

# 14 配方

## 14.1 概述

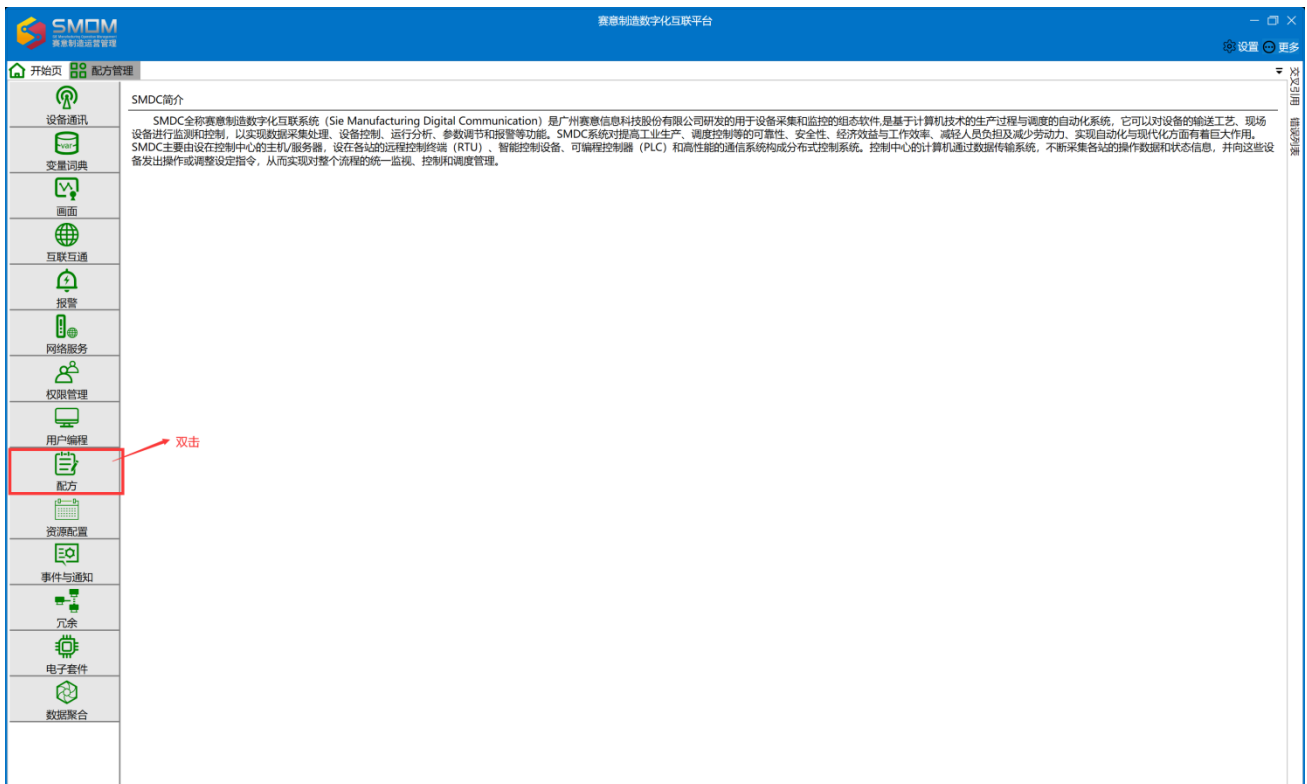
用户可在该模块辅助企业实现生产订单与工艺参数（工艺程序）绑定，实现混单生产或者自动切换，并实现订单追溯等目的，在 PCB、电子行业、机加工等行业广泛使用，SMDC 配方管理支持配方值表、配方文件、配方号三种方式下达，并支持配方二次确认，形成业务闭环。

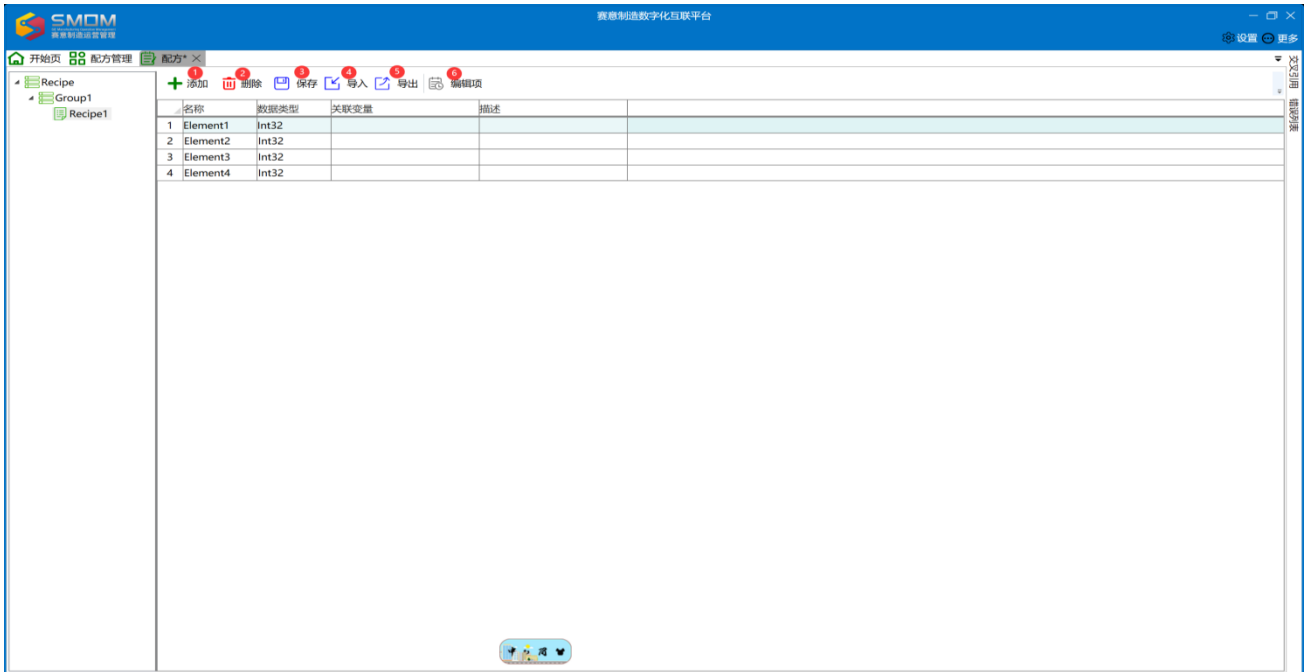
配方实际作用就是批量修改某一个设备的变量。

## 14.2 界面简介

### 14.2.1 主界面简介

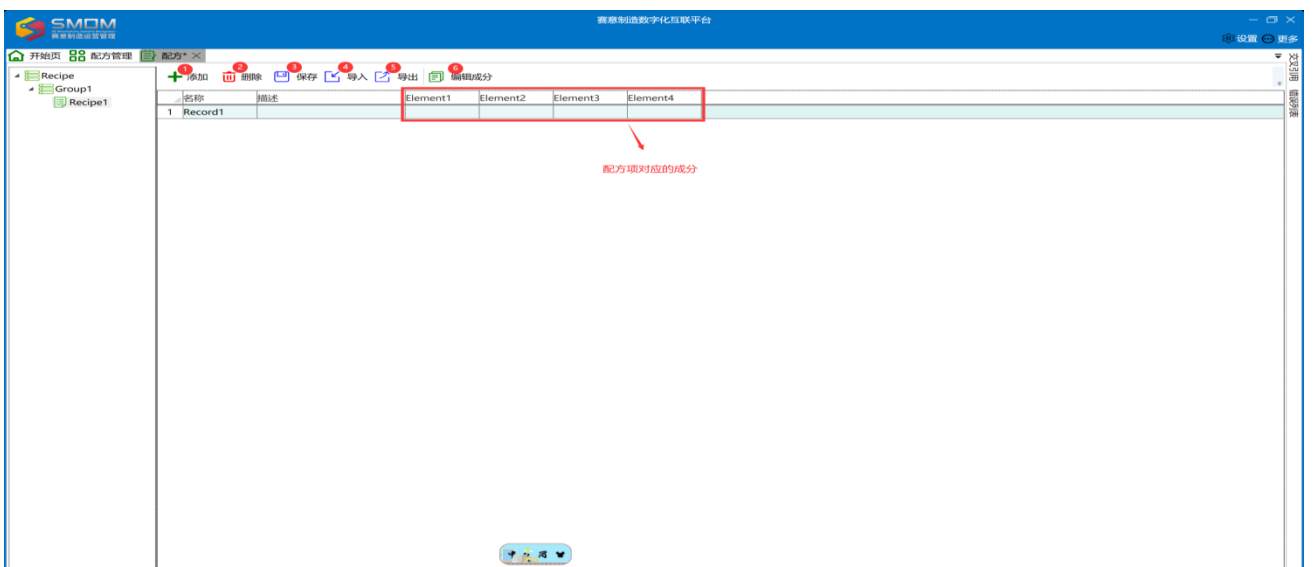
打开 SMDC 主界面，双击配方模组进入配置主界面：





- ① 添加：添加一条成分；
- ② 删除：添加选中的成分行；
- ③ 保存：保存编辑后的数据，数据保存在工程文件夹的 Recipe 文件夹下；
- ④ 导入：可以将符合格式的.xls 表格导入测配置界面，简化配置；
- ⑤ 导出：可以将界面配置的配方信息导出成.xls 表格；
- ⑥ 编辑项：点击进入编辑配方项界面；
- ⑦ 过期时间（天）：存放数据的时长；

## 14.2.2 编辑项界面简介

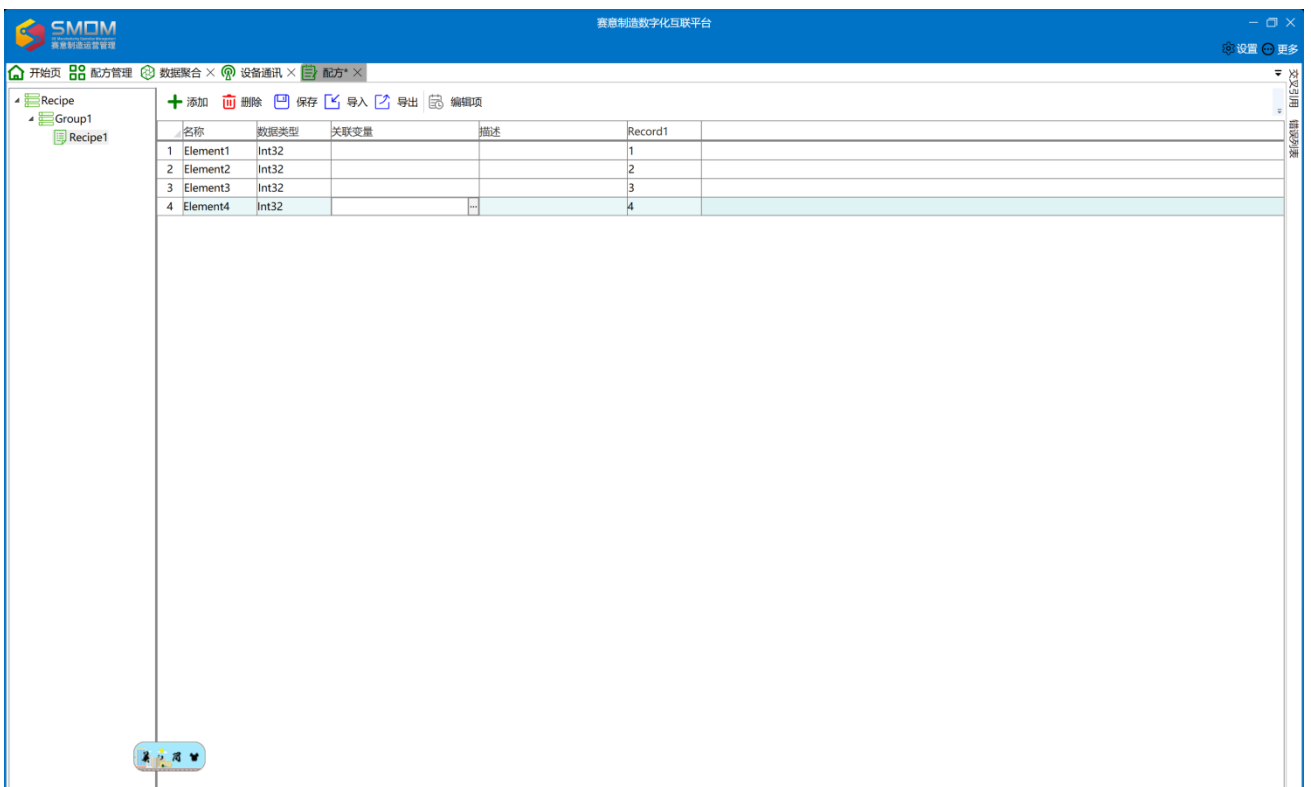


- ① 添加：添加一条配方项；
- ② 删除：添加选中的配方项；
- ③ 保存：保存编辑后的数据，数据保存在工程文件夹的 **Recipe** 文件夹下；
- ④ 导入：可以将符合格式的.xls 表格导入测配置界面，简化配置；
- ⑤ 导出：可以将界面配置的配方信息导出成.xls 表格；
- ⑥ 编辑项：点击进入编辑成分界面；
- ⑦ 过期时间（天）：存放数据的时长；

## 14.3 使用简介

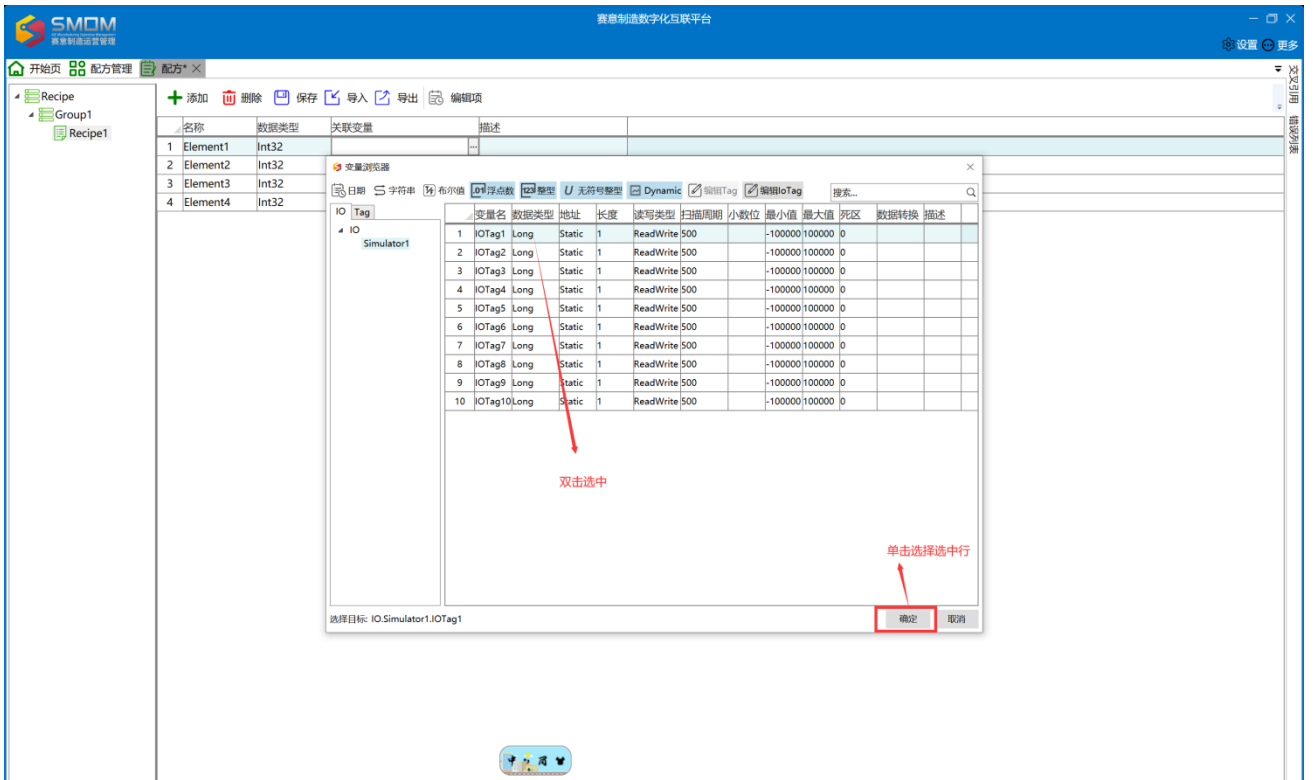
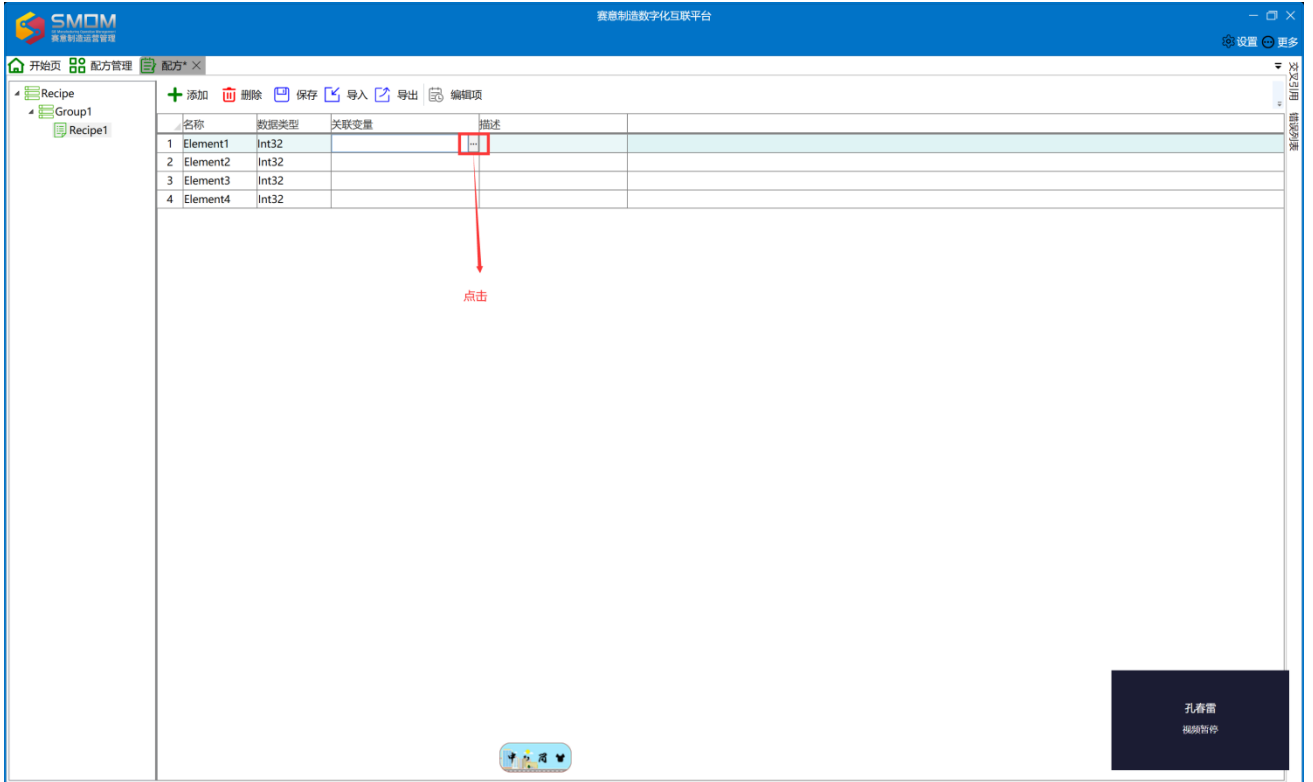
Step1:

进入配方主界面,并在 **Group1** 组下新建一个配方 **Recipe1**,然后点击添加按钮给配方 **Recipe1** 增加组成元素,即成份:

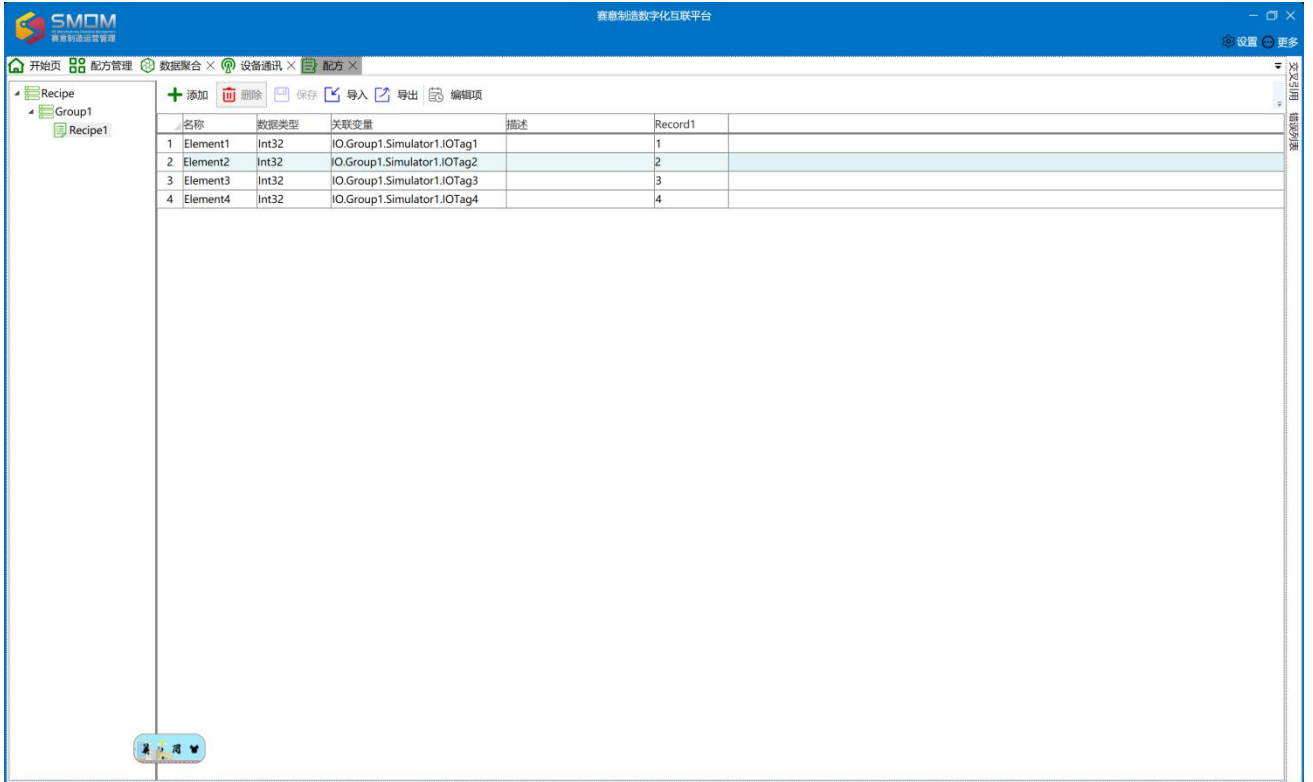


Step2:

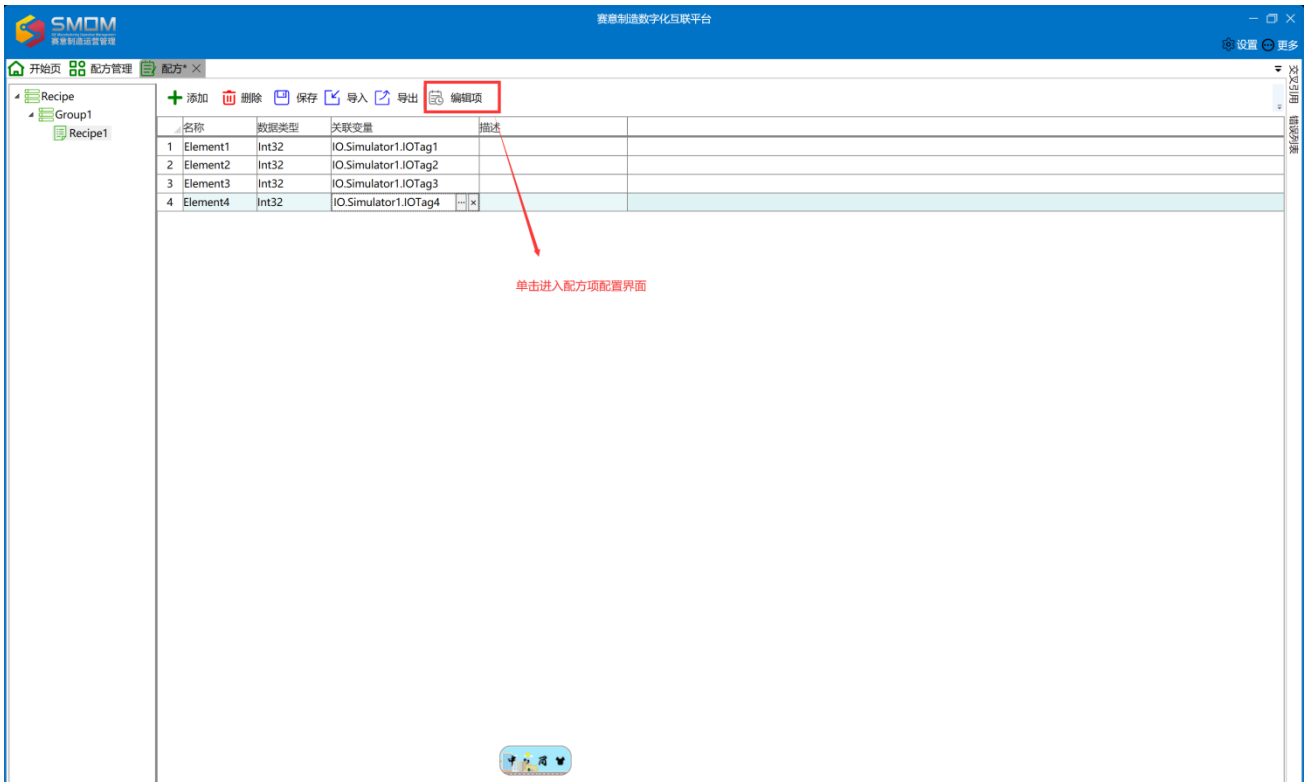
然后点击关联变量列的“...”省略号按钮,进入变量浏览器给每个组成元素关联 **Tag**,这样采集到的 **Tag** 值就能与元素项对应上:



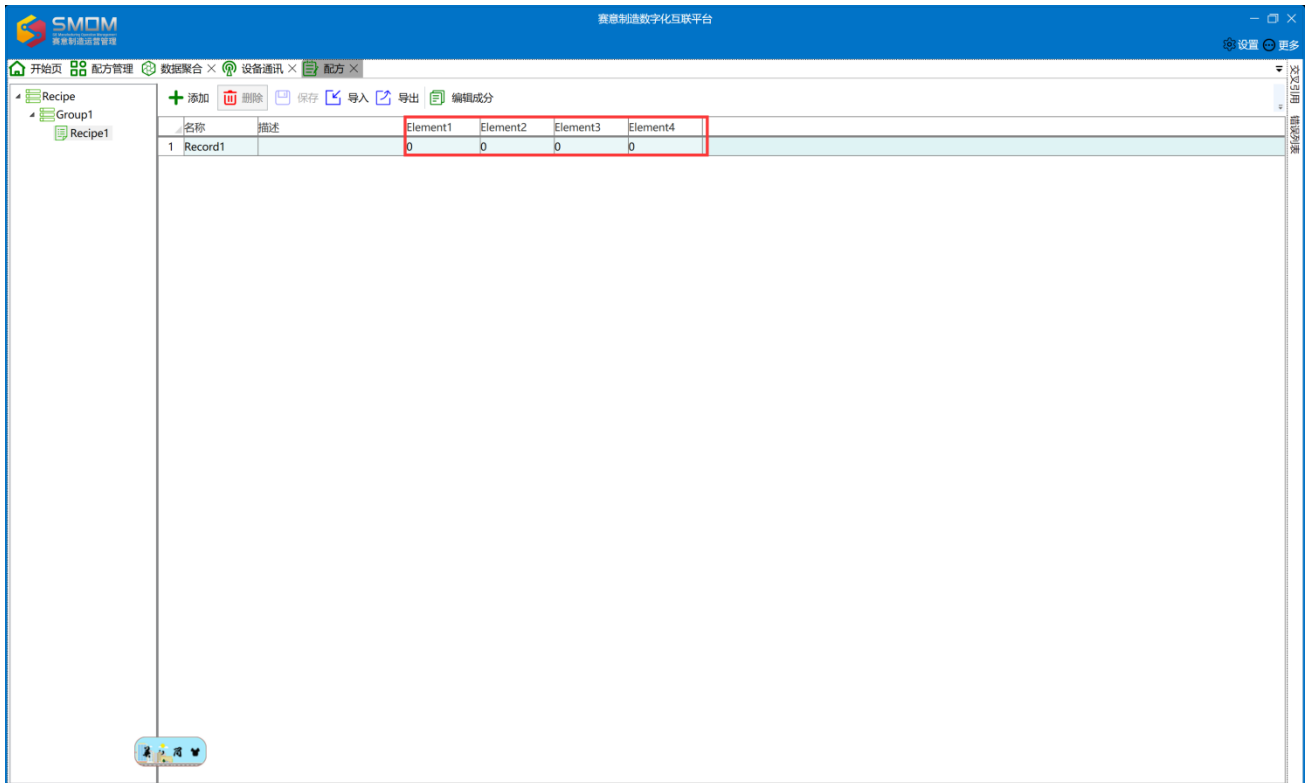
在这里给组成元素 1、元素 2、元素 3、元素 4 分别关联了四个设备点：



Step3: 然后点击菜单栏最右侧的“编辑项”按钮，进入编辑配方项的界面，配方真正要上传下载的就是这个配方项。



点击增添按钮，了一个配方项 Record1，可以看到该配方项和组成元素已自动关联显示：



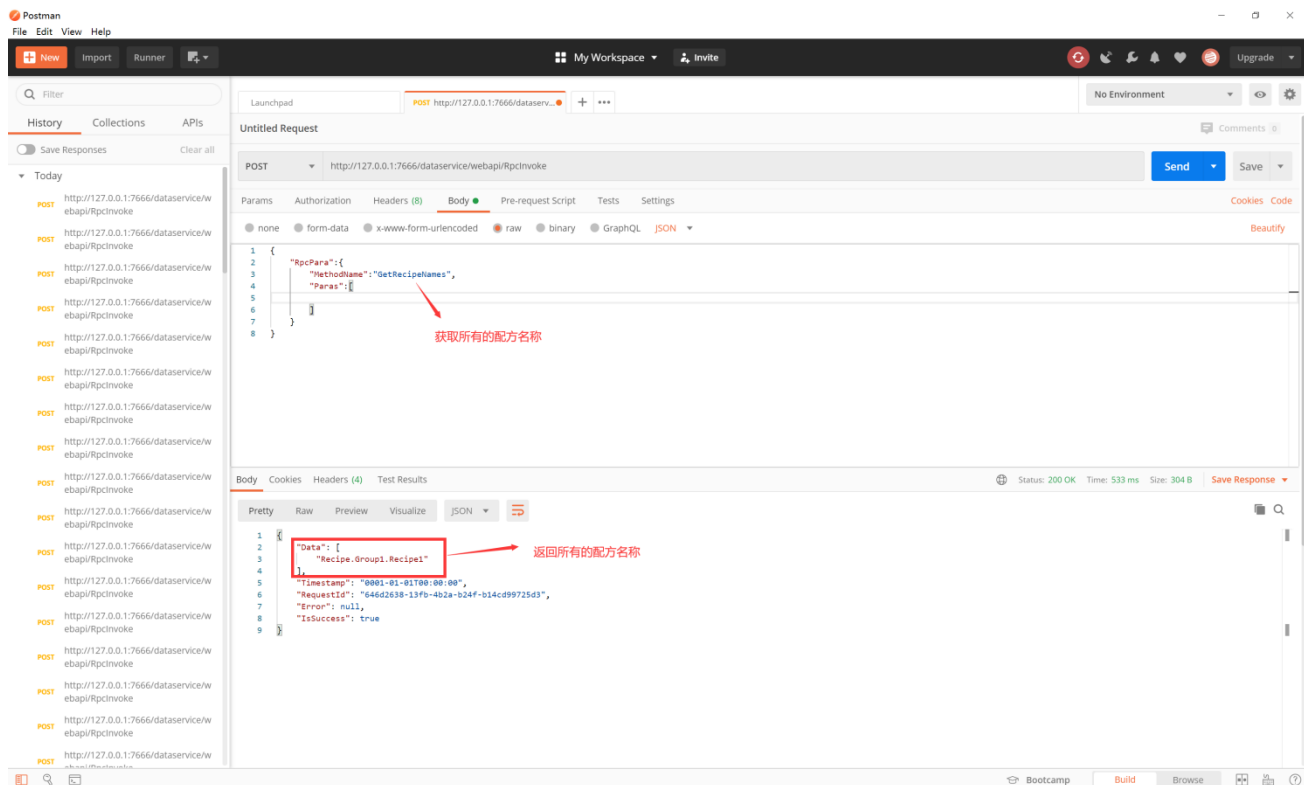
到此，配方的界面配置已完成，可点击“开始页”选项卡回到主页面再点击“运行”启动工程，或是按 F5 启动工程。

在此我们可以想象成工厂要生产一瓶可乐是需要一个配方的，这个配方就是水、碳水化合物、白糖、焦糖来组成，这里分别对应了我们新建组成元素的 Element1、Element2、Element3、Element4，但水、碳水化合物、白糖、焦糖混合生产时，各剂量不同得到的可乐口味就不一样，所以就有了我们配置的配方项，即水、碳水化合物、白糖、焦糖在特定剂量时的配方。所以当我们想得到更甜的可乐时，需要将白糖剂量调大至 5 克，不需要太甜时调至 2 克，这时只需要把 IO.Simulator1.IOTag3（与 Element3 关联的 Tag 值）的值调到相应的值，然后再下达配方项即可。

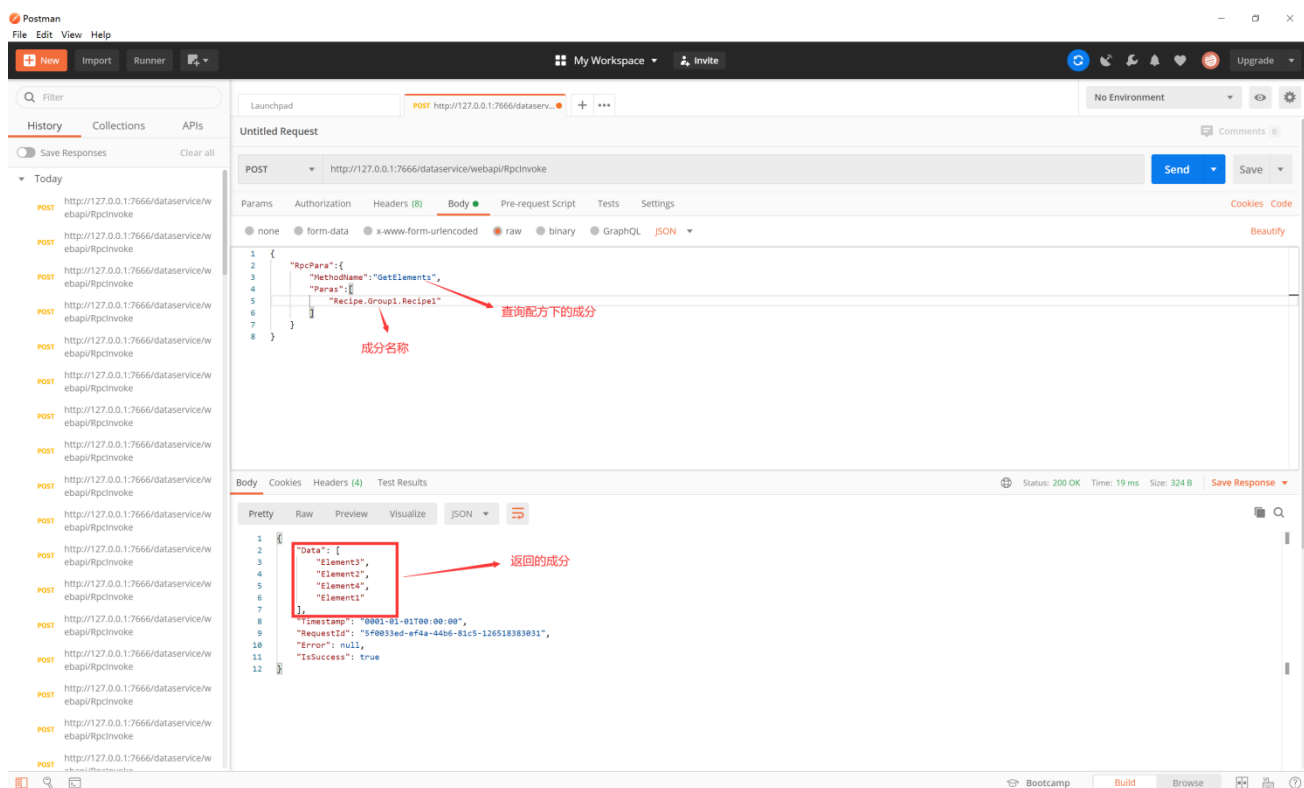
## 14.4 webapi 配方下达

接下来通过 webapi 来演示配方下达。如上，我们已启动了工程，在此我们可通过 PostMan 发送请求来查询有哪些配方（请求地址：[http://localhost:7666/swagger/ui/index#!/WebApi/WebApi\\_RpclInvoke](http://localhost:7666/swagger/ui/index#!/WebApi/WebApi_RpclInvoke)

请求内容 (获取配方名称 Usorn.txt):



然后我们再查询配方项的组成元素



此时，假若我们要生产由配方 1 克水、2 克碳水化合物、3 克白糖、4 克焦糖组成的可乐，首先要回到运行时界面修改值：

SMDC配方控制器 8.2 版

工程名称: 配方管理  
 启动时间: 2020/6/29 18:12:22  
 持续时间: 0 天 0 时 5 分 46 秒  
 IO点数: 10

IO信息 | Tag变量

设备名称: Simulator1

设备状态: Scanning 通讯状态: Communicate 通道: VirtualChannel1 IsLowSpeedScanning: False IOController: 277345431861035008

名称	数据类型	地址	读写类型	关联变量	变量值	工程值	质量戳	时间戳	写入值状态	描述
IOTag1	Long	Static,0,1	ReadWrite	1	1	1	Good	2020/6/29 18:18:09	Good	
IOTag2	Long	Static,1,1	ReadWrite	2	2	2	Good	2020/6/29 18:18:09	Good	
IOTag3	Long	Static,2,1	ReadWrite	3	3	3	Good	2020/6/29 18:18:09	Good	
IOTag4	Long	Static,3,1	ReadWrite	4	4	4	Good	2020/6/29 18:18:09	Good	
IOTag5	Long	Static,4,1	ReadWrite	0	0	0	Good	2020/6/29 18:18:09	Unknown	
IOTag6	Long	Static,5,1	ReadWrite	0	0	0	Good	2020/6/29 18:18:09	Unknown	
IOTag7	Long	Static,6,1	ReadWrite	0	0	0	Good	2020/6/29 18:18:09	Unknown	
IOTag8	Long	Static,7,1	ReadWrite	0	0	0	Good	2020/6/29 18:18:09	Unknown	
IOTag9	Long	Static,8,1	ReadWrite	0	0	0	Good	2020/6/29 18:18:09	Unknown	
IOTag10	Long	Static,9,1	ReadWrite	0	0	0	Good	2020/6/29 18:18:09	Unknown	

Annotations: Element1, 水; Element2, 水; Element3, 白糖; Element4, 焦糖

然后通过 webapi 下达配方项:

Postman

File Edit View Help

My Workspace

Launchpad POST http://127.0.0.1:7666/dataserv...

Untitled Request

POST http://127.0.0.1:7666/dataservice/webapi/RpcInvoke

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies Code

Body form-data x-www-form-urlencoded raw binary GraphQL JSON

```

1 {
2   "RpcPara": {
3     "MethodName": "UpdateFromTag",
4     "Params": {
5       "Recipe_Group1_Recipe1",
6       "Record1"
7     }
8   }
9 }

```

从Tag读值，并下达配方

Body Cookies Headers (4) Test Results Status: 200 OK Time: 25 ms Size: 283 B Save Response

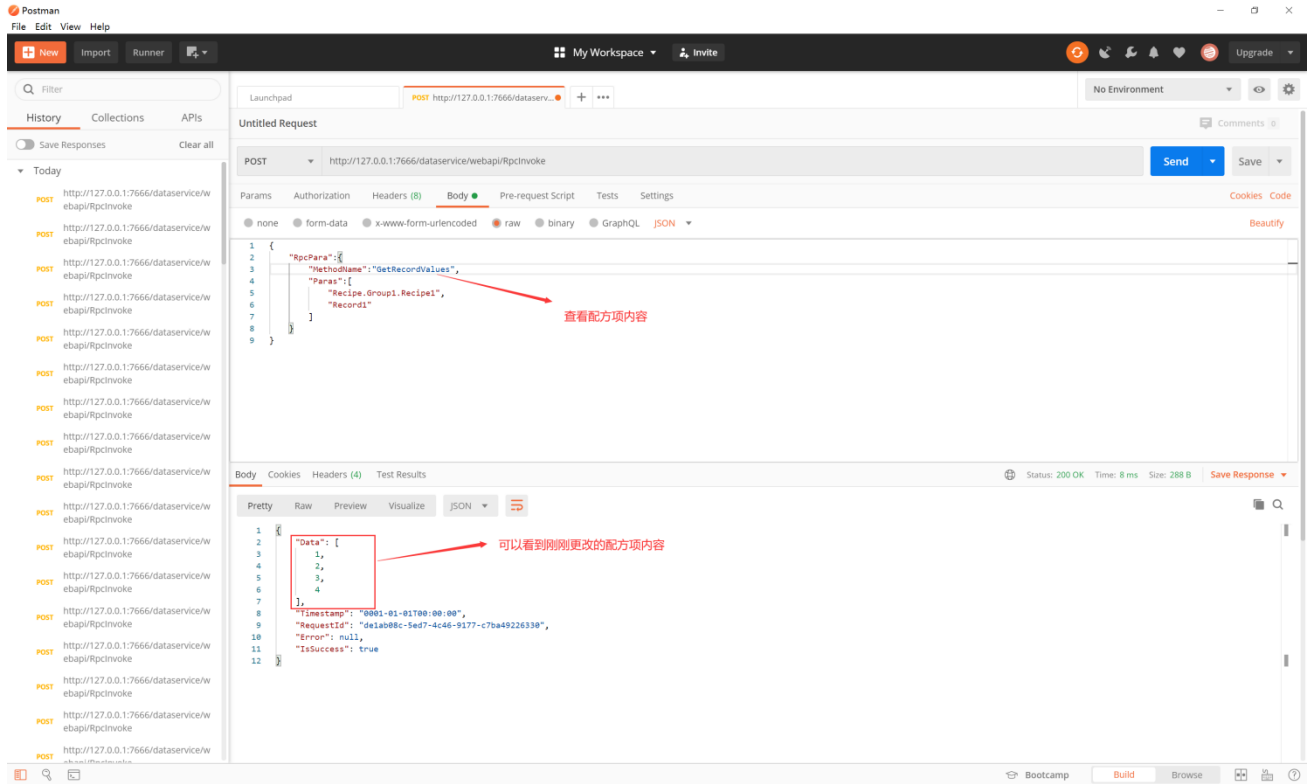
Pretty Raw Preview Visualize JSON

```

1 {
2   "Data": true,
3   "Timestamp": "0001-01-01T00:00:00",
4   "RequestId": "0e4b4e48-4596-4f03-9f05-8a3bb0dfef3d",
5   "Error": null,
6   "IsSuccess": true
7 }

```

此时，查询配方项可以看到配方项的值已改变，说明配方项已生效:



至此配方功能的使用介绍完毕。

