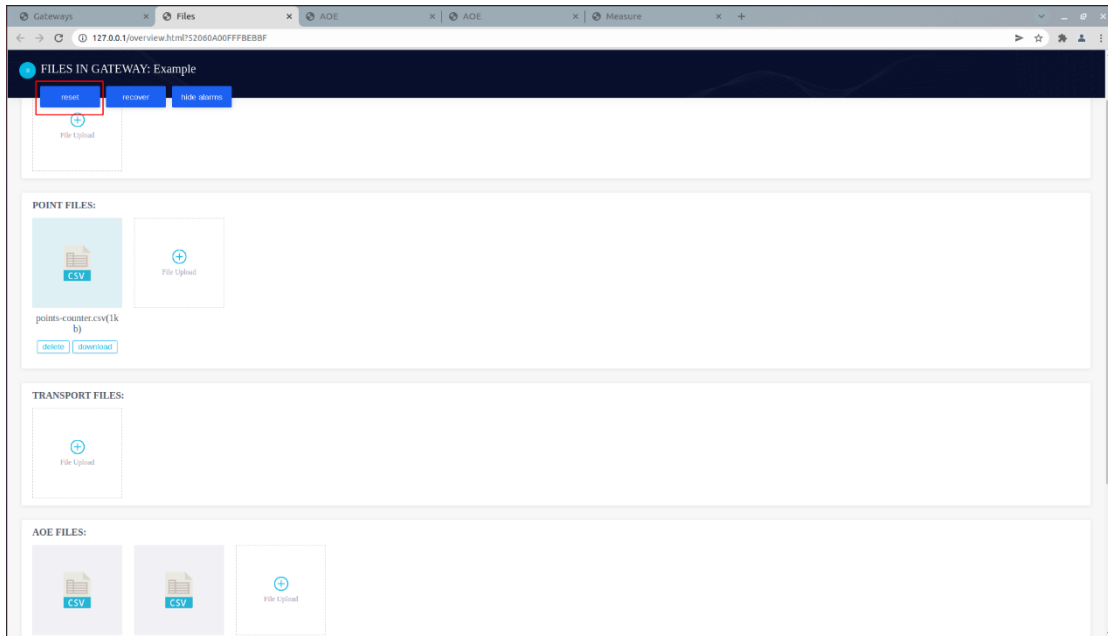
单击 **id** 进入应用配置界面。



单击“File Upload”即可上传配置文件添加文件。PLCC 允许用户上传多个配置文件



上传 POINTS, AOE 文件, 单击 RESET, 即可运行程序。



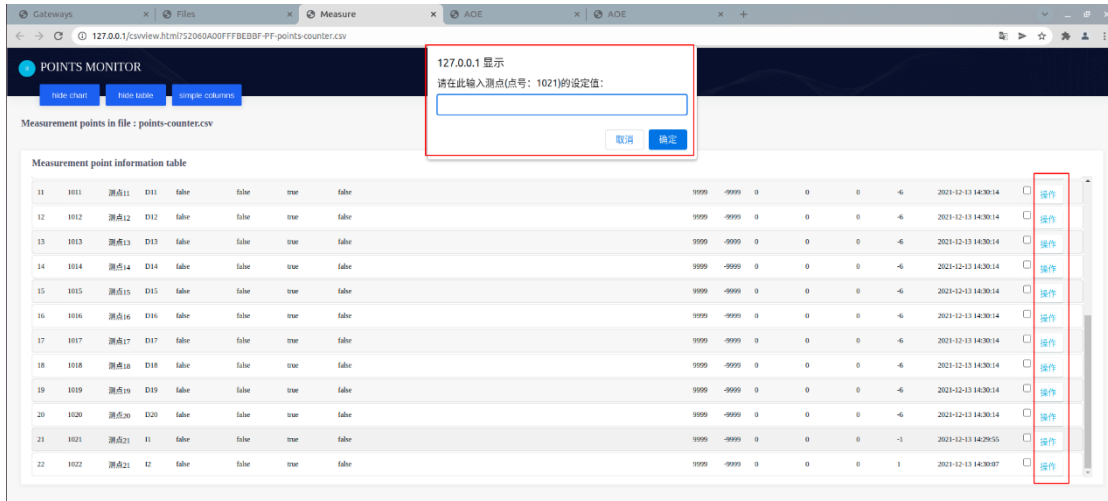
4.1.2 监控你的控制器运行

进入应用配置界面，使用 **ctrl+鼠标左键** 点击，您可以进入相应的监控页面。

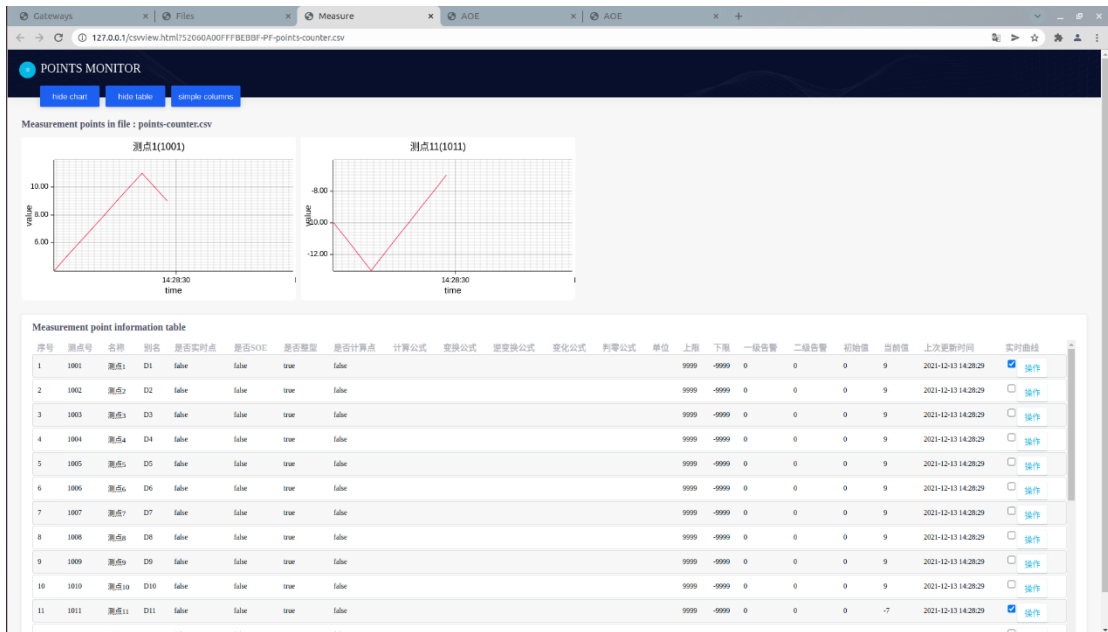
使用 **ctrl+鼠标左键** 点击 POINTS 文件，您可以打开测点监测页面，可以在这里观测测点的数据变化情况。

序号	测点号	名称	别名	是否实时点	是否SOE	是否整型	是否计算点	计算公式	变换公式	报警公式	变化公式	判断公式	单位	上限	下限	一级告警	二级告警	初始值	当前值	上次更新时间	实时曲线
1	1001	测点1	D1	false	false	true	false							9999	-9999	0	0	0	0	2021-12-13 14:31:56	<input type="checkbox"/> 操作
2	1002	测点2	D2	false	false	true	false							9999	-9999	0	0	0	0	2021-12-13 14:31:56	<input type="checkbox"/> 操作
3	1003	测点3	D3	false	false	true	false							9999	-9999	0	0	0	0	2021-12-13 14:31:56	<input type="checkbox"/> 操作
4	1004	测点4	D4	false	false	true	false							9999	-9999	0	0	0	0	2021-12-13 14:31:56	<input type="checkbox"/> 操作
5	1005	测点5	D5	false	false	true	false							9999	-9999	0	0	0	0	2021-12-13 14:31:56	<input type="checkbox"/> 操作
6	1006	测点6	D6	false	false	true	false							9999	-9999	0	0	0	0	2021-12-13 14:31:56	<input type="checkbox"/> 操作
7	1007	测点7	D7	false	false	true	false							9999	-9999	0	0	0	0	2021-12-13 14:31:56	<input type="checkbox"/> 操作
8	1008	测点8	D8	false	false	true	false							9999	-9999	0	0	0	0	2021-12-13 14:31:56	<input type="checkbox"/> 操作
9	1009	测点9	D9	false	false	true	false							9999	-9999	0	0	0	0	2021-12-13 14:31:56	<input type="checkbox"/> 操作
10	1010	测点10	D10	false	false	true	false							9999	-9999	0	0	0	0	2021-12-13 14:31:56	<input type="checkbox"/> 操作
11	1011	测点11	D11	false	false	true	false							9999	-9999	0	0	0	-8	2021-12-13 14:31:56	<input type="checkbox"/> 操作
12	1012	测点12	D12	false	false	true	false							9999	-9999	0	0	0	-8	2021-12-13 14:31:56	<input type="checkbox"/> 操作

点击操作按键可以写入测点数据。



点击实时曲线复选框可在网页上方显示数据走势图。



使用 **ctrl+鼠标左键** 点击 AOE 文件，您可以观测 AOE 的执行情况，开始与停止按键，可以控制 AOE 的开始与停止。

ID	名称	触发条件	开始时间	结束时间			开始	停止
5003	压力测试	EventDrive	2021-12-13 14:31:02	2021-12-13 14:31:02				
动作名称	源事件	源事件时间	目标事件	目标事件时间	动作时间	结束时间	失败模式	执行结果
进入判断是否超过上阈值的节点	超出阈值事件	2021-12-13 14:31:02	超过上阈值事件	2021-12-13 14:31:02	2021-12-13 14:31:02	2021-12-13 14:31:02	Default	true
进入判断是否超过下阈值的节点	超出阈值事件	2021-12-13 14:31:02	超过下阈值事件	2021-12-13 14:31:02	2021-12-13 14:31:02	2021-12-13 14:31:02	Default	true
减法器	超过上阈值事件	2021-12-13 14:30:40	结束节点	2021-12-13 14:30:40	2021-12-13 14:30:40	2021-12-13 14:30:40	Default	true
加法器	超过下阈值事件	2021-12-13 14:31:02	结束节点	2021-12-13 14:31:02	2021-12-13 14:31:02	2021-12-13 14:31:02	Default	true

ID	名称	触发条件	开始时间	结束时间			开始	停止
5004	压力测试	EventDrive	2021-12-13 14:31:00	2021-12-13 14:31:00				
动作名称	源事件	源事件时间	目标事件	目标事件时间	动作时间	结束时间	失败模式	执行结果
进入判断是否超过上阈值的节点	超出阈值事件	2021-12-13 14:31:00	超过上阈值事件	2021-12-13 14:31:00	2021-12-13 14:31:00	2021-12-13 14:31:00	Default	true
进入判断是否超过下阈值的节点	超出阈值事件	2021-12-13 14:31:00	超过下阈值事件	2021-12-13 14:31:00	2021-12-13 14:31:00	2021-12-13 14:31:00	Default	true
减法器	超过上阈值事件	2021-12-13 14:30:34	结束节点	2021-12-13 14:30:34	2021-12-13 14:30:34	2021-12-13 14:30:34	Default	true
加法器	超过下阈值事件	2021-12-13 14:31:00	结束节点	2021-12-13 14:31:00	2021-12-13 14:31:00	2021-12-13 14:31:00	Default	true

4.2 设计一个 AOE 网络

本章介绍如何设计一个 AOE 网络，首先介绍事件驱动和 AOE 网络的概念，然后分别说明 AOE 基本语法、AOE 网络触发类型、AOE 事件类型和 AOE 动作类型，最后阐述 AOE 网络的设计流程。

4.2.1 事件驱动与 AOE 网络

什么是事件驱动？ 一个事件驱动流程，由事件（Events）和动作（Actions）按照一定的逻辑关系组合而成。事件：事件是一种状态，这里用来描述被控对象。事件可以作为动作执行的条件，同时也可作为动作执行完成的标志。动作：动作是一种行为，这里用来表述控制器执行的具体策略（指令），例如给变量赋值、执行优化计算等指令。

什么是 AOE 网络？ AOE 网络是一种由多个事件驱动流程（及一些附加参数）组成的加权有向无环图。图中，每个节点表示事件，每条边表示动作，边的方向表示节点所示事件之间的递进和迁移关系，这种关系由该条边所表示的动作决定。当 AOE 网络中节点

事件发生时，则会按照节点中定义的规则，执行对应的动作（边）。当动作执行完成后，则会进入到相应的节点，按照节点定义规则判断事件是否发生。下图是一个 AOE 网络的示意，其中包含 6 个节点和 6 条边：

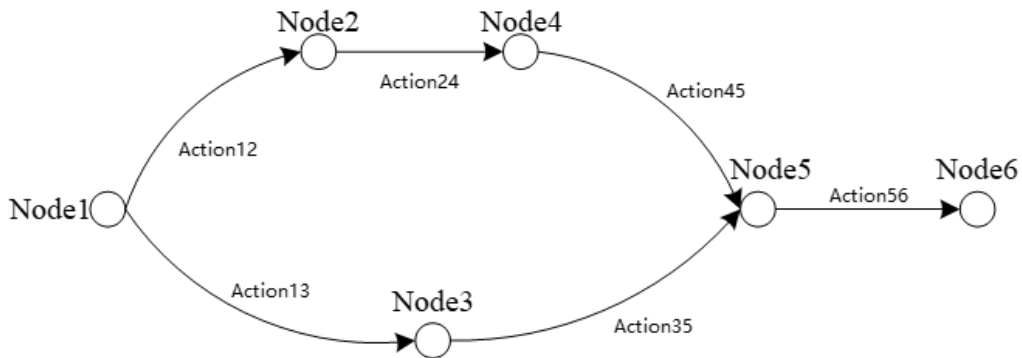


图 4.1 AOE 网络结构示意图

4.2.2 AOE 基本表达式

本平台基于 AOE 事件驱动的方式实现各项控制及仿真功能，在利用 AOE 网络进行控制策略设计时，所涉及的基本语法介绍如下：

4.2.2.1 赋值

AOE 可以直接将某一数值或表达式赋给某个变量。赋值号为冒号:或等号=，推荐使用冒号=，赋值顺序为将赋值号右边的值赋给左边。

相关示例如下：

#示例 1：直接将值 1 赋给变量 State_POINT

```
State_POINT: 1;
```

#示例 2：将由变量 Cartset_POINT 与 Cart_POINT 组成的表达式的值赋给变量

ECar

Ecar: Cartset_POINT - Cart_POINT;

4.2.2.2 基本数值运算

AOE 可完成的基本数值运算及涉及的运算符如下表所示。

名称	运算符
四则运算	+ - * /
关系运算	< <= > >= == !=
逻辑运算	&&
取余运算	%
幂运算	^
位运算	& ^^ ~ << >> @

表 4.1 AOE 基本数值运算及涉及运算符

相关示例如下：

#示例 3：简单四则运算

Iout: Iout + Igain * Ecarnl * Ts;

#示例 4：多种运算混合

Ilimout: (Iout <= Iuplim && Iout >= Idownlim) * Iout + (Iout > Iuplim) * Iuplim +
(Iout < Idownlim) * Idownlim;

4.2.2.3 内置函数

AOE 内置多种常用数学函数及实用时间获取函数，可通过向括号内传入参数进行直接调用，支持的函数及其具体功能如下表所示。

函数	功能
abs()	求绝对值
sqrt()	求平方根
exp()	求以 e 为底的指数

ln()、log10()	求自然对数（以 e 为底）、求常用对数（以 10 为底）
sin()、cos()、tan()	正弦函数、余弦函数、正切函数，单位为弧度
asin()、acos()、 atan()	反正弦函数、反余弦函数、反正切函数
sinh()、cosh()、 tanh()	双曲正弦函数、双曲余弦函数、双曲正切函数
asinh()、acosh()、 atanh()	反双曲正弦函数、反双曲余弦函数、反双曲正切函数
floor()	向下取整，即取不大于传入参数的最大整数
ceil()	向上取整，即取不小于传入参数的最小整数
round()	将传入参数进行四舍五入
signum()	符号函数，指出参数正负号，若为正则返回 1；零返回 0；负返回-1
atan2()	计算方位角，弧度表示
max()、min()	求多个参数的最大值/最小值，参数用“,”分隔
rand()	输出[0,1)之间的随机数
rand2(a,b)	输出[a,b)之间的随机数
indexOfYear()	返回日期的年
indexOfMonth()	返回日期的月
indexOfDay()	返回日期的日
indexOfMinute()	返回时间的分钟
indexOfSecond()	返回时间的秒
indexOfHour()	返回时间的小时

表 4.2 AOE 内置函数

4.2.2.4 数字表示约定

AOE 支持的数学常数和数字表示约定如下表。

类别	名称	AOE 表示
数学常数	圆周率	pi

	自然常数	e
数字表示方法	科学计数法	1e-2 1E-2
	十六进制表示	x123 _123 \$123

表 4.3 AOE 支持的数字表示约定

4.2.3 AOE 网络触发类型

AOE 网络被激活并从起始节点开始运行整个网络的行为称为**网络触发**。 AOE 网络触发类型默认为事件驱动方式，即起始节点所代表的事件满足其设定的触发条件时，起始节点被触发，继而触发后续边及节点；除此之外，根据实际场景的需要，AOE 网络触发类型也可以设定为简单重复驱动或时间驱动，使”微”能量管理器更具灵活性和普适性。

具体的 AOE 网络触发类型及说明见下表。

触发类型	说明
SimpleRepeat（简单重复驱动）	需要自行设定定时时间（触发条件参数），在设定时间下循环启用 AOE 网络
TimeDrive（时间驱动）	在设定时间节点启用 AOE 网络
EventDrive（事件驱动）	当满足事件发生条件时，AOE 网络动作

表 4.4 AOE 网络触发类型

4.2.4 AOE 事件类型

AOE 网络中的节点代表事件，节点与节点之间的边代表动作，详细定义可见事件驱动与 AOE 网络。节点的类型及其代表的事件说明如下表所示。

节点类型	说明
Condition 条件节点	Condition 节点为事件触发型。若在设定的超时时间内节点的表达式为真，则节点事件被触发，该节点触发的所有动作（即所有该

	节点发出的支路) 并行执行。
Switch 分支节点	<p>Switch 节点为分支逻辑判断, 该节点连接着两条支路, 以节点的表达式作为逻辑判断。</p> <p>触发后续动作的逻辑为: 当表达式为真时, 该节点序号为 1 的边进行动作; 表达式为假时, 序号为 2 的边进行动作 (其中边的序号是按 AOE 配置文件中定义的顺序确定)。</p> <p>特别的, 当 Switch 节点为 AOE 网入度为 0 的节点 (首节点) 时, Switch 节点直接触发, 若超时时间内表达式为真, 则执行动作 1, 为假或者发生超时事件执行动作 2。</p>
Switch2 分支节点 2	<p>Switch2 节点为分支动作判断, 该节点连接着两条支路, 节点自身无表达式 (即无需填写事件发生的条件), 而是以指向此节点的动作的执行情况来作为逻辑判断。</p> <p>触发后续动作的逻辑为: 若指向此节点的动作都执行成功, 该节点序号为 1 的边进行动作; 否则序号为 2 的边进行动作。</p>

表 4.5 AOE 事件类型

4.2.5 AOE 动作类型

动作指一个复杂的控制策略所细分出来的步骤, 在 AOE 中表现为节点之间的**有向边**。当首节点所表示的事件被触发后, 则按照节点事件中所定义的规则, 对应的动作 (边) 将被执行。

按照不同的动作内容, AOE 动作可分为以下几类。

动作类型	动作名称	含义
None	无动作	不进行任何动作
SetPoints	置点	对测点和变量赋值
Solve	方程组求解	对线性方程组进行求解
MILP	混合整数线性规划 (稀疏)	稀疏描述的混合整数线性规划问题求解
SimpleMILP	混合整数线性规划 (稠密)	稠密描述的混合整数线性规划问题求解

NLP	非线性规划	非线性规划问题求解
-----	-------	-----------

表 4.6 AOE 动作类型

4.2.5.1 None（无动作）

无动作即控制器不执行任何动作，该节点事件被触发后，直接进入下一节点事件的等待触发状态。

4.2.5.2 SetPoints（置点）

置点动作可以进行测点的赋值，测点即指测点配置文件中用户已定义的对象；也可以进行变量的定义，此处的变量指本控制策略中涉及到的非测点对象，可用于定义控制过程中出现的数学计算式、声明物理意义的中间变量、为减少策略撰写工作量而对表达式整体标识一个别名等等。

置点动作可以直接赋值或赋予表达式，测点或已定义的变量可用于定义新变量。赋值对象的类型可以为数字量或者模拟量。

4.2.5.3 Solve（方程组求解）

可求解线性方程组 $Ax = b$ ，其中 A 为系数矩阵， x 为未知数组成的向量， b 为常数项向量。

#1) 方程组: #

<方程式 1>;

<方程式 2>;

...

#2) 变量声明: #

<变量名 1>,<变量名 2>,...

4.2.5.4 MILP 与 SimpleMILP (混合整数线性规划)

此动作用于在给定可行域内，求解线性约束下线性目标函数的最优解及最优值。

MILP 全称为 Mixed-integer linear programming，即混合整数线性规划，指目标函数和约束条件均为线性，部分决策变量限制为整数的数学规划问题。具体来说，其约束条件支持等式约束及不等式约束；变量类型包括 0-1 变量、整型变量、实数型变量。

在本控制器的动作设置中，MILP 和 SimpleMILP 区别在于前者的约束为稀疏矩阵，即后者则为稠密矩阵。通常认为矩阵中非零元素的总数比上矩阵所有元素总数的值小于等于 0.05 时的矩阵为稀疏矩阵。

以混合整数线性规划 (MILP) 为例，其动作的设置参数包括：

#1) 目标函数：#

min/max(<表达式>)

#2) 约束条件：#

<bool 表达式 1>;

<bool 表达式 2>;

...

#3) 变量声明：#

<变量名 1>:<变量类型 1>，

<变量名 2>:<变量类型 2>，

...

其中变量类型包括以下三类:

变量类型	含义
1	0-1 二进制变量
2	整型变量
3	实数型变量

表 4.7 AOE 变量类型

实际列写时, 按顺序写下目标函数、约束条件、变量声明的标准格式式子即可, 上例中##间的内容不必写入。

4.2.5.5 NLP (非线性规划)

此动作用于在给定可行域内, 求解约束下目标函数的最优解及最优值, 其中目标函数和约束条件至少有一个为非线性。

NLP 全称为 **Nonlinear programming**, 即非线性规划, 指目标函数和约束条件至少有一个为非线性的数学规划问题。具体来说, 其约束条件支持等式约束及不等式约束; 变量类型包括 0-1 变量、整型变量、实数型变量。

#1) 目标函数: #

<表达式>

#2) 约束条件: #

<表达式 1>:[表达式下限/表达式上限];

<表达式 2>:[表达式下限/表达式上限];

...

#3) 变量声明: #

<变量名 1>:[变量下限/变量上限/初始值],

<变量名 2>:[变量下限/变量上限/初始值],

...

注：1.若无上下限对应位置不写，如 $x[3]$ 表示变量 $x \leq 3$ ；

2.变量声明若无初始值要求，可简写为[变量下限/变量上限]。

4.2.6 AOE 网络的设计流程

对于特定可控制对象和控制目标，AOE 设计流程包含**设计基于事件驱动的控制策略**和**构建 AOE 网络**两步。

4.2.6.1 设计基于事件驱动的控制策略

1. 单个动作进行计算时需要用到的中间量必须由测点获取、变量定义或者在前驱动作中通过置点确定。这里的前驱动作指事件发生前需要被执行的所有动作。

2. 确定每个动作开始执行时需要满足的事件条件和前驱动作，构建控制策略的拓扑逻辑关系，其中开始执行时需要满足的事件条件指动作执行前需要满足的条件。

3. 确定每个动作执行完成的事件条件、事件的超时时间，以及执行失败后需要采取的措施。其中：执行完成的事件条件指动作执行后变量或测点需要满足的条件，根据控制策略具体情况确定需要满足的条件；事件的超时时间指前驱动作执行后等待事件发生的最长时间；执行失败后需要采取的措施指动作执行失败后可以选择的几种处理措施，包括：尝试重新执行失败的动作、忽略本次失败并

不影响其它动作、停止受到影响的动作、停止本次控制策略的执行。

4.2.6.1 构建 AOE 网络

将策略分解为多个事件和由事件触发的动作之后，将这两类元素按照一定的执行逻辑组织起来形成 AOE 网络即可完成控制策略的设计。

1. 动作和节点在计算中依赖的变量只能来自测点、变量定义或已经被执行过的动作中计算出的变量。

2. 确定每个动作的顺序以及动作发生、需要满足的事件后，用节点表示事件，事件发生导致被执行的动作作为从这个节点出发的有向边，每条有向边指向的节点表示动作完成后需要判断是否发生的事件。

3. 按照上述逻辑组织所有的节点和边，构成 AOE 网络拓扑。

4.3 通讯协议

本平台支持 Modbus、IEC104、DL/T645、MQTT 等电力领域常见通信协议，实现高可靠、高效率通信。用户可以根据产品与微电网中各设备的通讯状况灵活选择通讯方式，部分支持的通讯协议如下表所示：

通讯协议		介绍	可配置项	
Modbus 通讯协议	Modbus TCP/IP 服务端	Modbus 是一种串行通讯协议，已经成为工业领域通信协议的业界标准。而	1) 通道名称 2) 连接个数 3) 服务端口 4) 连接名称	9) 通信协议: XA 10) 一次读寄存器数上限 11) 一次读开关数上限 12) 一次写寄存器上限

		<p>Modbus TCP/IP 是基于以太网 TCP 的 Modbus 网络通信协议。当通讯设备作为从站时，采用 ModBus TCP/IP 服务端通信协议，从站有多个，对主站请求给予相应。</p>	<p>5) 测点个数: 13) 一次写开关数上限 6) 客户端 IP 14) 轮询周期和超时时间 7) 客户端端口 15) 通道状态测点号 8) slave id</p>
	Modbus TCP/IP 客户端	<p>当设备采用 TCP 网络通信且作为主站时，采用 ModBus TCP/IP 客户端通信协议。主站唯一，向各从站发送请求帧，主动建立通讯连接。</p>	<p>1) 连接名称 7) 一次读寄存器数上限 2) 测点个数 8) 一次读开关数上限 3) 服务端 IP 9) 一次写寄存器上限 4) 服务端端口 10) 一次写开关数上限 5) slave id 11) 轮询周期和超时时间 6) 通信协议: XA 12) 通道状态测点号</p>
	Modbus RTU Over TCP/IP 服务端	<p>Modbus RTU Over TCP/IP 是使用 TCP 层传输 Modbus-RTU 报文的 Modbus 通信协议。当通讯设备作为从站时，采用 Modbus RTU Over TCP/IP 服务端通信协议，从站有多个，对主站请求给予相应。</p>	<p>1) 通道名称 9) 通信协议: ENCAP 2) 连接个数 10) 一次读寄存器数上限 3) 服务端端口 11) 一次读开关数上限 4) 连接名称 12) 一次写寄存器上限 5) 测点个数: 13) 一次写开关数上限 6) 客户端 IP 14) 轮询周期和超时时间 7) 客户端端口 15) 通道状态测点号 8) slave id</p>
	Modbus RTU Over TCP/IP 客户端	<p>当设备使用 TCP 层传输 Modbus-RTU 报文通信且作为主站时，采用 Modbus RTU Over TCP/IP 客户端通信协议。</p>	<p>1) 连接名称 7) 一次读寄存器数上限 2) 测点个数 8) 一次读开关数上限 3) 服务端 IP 9) 一次写寄存器上限 4) 服务端端口 10) 一次写开关数上限 5) slave id 11) 轮询周期和超时时间 6) 通信协议: 12) 通道状态测点号</p>

		主站唯一，向各从站发送请求帧，主动建立通讯连接。	ENCAP
	ModBus RTU 服务端	Modbus RTU 是采用 RTU 报文的 Modbus 串口通信协议。当通讯数据量大而且是二进制数值时，多采用该协议。当设备作为从站时，采用 ModBus RTU 服务端通信协议，每个从站都有唯一编号（即设备地址），对主站请求给予响应。	1) 通道名称 2) 测点个数 3) 文件路径 4) 波特率 5) slave id 6) 通信协议:RTU 7) 一次读寄存器数上限 8) 一次读开关数上限 9) 一次写寄存器上限 10) 一次写开关数上限 11) 轮询周期和超时时间 12) 通道状态测点号
	ModBus RTU 客户端	当设备采用 RTU 串口通信且作为主站时，采用 ModBus RTU 客户端通信协议。主站唯一，通过单播和广播两种方式向从站发出请求。	1) 通道名称 2) 连接个数 3) 波特率 4) 文件地址 5) 连接名称 6) 测点个数 7) 描述 8) 优先级 9) slave id 10) 通信协议: RTU。 11) 一次读寄存器数上限 12) 一次读开关数上限 13) 一次写寄存器上限 14) 一次写开关数上限 15) 轮询周期和超时时间 16) 通道状态测点号
IEC 104 通讯协议	IEC 104 服务端	IEC 60870-5-104 规约，简称 IEC 104，由国际电工委员会制定，该标准为远动信息的网络传输提供通信规约依据。设备作为服务端采用 IEC 104 服务端通讯协议。	1) 通道名称 2) 连接个数 3) 服务端口 4) 连接名称 5) 测点个数 6) 客户端 IP 7) 客户端端口 8) 通道状态点号
	IEC 104	设备作为客户端采	1) 通道名称 5) 客户端 IP

	客户端	用 IEC 104 客户端通讯协议。	2) 服务端端口 3) 连接名称 4) 测点个数	6) 客户端端口 7) 通道状态点号
	DL/T645 通信协议	DL/T 645 是多功能电能表与数据终端设备进行数据交换式的物理连接和协议的规范和标准。	1) 通道名称 2) 连接个数 3) 服务端端口 4) 文件地址 5) 数据位 6) 停止位 7) 校验位 8) 连接名称	9) 测点个数 10) slave id 11) 客户端端口 12) 通道状态点号 13) 轮询周期 14) 超时时间 15) 通道状态测点号
	MQTT 通讯协议	MQTT 协议是一种消息列队传输协议,采用订阅、发布机制,既保证必要的数据的交换,又避免了无效数据造成的储存与处理,在工业物联网中得到了广泛应用。	1) 通道名称 2) 服务端 IP 3) 服务端端口 4) 测点个数	5) 通道状态测点号 6) 读主题 7) 写主题

表 4.8 通讯协议

4.4 配置文件编写

平台中的智能微电网能量管理控制器与被控对象（储能系统、工业 PC 提供的微网模型、Simulink 模型等）进行通信，得到关于被控对象的测点信息，利用测点信息实现特定的控制策略，从而完成控制过程。控制策略由事件驱动的 AOE 网络实现，需要将控制策略转换为一个 AOE 网络。

与此对应的，能量管理系统控制器文件配置分为三个部分：测点配置、通信通道配置和 AOE 配置。

测点配置用于描述测点所需要的属性。

通信通道配置用于描述控制器与被控对象的通信方式，并依据测点配置文件中的测点属性，给出保存测点值的寄存器信息和地址。

AOE 配置是建立 AOE 网络模型来实现所需的控制策略。

三个配置文件可以用 excel 进行编辑，在 excel 表格中编写完成之后，将表格另存为.csv 格式（纯文本形式存储表格数据）文件使用。下面将分别介绍三个配置文件。

4.4.1 测点文件配置

如前所述，测点文件配置是对每一个测点的属性进行定义。

测点可以分为两类：采集点和计算点。**采集点**是指通过通信（串口、以太网、wifi、蓝牙、CAN 总线等方式）从设备采集上来的量测点；**计算点**是配置了计算公式的测点，计算点公式里面一般包含其他采集点的别名或点号，或其他计算点的别名或点号。要注意的是，计算点之间不能存在循环依赖，例如计算点 A 公式里面包含了计算点 B，计算点 B 的公式里面包含了计算点 A，这是不合法的。

4.4.1.1 采集点的属性解释

按照配置文件 excel 中列的顺序排列，采集点的属性解释包括：

1. 序号：u64 类型，是测点配置表格里给各个测点的编号，只是配置表格中的一个标记，一般从 1 开始连续对测点进行编号，方

便表格编写，不影响控制器程序功能。

2. 点号：u64 类型，是程序里给这个测点的编号 id，可以自己定义，使用时，测点配置文件和通信配置文件中具有相同点号的测点一一对应，完成对接。

3. 名称：字符串类型，可包含中文，一般用“测点”+序号的格式来定义，注意不要出现回车符等特殊字符。

4. 别名：字符串类型，由英文字母、数字与下划线组成，可以作为变量名用于计算点计算公式、AOE 条件判断公式、变量定义公式和具体动作中。

5. 是否离散：Bool 类型，有 TRUE 和 FALSE 两个选项，用于表示测点是使用 i64 还是 f64 方法存储。一般开关量使用 TRUE，表示测点是离散值，为整型数据；模拟量使用 FALSE，表示测点是连续值，为浮点型数据。

6. 是否是计算点：Bool 类型，有 TRUE 和 FALSE 两个选项，用于表示测点是采集点还是计算点。采集点该属性为 FALSE。

7. 计算公式：采集点属性中，计算公式不起作用。

8. 变换公式：对设备发来的数据进行函数变换后作为测点的值。该公式中的变量只能是该测点本身，也可以是简单的一个数字，后者将会和采集的数值相乘作为测量值，如果计算点配置了该测点作为变量，计算过程用的是变换后的值。

9. 逆变换公式：与变换公式对应，用于在遥控遥调时，目标值

经过变换再发给设备。例如：变换公式是 10， 则逆变换公式为 0.1； 变换公式为：测点别名²，则逆变换公式为：sqrt(测点别名)。

10. 变化公式：该公式用于判断是否需要变化上传，如果计算结果>0 则表示该次采集到的点需要上传。

11. 判零公式：该公式用于判断测点是否为 0，用于消除一些设备有零飘现象的影响，该公式中的变量只能是该测点本身，例如：测点别名 < 0.01。

12. 单位：字符串类型，表示测点的实际物理意义下的单位。

13. 上限值：可以为空或一个数字，用于坏数据辨识，如果采集点的值大于该值则说明是坏数据，本次采集值会被抛弃。

14. 下限值：可以为空或一个数字，用于坏数据辨识，如果采集点的值小于该值则说明是坏数据，本次采集值会被抛弃。

15. 告警级别 1 判据：公式，用于判断是否需要告警，例如计算结果>0，将会产生告警信息。

16. 告警级别 2 判据：公式，用于判断是否需要告警，例如计算结果>0，将会产生告警信息。

17. 是否实时：Bool 类型，有 TRUE 和 FALSE 两个选项。当属性是 TRUE 时，每次采集均会发布到云端，变化公式将不再起作用，常常用于测试，或某些比较重要的测点。当属性是 FALSE 时，不会实时上传。

18. 是否 SOE：Bool 类型，有 TRUE 和 FALSE 两个选项。这

里 SOE 表示事件顺序记录(Sequence Of Event), 当属性是 TRUE 时, 会将该测点状态发生的变化及其时间保存起来, 后面查询时可按照时间顺序排列测点变化事件。

19. 默认值: 测点的初始值。
20. 本地地址: u64 类型, 是本地程序里的地址。
21. 备注: 可以用来解释测点的具体作用。

4.4.1.2 计算点的属性解释

按照配置文件 excel 中列的顺序排列, 采集点的属性解释包括:

1. 序号、点号、名称: 与采集点相同。
2. 别名: 字符串类型, 由英文字母、数字与下划线组成, 可以当做变量名用于其他计算点的计算公式、AOE 条件判断公式、变量定义公式、设点动作中。
3. 是否离散: 与采集点相同。
4. 是否是计算点: 计算点该属性为 TRUE。
5. 计算公式: 表示计算点的取值的计算公式, 计算公式的写法详见 AOE 基本语法。
6. 变换公式、逆变换公式、判零公式: 不起作用。
7. 变化公式、单位、上限值、下限值、告警级别 1 判据、告警级别 2 判据、是否实时: 与采集点相同
8. 是否 SOE: 不起作用。
9. 默认值: 测点的初始值。

10. 备注：可以用来解释测点的具体作用。

4.4.1.3 属性说明

告警：告警信息会发送、保存（如果 `isHisDb=true`），当告警级别发生变化或复位（告警消失）均会发出消息，多次采集但告警级别不变，则只会发送最开始发生的告警。

4.4.1.4 测点(Point)文件相关规则与限制

测点配置文件的名称格式为"`points-aoe-XXX.csv`"，其中"`XXX`"由用户自定义。

单个文件大小不能超过设置，目前默认是不超过 2M。

测点文件增加删除后需要 `reset` 才能生效。

4.4.2 通道文件配置

通道文件配置可以分为通道参数配置和测点寄存器参数配置两个部分。

4.4.2.1 通道参数配置

目前通信协议仅支持 Modbus RTU 和 Modbus TCP 通信协议，后期会支持 IEC104 协议。控制器可以作为服务端，也可以作为客户端。

若采用 Modbus 通讯协议，通道参数配置的内容包括客户端 IP、服务端 IP、客户端端口、服务端端口、slave id、通信协议等。

4.4.2.2 通信通道（transport）文件相关规则与限制

命名规则：通信通道（transport）文件命名要求以(xa-

mbc,encap-mbc,rtu-mbc,tcp-mdb,rtu-mbd)开头，这些开头分别表示不同的协议，不得修改和删去。

文件名开头	协议	服务端/客户端
tcp-mbd	Modbus TCP	服务端
rtu-mbd	Modbus RTU	服务端
xa-mbc;encap-mbc	Modbus TCP	客户端
rtu-mbc	Modbus RTU	客户端
dlt645	DL /T 645	服务端
iec104c	DL /T 645	客户端
iec104d	DL /T 645	服务端
mqtt	DL /T 645	/

表 4.9 通信通道 (transport) 文件相关规则与限制

文件大小：单个文件大小不能超过设置，目前默认是不超过 2M 大小。

文件上传：通道文件删除、更换或增加后需要 reset 才能生效，如果同名的通道文件更新不会增加新的文件，不需要 reset 通道也会自动生效。

4.4.3 AOE 策略文件配置

4.4.3.1 AOE 网络定义

AOE 网络定义部分，定义了表格之中的 AOE 网络个数和整体信息以及一些变量的初始值，其配置内容包括 AOE ID、是否启用、名称、触发条件（及触发条件参数）、变量初始值。

AOE ID	是否启用	名称	触发条件	触发条件参数	变量初始值
<u64>	<Bool>	<String>	SimpleRepeat TimeDriven	<u64> <String>	<String>:<Float>

			EventDriven	<\>	
--	--	--	-------------	-----	--

表 4.10 AOE 网络定义配置内容

1. AOE ID: 不同 AOE 网络的标识, 用<u64> (64 位无符号整数) 表示。(注: AOE ID 需从 501 后开始编号, 因为前 500 已被系统占用)

2. 是否启用: 决定该 AOE 网络是否启用, 用<Bool> (布尔类型) 表示。

3. 名称: 表达 AOE 网络的功能, 便于理解, 用<String> (表示字符串类型) 表示。

4. 触发条件分为三类, 即 SimpleRepeat (定时触发)、TimeDriven (时间驱动)、EventDriven (事件驱动)。

5. 触发条件参数: 参数配置格式如下表

触发条件	说明
SimpleRepeat	触发条件参数为重复驱动之间的时间间隔, 用正整数<u64>表示, 单位为 ms
TimeDriven	表示执行计划, 写法参照 cron 表达式, 在设置时间点触发
EventDriven	为缺省值, 无需设置

表 4.11 AOE 网络触发条件参数

6. 变量定义方式: <变量名 1>:<表达式 1>;<变量名 2>:<表达式 2>;...变量名为<String>, 表达式为<Float> (32 位浮点数)

注: 时间驱动的写法为标准的 cron 表达式。

4.3.2 AOE 变量定义

变量的定义包含**变量名称**和**变量的表达式**, 变量的表达式中可以使用测点值、变量和常数, 若使用预先载入的测点表内的测点

名，当测点名对应的测点数据在动作中被更新时，变量的信息也会被更新。

定义格式为：

AOE ID	变量定义
<u64>	<变量名 1>:<表达式 1>

表 4.12 AOE 变量定义格式

对于**无需公式表达**的变量定义，可以在 AOE 网络定义的“变量初始值”一栏中将名称和初始值定义好。

对于**有公式表达**的变量定义，公式可以包括测点相关的变量（与计算点类似），也可以包括第一类中的变量，和第二类中的变量，该类变量的初始值由对应测点和变量的初始值计算而来，无需设置。公式中的变量写法见本节末尾表格。

变量声明需要规定变量生效的 AOE 网络。多个 AOE 有同样的变量定义可以只定义一次，在“AOE ID”中用分号隔开多个 AOE 的 ID 即可，声明的方式为：<AOEID1>;<AOEID2>;...

4.4.3.3 AOE 事件定义

AOE 事件定义（节点声明）配置内容：AOE ID、节点 ID、名称、节点类型、超时时间、表达式。

AOE ID	节点 ID	名称	节点类型	超时时间	表达式
<u64>	<u64>	<String>	Switch Condition	<u64>	<Expr>

表 4.13 AOE 事件定义格式

1. AOE ID: 声明节点属于哪个 AOE 网络，用<u64>（64 位无符号整数）表示。

2. 节点 ID: 一个 AOE 网络下节点编号。
3. 名称: 便于理解事件内容, 往往是对表达式的简要文字解释, 用<String> (表示字符串类型) 表示。
4. 节点类型分为两类: Switch (分支节点)、Condition (条件节点)。具体含义及参数详见前文 AOE 事件类型。
5. 超时时间: 超时时间用正整数表示, 含义是在超时时间内表达式成立才为真, 若超出时间则判断为假; 当设置为 0 时, 认为超时时间为无穷大; 单位为 ms, 具体设置根据实际控制决定。
6. 表达式: 表达式格式详见前文 AOE 基本语法。

4.4.3.4 AOE 动作定义

AOE 动作定义 (边声明) 配置内容: AOE ID、首尾节点、动作名称、失败模式、动作类型、动作参数。

AOE ID	首尾节点	动作名称	失败模式	动作类型	动作参数
<u64>	<u64>;<u64>	<String>	Default	None	<\>
			Ignore	SetPoints	<String>:<Expr>;
			StopFailed	Solve	...
			StopAll	MILP	<Expr>
				SimpleMil	<Expr>
				p	<Expr>
				NLP	<Expr>

表 4.14 AOE 动作定义

1. AOE ID: 声明动作属于哪个 AOE 网络, 用<u64> (64 位无符号整数) 表示。
2. 首尾节点: 边需要声明首尾节点, 用一对正整数表示, 格式

为：<尾节点 ID>;<首节点 ID>。边的方向为：尾节点→首节点。节点 ID 与 AOE 事件定义中相对应。

3. 动作名称：便于理解动作内容，是对动作的简要文字解释，用<String>（表示字符串类型）表示。

4. 失败模型：指动作执行失败后需要采取的措施。执行失败后可以选择以下几种处理措施：**Default**（尝试重新执行失败的动作）、**Ignore**（忽略本次失败并不影响其它动作）、**StopFailed**（停止受到影响的动作）、**StopAll**（停止本次控制策略的执行）。

5. 动作类型：分为 **None**（无动作）、**SetPoints**（置点）、**Solve**（方程组求解）、**MILP** 与 **SimpleMilp**（混合整数线性规划）、**NLP**（非线性规划）。其中，**SetPoints** 除了对测点设值之外，还可以进行变量定义和计算，动作参数分为数字量和模拟量。动作的含义及其参数详见 前文 AOE 动作类型。

6. 动作参数：根据动作类型不同而表示不同的含义：

动作类型为无动作时，动作参数缺省；

动作类型为置点时，动作参数为字符串和表达式，并按照变量类型分为数字量和模拟量两部分，置点动作既可以进行变量的定义，也可以把数字量或者模拟量设置给测点，其中字符串表示变量或测点名称，表达式表示变量定义式或者测点被设置的值；

动作类型为线性方程组求解、稀疏混合整数线性规划问题求解、稠密描述的混合整数线性规划问题求解和非线性规划问题求解

时，动作参数为描述对应的数学模型的字符串。

示例 1: None (无动作)

AOE ID	首尾节点	动作名称	失败模式	动作类型	动作参数
5001	1;2	事件不动作	Default	None	

动作含义：ID 为 5001 的 AOE 网络，节点 1 至节点 2 无动作。

示例 2: SetPoints (置点)

AOE ID	首尾节点	动作名称	失败模式	动作类型	动作参数
5002	2;3	变量赋值	Default	SetPoints	a:1;

动作含义：ID 为 5002 的 AOE 网络，节点 2 至节点 3 进行赋值动作，测点 a (或变量) 赋值为 1。

示例 3: Solve (方程组求解)

AOE ID	首尾节点	动作名称	失败模式	动作类型	动作参数
5003	2;3	方程组求解	Default	Solve	$x_1 + 5/3 * x_2 + \min(2,5) * x_3 = 1;$ $3 * x_2 = 4 * 3;$ $3 * x_2 + \sin(8-2) * x_3 = 7;$ x_1, x_2, x_3

动作含义：ID 为 5003 的 AOE 网络，节点 2 至节点 3 进行方程组求解动作。

示例 4: MILP 与 SimpleMilp (混合整数线性规划)

AOE ID	首尾节点	动作名称	失败模式	动作类型	动作参数
5004	2;3	混合整数线性规划	Default	MILP	$\max((5*2)*x_1 + 3*x_2 + 2*x_3 + (10-3)*x_4 + 4*x_5);$ $2*x_1 + (2*4)*x_2 + 4*x_3 + 2*x_4 + \max(1,5)*x_5 \leq 5*2;$ $x_1:1, x_2:1, x_3:1, x_4:1, x_5:1$

动作含义：ID 为 5004 的 AOE 网络，节点 2 至节点 3 进行混合整数线性规划动作。

示例 5: NLP (非线性规划)

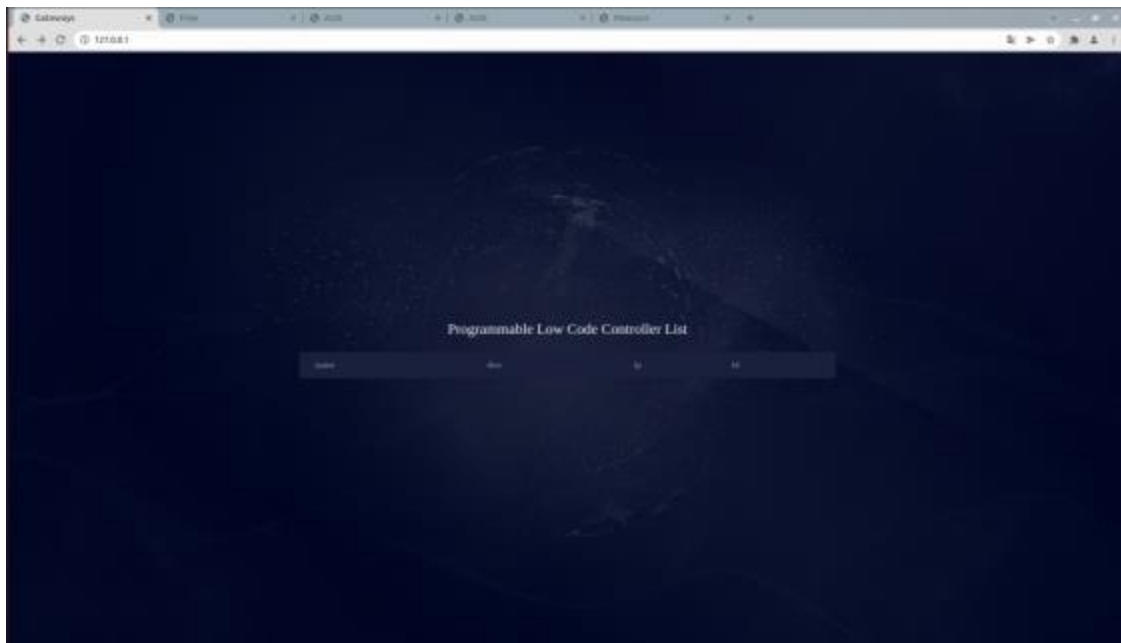
AOE ID	首尾节点	动作名称	失败模式	动作类型	动作参数
5005	2;3	非线性规划	Default	NLP	$x1 * x4 * (x1 + x2 + x3) + x3;$ $x1 * x2 * x3 * x4: [25/2e19];$ $x1^2 + x2^2 + x3^2 + x4^2: [40/40];$ $x1: [1/5], x2: [1/5], x3: [1/5], x4: [1/5]$

动作含义: ID 为 5005 的 AOE 网络, 节点 2 至节点 3 进行非线性规划动作。

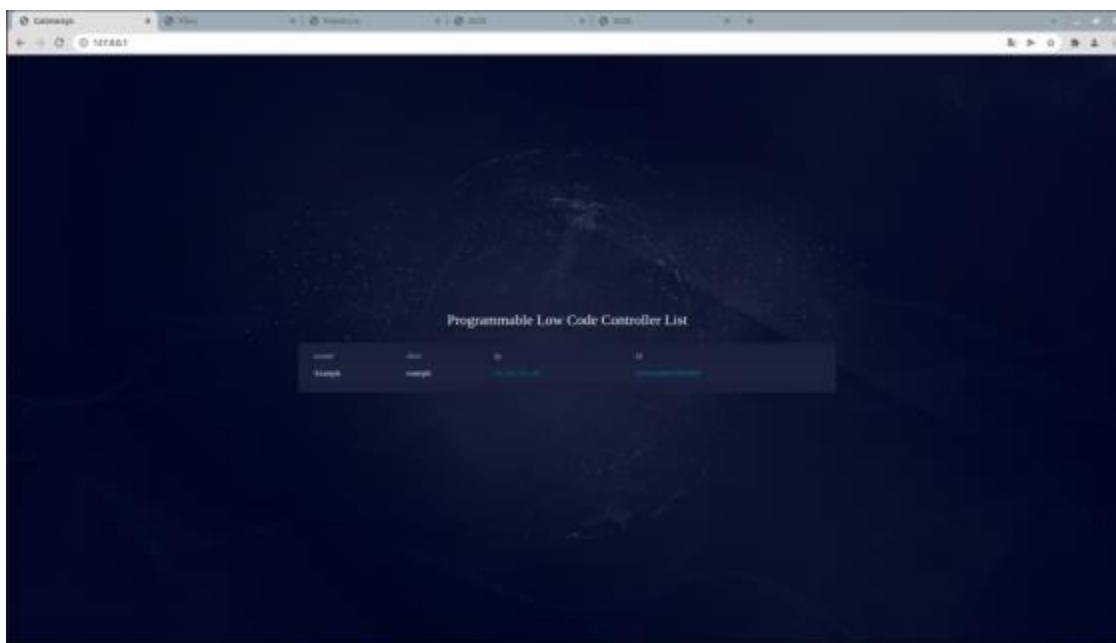
4.5 界面操作

4.5.1 应用界面

在浏览器中输入智能微电网能量管理控制器 IP 地址, 即可进入应用界面。



稍等片刻以供加载相关服务。



该界面中，单击 **ip** 地址即可进入系统设置界面，单击 **id** 即可进入文件配置界面。

4.5.2 系统设置界面

系统设置界面如下图所示。



在该界面中，可以设置此台设备的名称和描述，并配置数据转发和告警转发、数据上云等。

4.5.3 文件配置界面

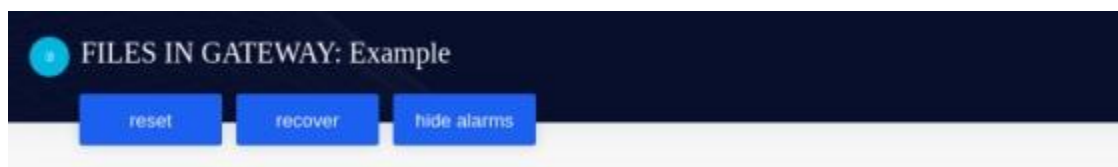
文件配置界面如下图所示。



文件配置界面总共分为三部分，按键操作区、文件配置区以及最下面的运行告警区。

4.5.3.1 按键操作区

在该区域中，共有三个按键，Reset、Recover 以及 Hide/Show alarms。



Reset:该按键可以复位 AOE 程序。

Recover:该按键用来恢复 LCC 控制器程序配置为初始状态，即停止所有 AOE 程序运行，并清空所有 AOE 配置文件。

hide/show alarms:该按键用以隐藏/显示该文件配置界面的运行告

警区域。

4.5.3.2 文件配置区

在该区域中可以上传、下载、删除、查看、监控各程序配置文件。



单击“File Upload”，可以上传程序配置文件，LCC 允许用户上传多个并列的配置文件，配置文件为 csv 格式或 svg 格式。当配置文件格式错误时，界面将会提示错误的原因，并指出具体的错误位置。



单击“delete”，可以删除选中的程序配置文件。

单击“download”，可以下载选中的程序配置文件。

单击程序配置文件，可以在线查看配置文件内容。

按住 **ctrl** 并单击程序配置文件，可以进入程序监控界面，监控配置文件执行情况。

文件说明:

SVG FILES: 可以上传 SVG 模型, 文件格式为 **svg**。

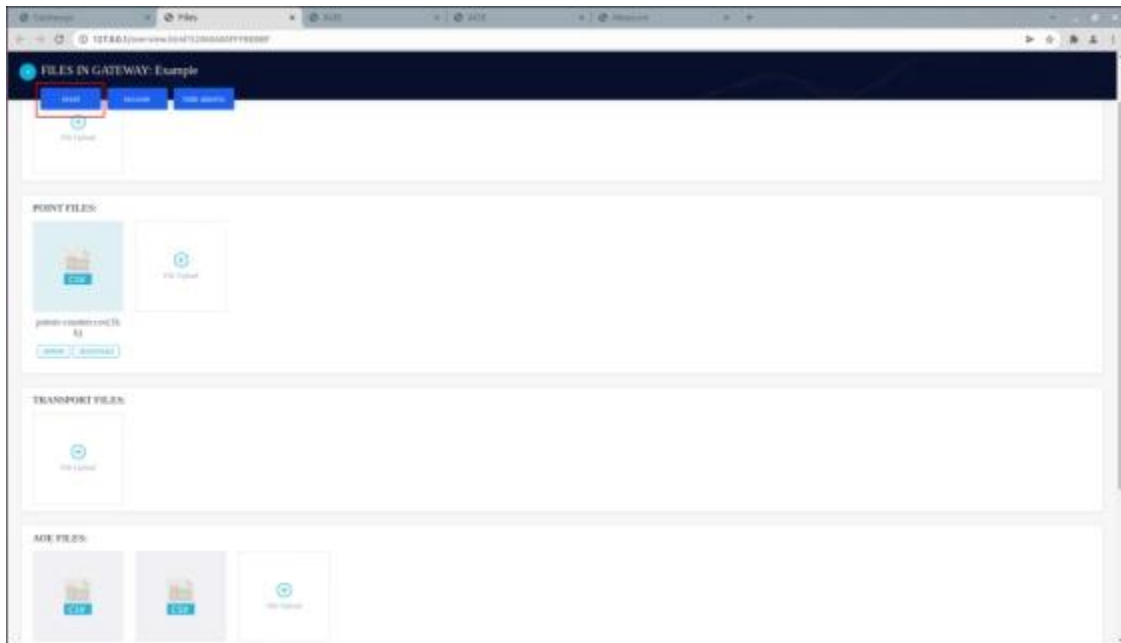
POINT FILES: 可以上传测点配置文件, 文件格式为 **csv**, 文件格式及配置方法参见测点文件配置。

TRANSPORT FILES: 可以上传通道配置文件, 文件格式为 **csv**, 文件格式及配置方法参见通道文件配置。

AOE FILES: 可以上传 AOE 策略配置文件, 文件格式为 **csv**, 文件格式即配置方法参见 AOE 策略文件配置。

4.5.3.3 运行告警区

当程序运行出现警告和错误时, 系统将在这里告警。



4.5.4 测点监控界面

按住 **ctrl** 并点击测点文件, 即可进入测点监控界面。

测点监控界面三个操作按钮。



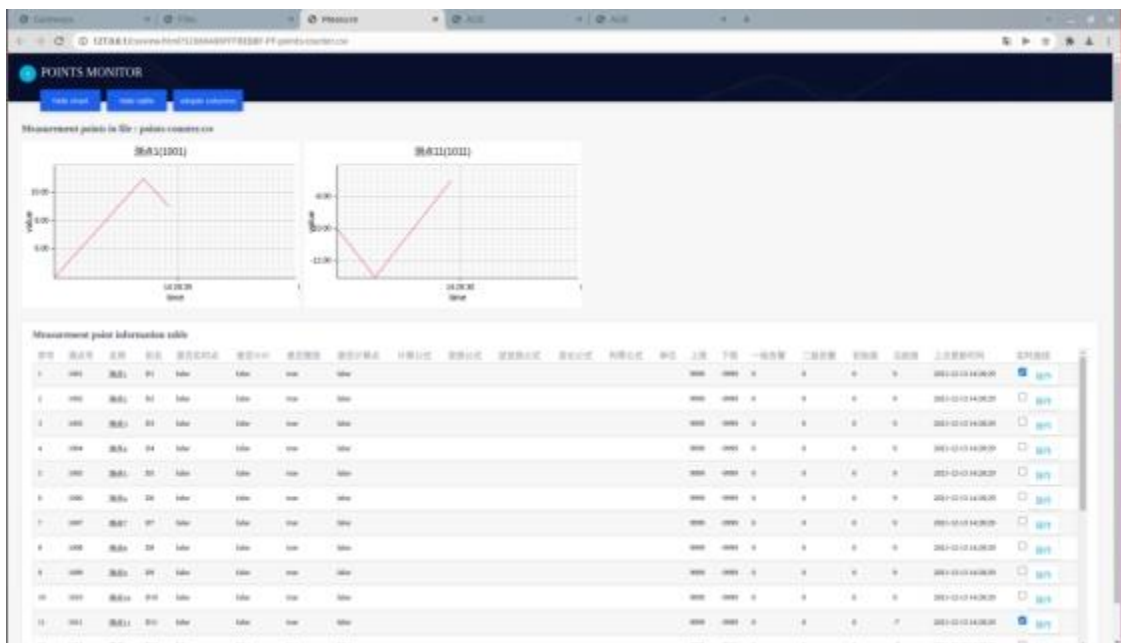
hide/show chart:该按钮用以隐藏/显示测点的实时曲线。

hide/show table:该按钮用以隐藏/显示测点信息表格。

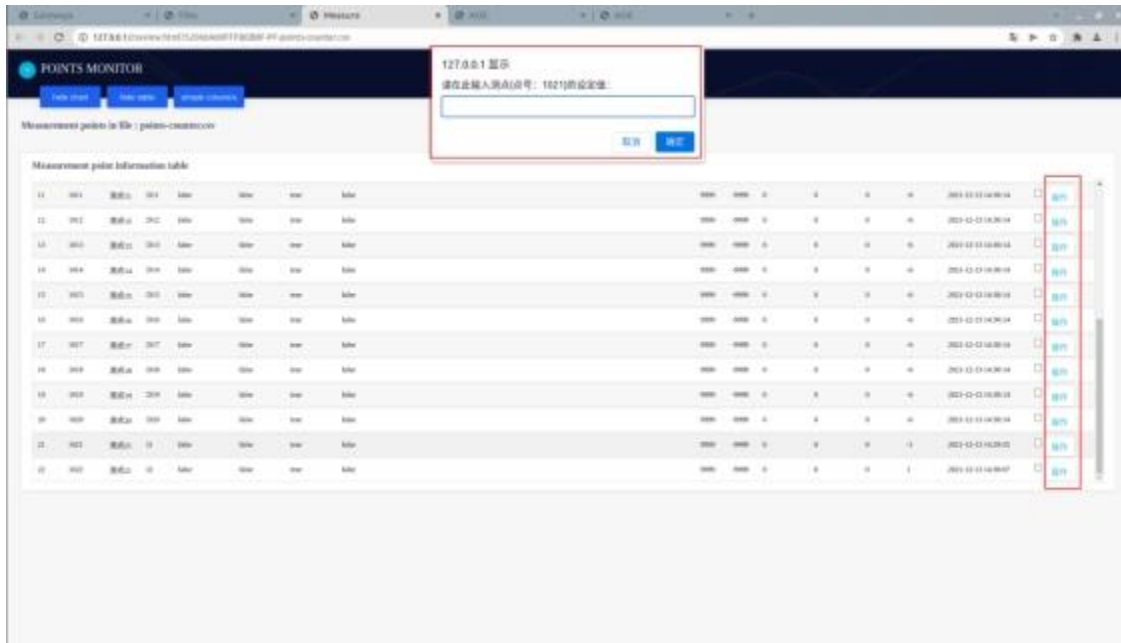
simple/full columns:该按钮用以显示简单/完整的测点信息表格。

在测点信息表格中，记录了测点当前数据，和该数据最近一次更新的时间。

勾选测点信息表格实时曲线的复选框，可以在测点信息表格上方显示所勾选测点的实时曲线图。



点击操作按钮可以写入测点数据。



4.5.5 AOE 监控界面

按住 ctrl 并点击 AOE 文件，即可进入 AOE 监控界面。在该界面中，记录着该 AOE 文件中所包含的 AOE 程序的执行状态、执行时间和执行结果。



每个 AOE 可由开始与停止两个按钮控制。

开始：开始按照 AOE 配置文件的配置参数等待程序触发运行。

停止：停止 AOE 程序运行，且该 AOE 不会再被触发，直到单击开始按钮。

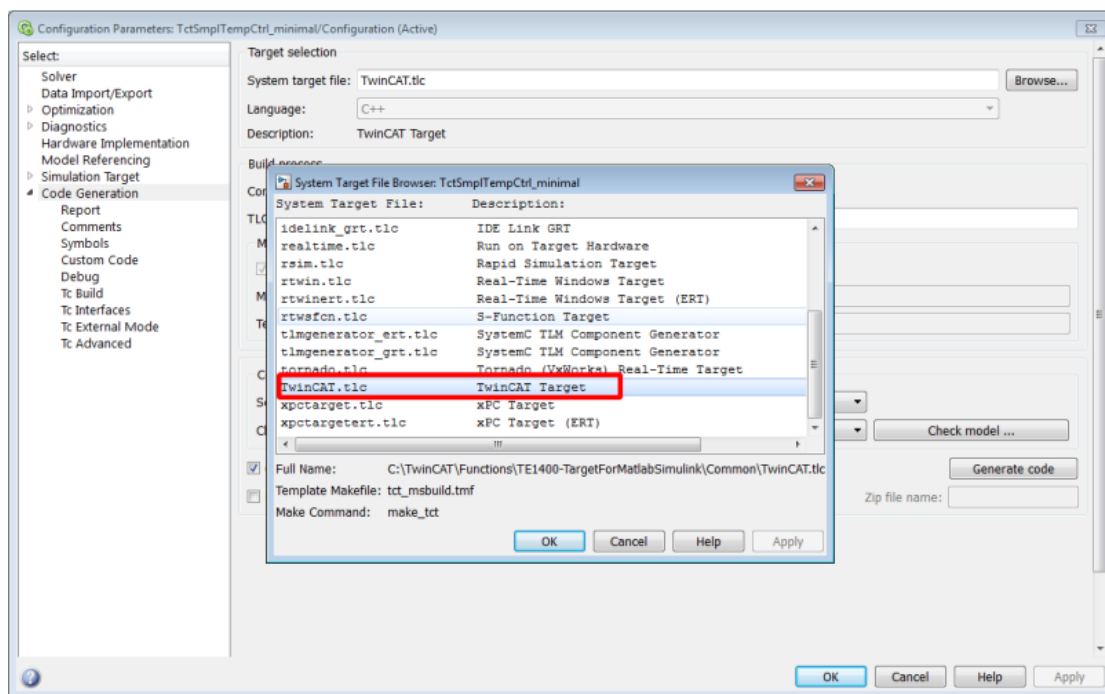
4.6 TwinCAT3 软件

4.6.1 从 MATLAB/Simulink 到 TwinCAT3

在传统的综合能源系统仿真部署中多采用在 MATLAB/Simulink 环境中进行，首先要搭建某一个系统组件的控制模型，之后通过反复的仿真和调整控制参数来获得理想的控制策略模型，之后再导出为工业控制软件能够识别的形式，如 C++代码等，以驱动 EtherCAT 或其他现场总线与实际硬件连接测试在实际硬件上控制策略的有效性。

这种调试方式在综合能源系统仿真的初期可以对综合能源系统内各部分的控制策略进行高效的仿真验证，但当综合能源系统运行仿真进入到系统级的运行调控阶段时，MATLAB/Simulink 仿真环境仿真速度慢、还需连接实际硬件进行二次调试等缺点越发凸显。

因此，极简物控开放式系统能源储能仿真平台配备工业 PC，支持 TwinCAT3 软件，内含 Simulink Module 的功能，通过综合能源分布式硬件加速仿真算法，允许用户事先在 MATLAB 中创建控制模型，包括控制代码和接口，然后将模块导入 TwinCAT3 实现直接的对硬件调试，其中 TE1400，即 Target for Simulink，可以将 Simulink 中搭建好的可视化控制模块通过 Simulink Coder 自动转换为 C/C++代码，在转换为 TwinCAT3 可以识别的控制对象。下图为使用 TE1400 将 Simulink 模型转换为 TwinCAT3 兼容的 PLC 项目时的配置界面。



通过 TE1400 创建的 TwinCAT 对象具有与其它 TwinCAT 对象相同的接口和属性，因此它们可以在 TwinCAT3 开发环境中得到充分应用，例如，扩展为一个带 PLC 源代码、调试和与现场总线站连接的完整项目。在 TwinCAT 开发环境中可以对 Simulink 框图进行可视化。除了 Simulink 外部模式外，嵌入在开发环境中的框图也可以用作参数调整、调试、信号和状态监测的控件。自动生成的模块既可以作为 TcCOM 对象，也可以作为 PLC 功能模块集成到 TwinCAT 解决方案中。集成式模块的作用通过 TwinCAT3 Runtime 中完整的 TwinCAT 项目被淡化，并且可以在实时环境中像所有其它对象一样被执行。

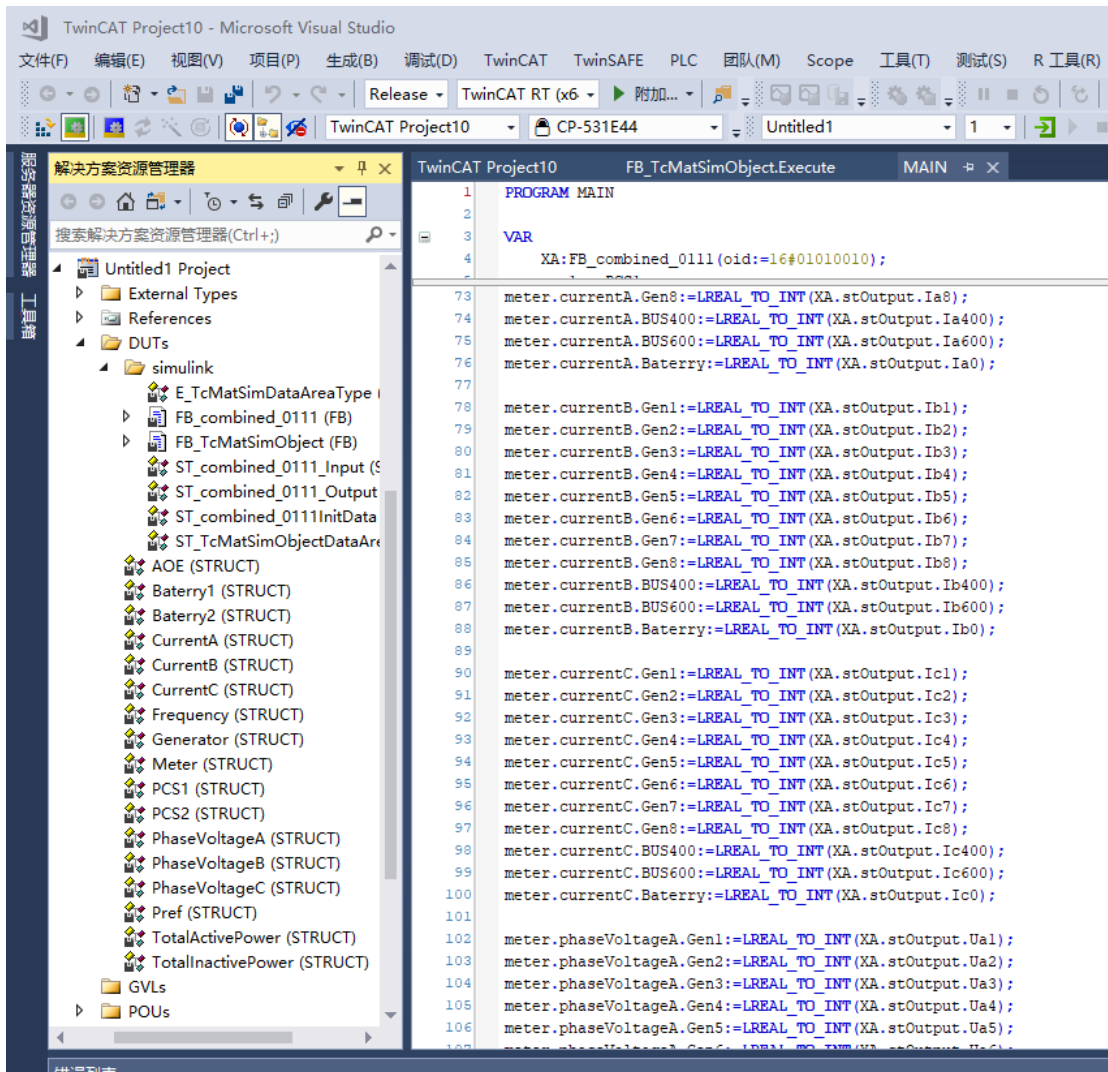
在实现了以上的转换后，复杂的控制对象就可以通过 TwinCAT 技术加速综合能源分布式硬件仿真。

4.6.2 TwinCAT3 仿真设置

基于 TwinCAT3 的硬件加速仿真利用了其 TcCOM Objects 功能将

使用 TE1400 编译之后的 Simulink 模型实例化，并通过简单的代码构造完整仿真系统。之后在 TwinCAT3 的仿真解散器（Solver）中设置合适的仿真步长以及仿真基准时间，使得下位仿真加速硬件采用的高性能工控机能够完全发挥其强大的算力，使得仿真系统运行速度大幅提升，仿真模型执行效率满足要求。

TwinCAT 3 软件的仿真设置分为两步，首先是在将 Simulink 模型编译成的 PLC 项目导入到 TwinCAT3 软件后，建立起一系列外部控制变量并通过简单的代码将其一一对应起来，TwinCAT3 软件主界面左侧为解决方案资源管理器，其中的 DUTs-Simulink 文件夹中的项为 Simulink 模型编译成的 PLC 项目导入到 TwinCAT3 软件后自动生成的，另外的项则是需要用户根据需要自行建立的。之后在 main 函数中采用形如下图右侧的代码将外部控制器的 I/O 口与 TwinCAT PLC 项目的 O/I 口分别对应起来。



格式如下:

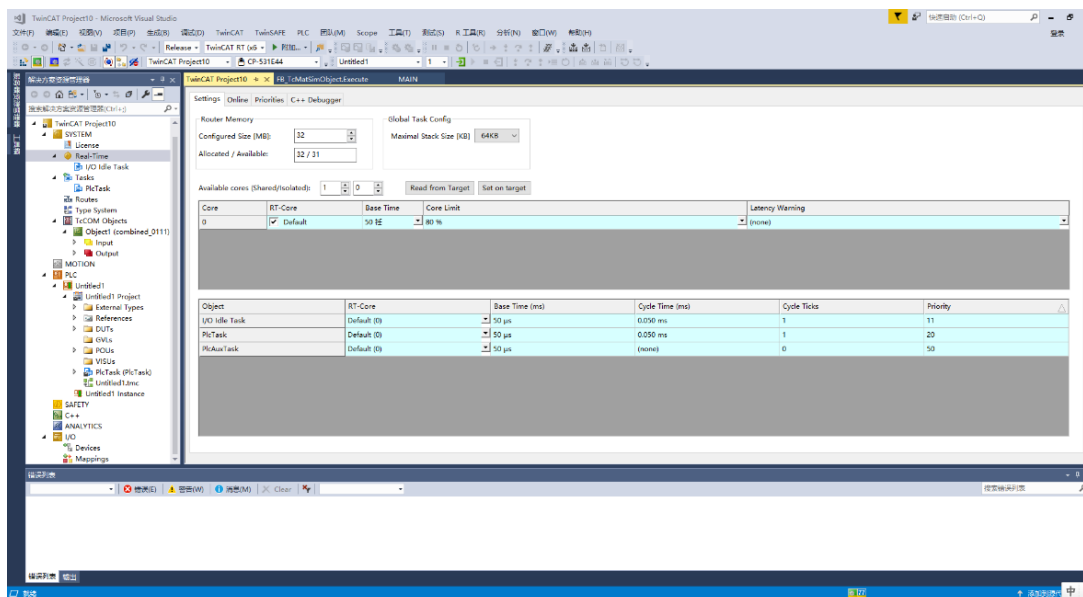
<输入变量>:=<输出变量数据类型>_TO_<输入变量数据类型>(<输出变量>);

或

<输出变量>:=<输入变量数据类型>_TO_<输出变量数据类型>(<输入变量>);

若输入变量与输出变量数据类型相同,那么就可以省略“<输出变量数据类型>_TO_<输入变量数据类型>”部分。

之后对 TwinCAT3 的仿真解散器 (Solver) 的仿真步长以及仿真基准时间进行设置应被选为与实际模型相符的定步长 (Fixed-step), 仿真基准时间设置为 $50\mu s$ 。具体的设置界面如图所示。



5 实验案例

5.1 实验环境

该程序可在以下环境运行：

1. Matlab R2019a + Simulink v9.3 及以上版本；
2. Matlab APP: Instrument Control Toolbox;
3. “微”能量管理器。

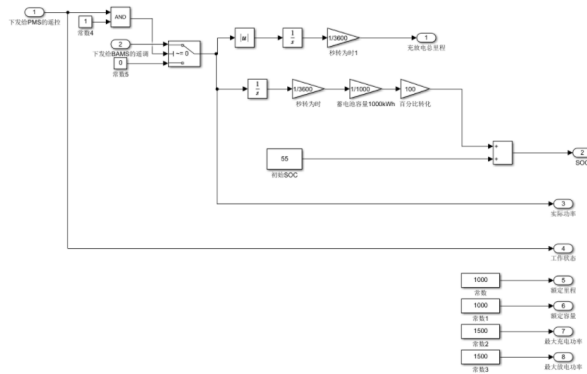
5.2 实验介绍

许多地区以火电机组做为主要调频资源，而储能的 AGC 调频效果远好于火电机组，引入相对少量的储能系统，将能够迅速并有效地解决区域电网调频资源不足的问题。

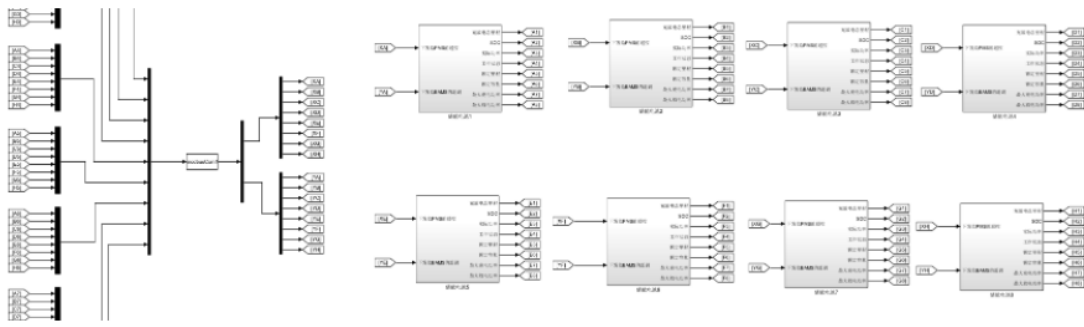
本案例中“微”能量管理器接收主控系统下发的调频指令，根据八个调频储能的 SOC、充放电总里程等条件，通过优化求解下发出力指令，控制八个储能的出力，从而达到储能辅助调频的目的。

5.3 实验模型

本实验模型需要自行搭建。根据储能电池模型，可搭建模型如下：



完成储能电池模型搭建后，对其进行封装，并引入 S-函数，在 S-函数中实现与“微”能量管理器通信功能，接收主控系统下发的调遥，完成储能功率分配并下发，如下图所示。



5.4 实验原理

本实验案例求解优化问题实现储能功率的分配。考虑到方差优化的复杂性，尽量避免将二次式作为优化目标，以 SOC 相对平均值的总偏差的绝对值和功率相对平均值的偏差绝对值作为目标函数：

$$\min w_1 \sum_{i=1}^8 |SOC_i - SOC_{mean}| |SOC_i^0 - SOC_{mean}| + w_2 \sum_{i=1}^8 e^{-|SOC_i - SOC_{mean}|} |P_i - P_{mean}|$$

式中，w1 表示 SOC 均衡的影响权重，w2 表示功率变动带来的代价权重；SOCmean 为各 SOC 均值；SOC0i 为当前电池的 SOC

值； P_i 表示当前电池的功率值； P_{mean} 表示当前电池功率的平均值； SOC_i 表示在功率分发策略所计算出的功率工作下，下一时刻第 i 个电池的 SOC 值，具体计算式如下：

$$SOC_i = SOC_i^0 + \frac{x_i \Delta t}{E_{act,i}}$$

其中 $E_{act,i}$ 表示电池 i 的实际容量。

5.5 实验步骤

5.5.1 编写配置文件

文件配置分为三个部分：测点配置、通信通道配置和 AOE 配置。测点配置用于配置控制器输入和输出信号点的参数，本案例中需要的信号是八个储能的启停指令、充放电总里程、SOC、输出功率、工作状态、输出功率指令、额定里程、额定容量、最大充电功率、最大放电功率，以及系统收到的是否调频指令与调频功率指令。通信通道的配置是为了将控制器与 simulink 仿真模型建立 modbus 通信，实现仿真模型与控制器的信息交互，而 AOE 配置是建立 AOE 模型网络来实现所需的控制逻辑和功能。

5.5.1.1 测点配置

测点配置表格的内容包含了测点的点号、名称、别名、是否离散、是否是计算点以及默认值等。点号是程序里给这个测点的编号，可以自己定义，使用时，测点与通信通道的点号一一对应完成对接。别名是英文字母与下划线组合使用，可以当做变量名用于 AOE 表格里表达式的书写，使之可以不用点号来表达，使得表达式

的意思表达得更加清楚。测点配置如下图所示：测点配置表格的内容包含了测点的点号、名称、别名、是否离散、是否是计算点以及默认值等。点号是程序里给这个测点的编号，可以自己定义，使用时，测点与通信通道的点号一一对应完成对接。别名是英文字母与下划线组合使用，可以当做变量名用于 AOE 表格里表达式的书写，使之可以不用点号来表达，使得表达式的意思表达得更加清楚。部分测点配置如下图所示：

序号	点号	名称	别名	是否离散	是否是计算点	计算公式	变换公式	逆变换公式	变化公式	判零公式	单位	上限值	下限值	最大变化	最小变化	是否实时点	是否SOE	默认值	备注
1	1001	测点1	EMS_MOD	TRUE	FALSE							0	0	0	0	FALSE	FALSE	0	0 主控系统下发的遥控-->EMS模式
2	1002	测点2	GEN_TOTA	FALSE	FALSE							0	0	0	0	FALSE	FALSE	1	1 主控系统下发的遥控-->系统有功指令
3	1003	测点3	PCS1_CTRL	TRUE	FALSE							0	0	0	0	FALSE	FALSE	0	0 下发给PMS的遥控
4	1004	测点4	PCS2_CTRL	TRUE	FALSE							0	0	0	0	FALSE	FALSE	0	0 下发给PMS的遥控
5	1005	测点5	PCS3_CTRL	TRUE	FALSE							0	0	0	0	FALSE	FALSE	0	0 下发给PMS的遥控
6	1006	测点6	PCS4_CTRL	TRUE	FALSE							0	0	0	0	FALSE	FALSE	0	0 下发给PMS的遥控
7	1007	测点7	PCS5_CTRL	TRUE	FALSE							0	0	0	0	FALSE	FALSE	0	0 下发给PMS的遥控
8	1008	测点8	PCS6_CTRL	TRUE	FALSE							0	0	0	0	FALSE	FALSE	0	0 下发给PMS的遥控
9	1009	测点9	PCS7_CTRL	TRUE	FALSE							0	0	0	0	FALSE	FALSE	0	0 下发给PMS的遥控
10	1010	测点10	PCS8_CTRL	TRUE	FALSE							0	0	0	0	FALSE	FALSE	0	0 下发给PMS的遥控
11	1011	测点11	BAMS1_MI	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 充放总里程
12	1012	测点12	BAMS2_MI	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 充放总里程
13	1013	测点13	BAMS3_MI	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 充放总里程
14	1014	测点14	BAMS4_MI	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 充放总里程
15	1015	测点15	BAMS5_MI	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 充放总里程
16	1016	测点16	BAMS6_MI	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 充放总里程
17	1017	测点17	BAMS7_MI	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 充放总里程
18	1018	测点18	BAMS8_MI	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 充放总里程
19	1019	测点19	BAMS1_SC	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 soc
20	1020	测点20	BAMS2_SC	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 soc
21	1021	测点21	BAMS3_SC	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 soc
22	1022	测点22	BAMS4_SC	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 soc
23	1023	测点23	BAMS5_SC	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 soc
24	1024	测点24	BAMS6_SC	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 soc
25	1025	测点25	BAMS7_SC	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 soc
26	1026	测点26	BAMS8_SC	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 soc
27	1027	测点27	BAMS1_PJ	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 功率
28	1028	测点28	BAMS2_PJ	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 功率
29	1029	测点29	BAMS3_PJ	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 功率
30	1030	测点30	BAMS4_PJ	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 功率
31	1031	测点31	BAMS5_PJ	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 功率
32	1032	测点32	BAMS6_PJ	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 功率
33	1033	测点33	BAMS7_PJ	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 功率
34	1034	测点34	BAMS8_PJ	FALSE	FALSE							0	0	0	0	FALSE	FALSE	0	0 功率
35	1035	测点35	BAMS1_WI	TRUE	FALSE							0	0	0	0	FALSE	FALSE	0	0 工作状态
36	1036	测点36	BAMS2_WI	TRUE	FALSE							0	0	0	0	FALSE	FALSE	0	0 工作状态
37	1037	测点37	BAMS3_WI	TRUE	FALSE							0	0	0	0	FALSE	FALSE	0	0 工作状态
38	1038	测点38	BAMS4_WI	TRUE	FALSE							0	0	0	0	FALSE	FALSE	0	0 工作状态
39	1039	测点39	BAMS5_WI	TRUE	FALSE							0	0	0	0	FALSE	FALSE	0	0 工作状态

本程序共需要 82 个测点，具体包括八个储能的启停指令、充放电总里程、SOC、输出功率、工作状态、输出功率指令、额定里程、额定容量、最大充电功率、最大放电功率，以及系统收到的是否调频指令与调频功率指令。

5.5.1.2 通信通道配置

本案例中，控制器作为主站，具体的通道配置如下：

通道名称	server测试通道	连接名称	测试通道1_序号	寄存器类	起始地址	数据类型	新请求标志	轮询周期	点号	
连接个数	1	测点个数	82	1	HOLDING	1	TwoByteIn	FALSE	2000	1001
服务端IP	502	客户端IP	127.0.0.1	2	HOLDING	2	EightByteF	FALSE	2000	1002
		客户端端口	9999	3	HOLDING	6	TwoByteIn	FALSE	2000	1003
		slave id	1	4	HOLDING	7	TwoByteIn	FALSE	2000	1004
		通信协议	XA	5	HOLDING	8	TwoByteIn	FALSE	2000	1005
		一次读寄存器	125	6	HOLDING	9	TwoByteIn	FALSE	2000	1006
		一次读开关	2000	7	HOLDING	10	TwoByteIn	FALSE	2000	1007
		一次写寄存器	120	8	HOLDING	11	TwoByteIn	FALSE	2000	1008
		一次写开关	1968	9	HOLDING	12	TwoByteIn	FALSE	2000	1009
		轮询周期(r)	5000	10	HOLDING	13	TwoByteIn	FALSE	2000	1010
		超时(ms)	1000	11	HOLDING	17	EightByteF	FALSE	2000	1011
		通道状态源	4999	12	HOLDING	21	EightByteF	FALSE	2000	1012
		数据类型	说明	13	HOLDING	25	EightByteF	FALSE	2000	1013
		Binary	bool	14	HOLDING	29	EightByteF	FALSE	2000	1014
		OneByteIn	byte	15	HOLDING	33	EightByteF	FALSE	2000	1015
		OneByteIn	byte	16	HOLDING	37	EightByteF	FALSE	2000	1016
		TwoByteIn	u16	17	HOLDING	41	EightByteF	FALSE	2000	1017
		TwoByteIn	i16	18	HOLDING	45	EightByteF	FALSE	2000	1018
		TwoByteIn	SignedSw	19	HOLDING	49	EightByteF	FALSE	2000	1019
		TwoByteBcd		20	HOLDING	53	EightByteF	FALSE	2000	1020
		TwoByteIn	UnsignedS	21	HOLDING	57	EightByteF	FALSE	2000	1021
		FourByteIn	Unsigned	22	HOLDING	61	EightByteF	FALSE	2000	1022
		FourByteIn	u32	23	HOLDING	65	EightByteF	FALSE	2000	1023
		FourByteIn	UnsignedS	24	HOLDING	69	EightByteF	FALSE	2000	1024
		FourByteIn	SignedSw	25	HOLDING	73	EightByteF	FALSE	2000	1025
		FourByteIn	UnsignedS	26	HOLDING	77	EightByteF	FALSE	2000	1026
		FourByteIn	SignedSw	27	HOLDING	81	EightByteF	FALSE	2000	1027
		FourByteFloat		28	HOLDING	85	EightByteF	FALSE	2000	1028
		FourByteFloat	Swappe	29	HOLDING	89	EightByteF	FALSE	2000	1029
		FourByteBcd		30	HOLDING	93	EightByteF	FALSE	2000	1030
		FourByteBcd	Swapped	31	HOLDING	97	EightByteF	FALSE	2000	1031
		FourByteMod10k		32	HOLDING	101	EightByteF	FALSE	2000	1032
		FourByteMod10k	Swap	33	HOLDING	105	EightByteF	FALSE	2000	1033
		SixByteMod10k		34	HOLDING	109	EightByteF	FALSE	2000	1034
		SixByteMod10k	Swapp	35	HOLDING	110	TwoByteIn	FALSE	2000	1035
		EightByteIn	Unsigned	36	HOLDING	111	TwoByteIn	FALSE	2000	1036
		EightByteIn	Signed	37	HOLDING	112	TwoByteIn	FALSE	2000	1037
		EightByteIn	Unsigned	38	HOLDING	113	TwoByteIn	FALSE	2000	1038

本案例中总共用到两种数据类型，TwoByteIntUnsigned 对应开关量，占据一位寄存器；EightByteFloat 对应连续或者离散的数据变量，占据 4 位寄存器。根据数据类型完成对起始地址的分配，即可完成通道文件的配置。

5.5.1.3 AOE 配置

a.AOE 网络声明

AOE 网络声明部分，定义了表格之中的 AOE 网络个数和整体信息以及一些变量的初始值。其中触发条件分为三类：SimpleRepeat（定时触发，需要自行设定定时时间）、TimeDriven（时间驱动）、EventDriven（事件驱动）；变量的定义方式为：<变量名 1>:<初值 1>;<变量名 2>:<初值 2>;...

本案例中的 AOE 网络声明如下所示：

AOE ID	是否启用	名称	触发条件	触发条件参数	变量初始值
33	TRUE	AGC辅助调频	Event_Drive		deadband:100;SOCref:0.55;dt:0.25;last_cmd:100;w:0.0000005;max_power:1500

本实验中需要定义多个变量初始值。具体包括：deadband:

SOC 死区; SOCref: SOC 参考值; dt: 每步步长; w: 系数 w2;
 max_power: 充放电功率上下限。

b.AOE 节点声明

节点类型分为 Switch 和 Condition 两类，其中 Switch 节点为二分支逻辑判断，触发逻辑为：表达式为真时，该节点序号为 1 的边进行动作；表达式为假时，序号为 2 的边进行动作。当 Switch 节点为 AOE 网入度为 0 的节点（首节点）时，Switch 节点直接触发，若超时时间内表达式为真，则执行动作 1，为假或者发生超时事件执行动作 2；Condition 节点的表达式为事件触发条件，超时时间内表达式为真，则节点事件发生，该节点出发的所有动作并行执行。

本案例的节点声明如下：

AOE ID	节点ID	名称	节点类型	超时时间(ms)	表达式
33	1	开始节点	Condition	1000	EMS_MODE_AGC_POINT == 1 && GEN_TOTAL_P_POINT != last_cmd
33	2	中间量计算完成	Condition	15000	1
33	3	优化求解结束	Switch	1000	flag == 1
33	4	存在最优解	Condition	1000	1
33	5	不存在最优解	Condition	1000	1

本案例中一共有 5 个节点，第 1 个节点是 simulink 通信事件节点，为事件触发类型，当 EMS_MODE_AGC_POINT>0.5 时，该节点被触发，进入运算流程。

第 2 个节点表示中间变量计算完成的事件，当考虑 SOH 因总充电里程变化而减小时，计算并更新当前各储能的 SOH 以及储能运行寿命损耗导致的实际容量减少等，当计算动作完成后，节点 2 触发，开始进行优化计算。

第 3 个节点表示优化计算完成的事件，在控制器内置的优化求解方法里，当最优解不存在时所有优化变量的值均为 0；而有最优

解存在时，由于目标函数的设计，**flag** 的值等于 1，因此可以通过设置一位二进制变量 **flag** 来作为优化问题是否有解的判断条件。该节点类型为 **Switch**，节点事件触发时，节点动作 1 被执行，否则发生超时之后，节点动作 2 被执行

第 4 个节点表示优化问题已得到最优解的事件，该事件发生说明最优解被找到并下发，**AOE** 策略结束。

第 5 个节点表示优化问题没有找到最优解，这种情况发生说明发生以下两种情况：1.EMS 下发的 **AGC** 指令不合理；2.储能可调度容量太小。

c.AOE 边声明

边需要声明首尾节点，格式为：<尾节点 ID>;<首节点 ID>。边的方向为：尾节点→首节点。边的动作类型有：**None**、**SetPoints**、**Solve**、**MILP**、**SimpleMilp**、**NLP**。其中，**SetPoints** 除了对测点设值之外，还可以进行变量定义和计算，动作参数分为数字量和模拟量；**Solve** 为求解方程 $Ax=b$ ；**Milp** 为混合整数线性规划问题求解；**Nlp** 为非线性规划问题求解。

本案例中的 **AOE** 边声明如下图所示：本案例中的 **AOE** 边声明如下图所示：

AOE ID	首尾节点	动作名称	失败模式	动作类型	动作参数
33	1,2	计算中间量	Default	Set_Points	QRate 10000;
33	2,3	优化求解	Default	MILP	$\min(\text{flag} + \text{abs}(\text{BAMS1_SOC_POINT} - (\text{BAMS1_SOC_POINT} + \text{BAMS2_SOC_POINT} + \text{BAMS3_SOC_POINT} + \text{BAMS4_SOC_POINT} + \text{BAMS5_SOC_POINT} + \text{BAMS6_SOC_POINT} + \text{BAMS7_SOC_POINT} + \text{BAMS8_SOC_POINT})/\theta) * \alpha1 + w * \exp(-\text{abs}(\text{BAMS1_SOC_POINT} - (\text{BAMS1_SOC_POINT} + \text{BAMS2_SOC_POINT} + \text{BAMS3_SOC_POINT} + \text{BAMS4_SOC_POINT} + \text{BAMS5_SOC_POINT} + \text{BAMS6_SOC_POINT} + \text{BAMS7_SOC_POINT} + \text{BAMS8_SOC_POINT})/\theta)) * \beta1 + \text{abs}(\text{BAMS2_SOC_POINT} - (\text{BAMS1_SOC_POINT} + \text{BAMS2_SOC_POINT} + \text{BAMS3_SOC_POINT} + \text{BAMS4_SOC_POINT} + \text{BAMS5_SOC_POINT} + \text{BAMS6_SOC_POINT} + \text{BAMS7_SOC_POINT} + \text{BAMS8_SOC_POINT})/\theta) * \alpha2 + w * \exp(-\text{abs}(\text{BAMS2_SOC_POINT} - (\text{BAMS1_SOC_POINT} + \text{BAMS2_SOC_POINT} + \text{BAMS3_SOC_POINT} + \text{BAMS4_SOC_POINT} + \text{BAMS5_SOC_POINT} + \text{BAMS6_SOC_POINT} + \text{BAMS7_SOC_POINT} + \text{BAMS8_SOC_POINT})/\theta)) * \beta2 + \text{abs}(\text{BAMS3_SOC_POINT} - (\text{BAMS1_SOC_POINT} + \text{BAMS2_SOC_POINT} + \text{BAMS3_SOC_POINT} + \text{BAMS4_SOC_POINT} + \text{BAMS5_SOC_POINT} + \text{BAMS6_SOC_POINT} + \text{BAMS7_SOC_POINT} + \text{BAMS8_SOC_POINT})/\theta) * \alpha3 + w * \exp(-\text{abs}(\text{BAMS3_SOC_POINT} - (\text{BAMS1_SOC_POINT} + \text{BAMS2_SOC_POINT} + \text{BAMS3_SOC_POINT} + \text{BAMS4_SOC_POINT} + \text{BAMS5_SOC_POINT} + \text{BAMS6_SOC_POINT} + \text{BAMS7_SOC_POINT} + \text{BAMS8_SOC_POINT})/\theta)) * \beta3 + \text{abs}(\text{BAMS4_SOC_POINT} - (\text{BAMS1_SOC_POINT} + \text{BAMS2_SOC_POINT} + \text{BAMS3_SOC_POINT} + \text{BAMS4_SOC_POINT} + \text{BAMS5_SOC_POINT} + \text{BAMS6_SOC_POINT} + \text{BAMS7_SOC_POINT} + \text{BAMS8_SOC_POINT})/\theta) * \alpha4 + w * \exp(-\text{abs}(\text{BAMS4_SOC_POINT} - (\text{BAMS1_SOC_POINT} + \text{BAMS2_SOC_POINT} + \text{BAMS3_SOC_POINT} + \text{BAMS4_SOC_POINT} + \text{BAMS5_SOC_POINT} + \text{BAMS6_SOC_POINT} + \text{BAMS7_SOC_POINT} + \text{BAMS8_SOC_POINT})/\theta)) * \beta4 + \text{abs}(\text{BAMS5_SOC_POINT} - (\text{BAMS1_SOC_POINT} + \text{BAMS2_SOC_POINT} + \text{BAMS3_SOC_POINT} + \text{BAMS4_SOC_POINT} + \text{BAMS5_SOC_POINT} + \text{BAMS6_SOC_POINT} + \text{BAMS7_SOC_POINT} + \text{BAMS8_SOC_POINT})/\theta)) * \beta5 + \text{abs}(\text{BAMS5_SOC_POINT} - (\text{BAMS1_SOC_POINT} + \text{BAMS2_SOC_POINT} + \text{BAMS3_SOC_POINT} + \text{BAMS4_SOC_POINT} + \text{BAMS5_SOC_POINT} + \text{BAMS6_SOC_POINT} + \text{BAMS7_SOC_POINT} + \text{BAMS8_SOC_POINT})/\theta)) * \beta5$
33	3,4	下发优化策略	Default	Set_Points	BAMS1_CTRL_POINT(1)=VL
33	3,5	下发默认策略	Default	Set_Points	BAMS2_CTRL_POINT(5)=TOTAL_P_POWER

本实验中 AOE 主要进行的为优化求解功能。

5.5.2 配置文件存储

在 excel 表格中编写完成之后，将表格另存为.csv 格式（纯文本形式存储表格数据）文件供“微”能量管理器使用。

5.5.3 编写通信程序

modbusCom.m 是本实验模型控制器的源文件，建立与智能微电网能量管理控制器通信的作用。

下面的代码表示的是输入和输出值的个数，本实验中的输入有 66 个，包括八个储能的充放电总里程、SOC、输出功率、工作状态、额定里程、额定容量、最大充电功率、最大放电功率，以及系统收到的是否调频指令与调频功率指令。输出有 16 个，为八个储能的启停指令与输出功率指令。

```
sizes.NumContStates = 0;
```

```
sizes.NumDiscStates = 0;
```

```
sizes.NumOutputs = 16;
```

```
sizes.NumInputs = 66;
```

```
sizes.DirFeedthrough = 1;
```

```
sizes.NumSampleTimes = 1;
```

下式表示的是定步长仿真的仿真步长 Modbus 通信模块的在 Simulink 仿真模型中的采样时间为 0.02s。下图表示的是定步长仿真的仿真步长 Modbus 通信模块的在 Simulink 仿真模型中的采样时间为 0.02s。

```
ts = [0.02,0];
```

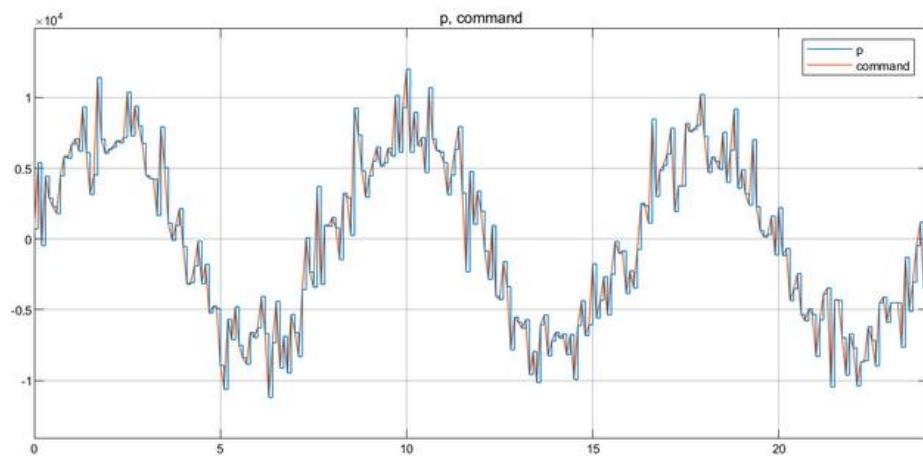
下式表示建立与控制器的通信。

```
m = modbus('tcpip', '192.168.137.170', 502);
```

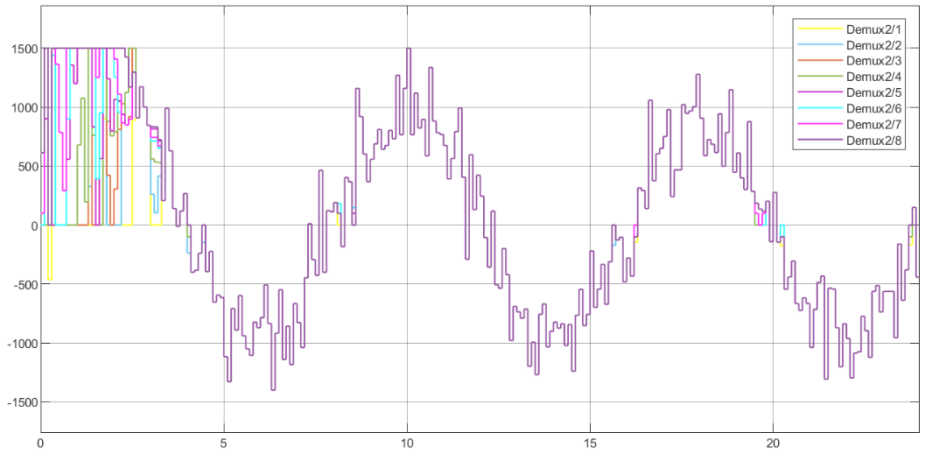
5.5.4 仿真运行结果

先运行智能微电网能量管理控制器，再运行 Simulink 模型，进行仿真，得到的结果如下所示：

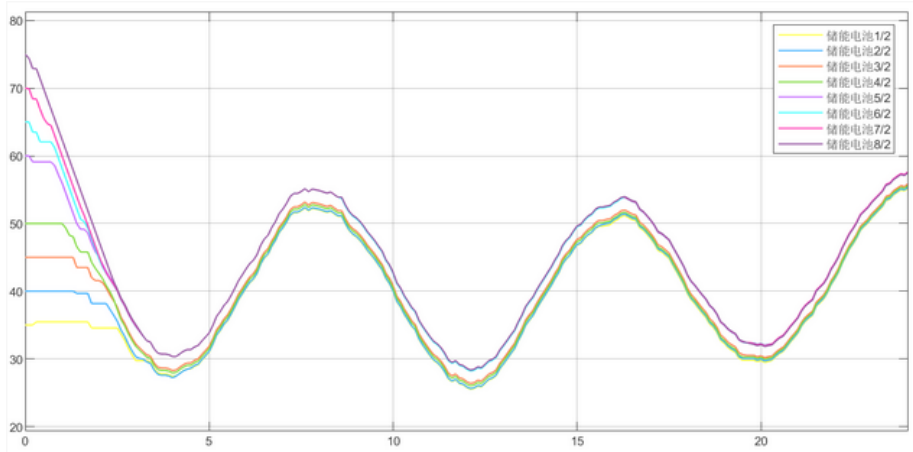
储能总功率与 AGC 指令功率



各储能功率曲线



各储能 SOC 曲线



5.6 实验总结

由该仿真实验可以看出，本实验平台并不需要繁复的程序代码编写，只需要编写测点配置、通信通道配置和 AOE 配置文件就可以实现通信和控制功能，便于高校开展仿真实验教学使用。

6 设备参数

6.1 智能开关柜

- 主开关 630A
- 4 个分路: 250A*4
- 每个回路均配备智能计量表。

6.2 工业 PC

- 硬盘上限 SSD 1T
- 内存上限 DDR4 16GB
- 配置 CPU 为 processor 9th generation Intel® Core™ i7-9700E8 核,
睿频上限 4.40GHz
- 支持 Microsoft Windows 10 IoT Enterprise 2019 LTSC, 64 bit,
English 操作系统
- 支持 Modbus 通信模块
- 支持 Simulink 对象运行模块

6.3 TwinCAT 软件

- 含 Licence-Key-USB-Stick for TwinCAT 3.1
- 含 MATLAB/Simulink 中的 TwinCAT 转换对象, 用于生成
TwinCAT 3 模块
- 含 MATLAB/Simulink 和 TwinCAT 3 通信接口
- 含 Modbus 通信协议模块

6.4 智能微电网能量管理控制器

- 支持 Modbus TCP、Modbus RTU、UART、IEC 104、DL/T 645-2007、MQTT、EtherCAT 电力协议接入及转换
- 支持实时监控、优化调度、智能控制、异常告警、数据报表能量管理功能模块
- 具备 RJ45、WIFI、RS485、RS232 接口。

6.5 智能微电网能量管理控制软件

- 支持实时监控功能，监测分析设备状态，采集存储实时信息；
- 支持优化调度功能，高效分配能源资产，实现储能辅助调频、储能削峰填谷、过零检测、线损分析、直流法潮流计算、交流法潮流计算、低压继电保护，以及基于事件-边活动网（AOE 网）控制组态的其他自定义优化调度策略。
- 支持智能控制功能，实现标准控制组态，支持逻辑运算、矩阵计算、线性方程组求解、非线性方程求解、混合整数线性规划（稀疏）、混合整数线性规划（稠密）、非线性规划。
- 支持异常告警功能，实现失败响应、系统告警。
- 支持数据报表功能，实现数据日志导出、自定义报表。

6.6 可编程逻辑控制器

- 支持 Modbus 通信模块；
- 支持 Microsoft Windows CE 系统；
- 配置 CPU Intel® IXP420，主频 533MHz
- 闪存 32MB

- 主内不低于 128MB RAM
- 具备 2*RJ45、RS232 接口

6.7 端子

- 针对逻辑量支持 24V 输入输出
- 针对模拟量支持 16 位 4-20mA 电流差分输入输出
- 针对模拟量支持 16 位 4 ~ 20mV 电压差分输入输出