

智能公元平台 协议自定义手册



深圳市机芯智能有限公司

www.aimachip.com

www.smartpi.cn

1. 概述

在产品设计中，通常需要在各模块/芯片之间进行通讯，比如通过串口发送和接收消息，这就要求在通讯的两端约定消息的格式，这就是通讯协议的主要内容。对于 Wi-Fi 这类通讯产品来说，通讯协议的实现更是产品开发的重要部分，代码编写和调试都需要花费大量的时间。为了降低产品开发的难度，减小开发的工作量，提高产品协议对接的灵活性，智能公元平台提供了协议自定义的能力。

智能公元平台支持对如下格式种类的协议进行定义：

- ✧ 二进制协议
- ✧ JSON 协议
- ✧ 文本协议

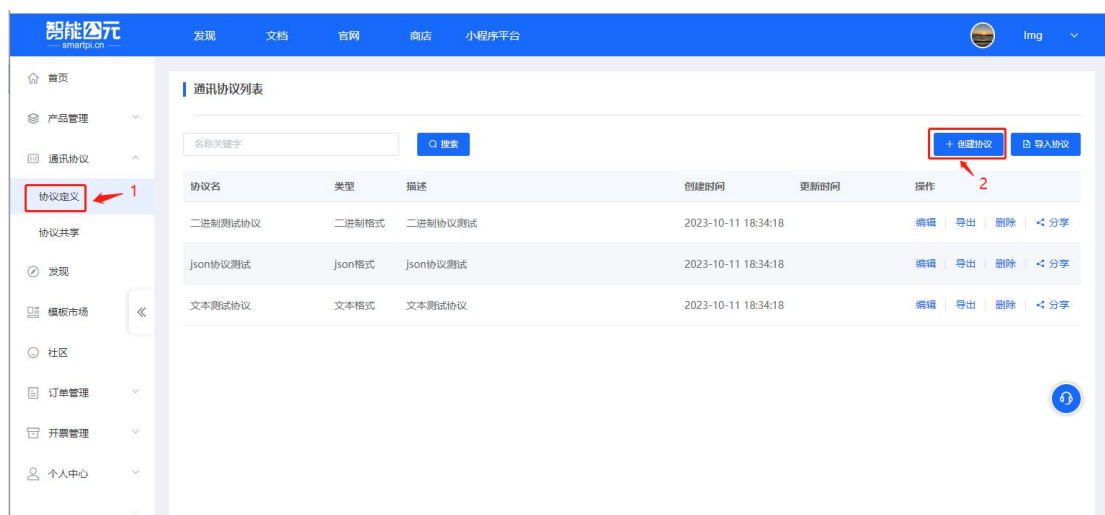
2. 基本术语

- ✧ 协议：包含一条以上的消息，并指定了格式类型为二进制协议、JSON 协议还是文本协议
- ✧ 消息：包含一个消息结构定义和一个以上的消息模板
- ✧ 消息结构：定义消息由哪些字段组成，设置字段的类型和长度等要求
- ✧ 消息模板：在消息结构基础上，将一些字段设置为固定值，一些字段设置为待定值，就成为了一个消息模板。给消息模板传入待定字段的具体值，平台使用内部的协议编码算法，就可以生成需要的消息内容。相反，如果给消息模板传入具体的消息内容，平台使用内部的协议解码算法，就可以检查消息是否匹配模板，如果匹配，就可以获得模板待定字段的具体值。

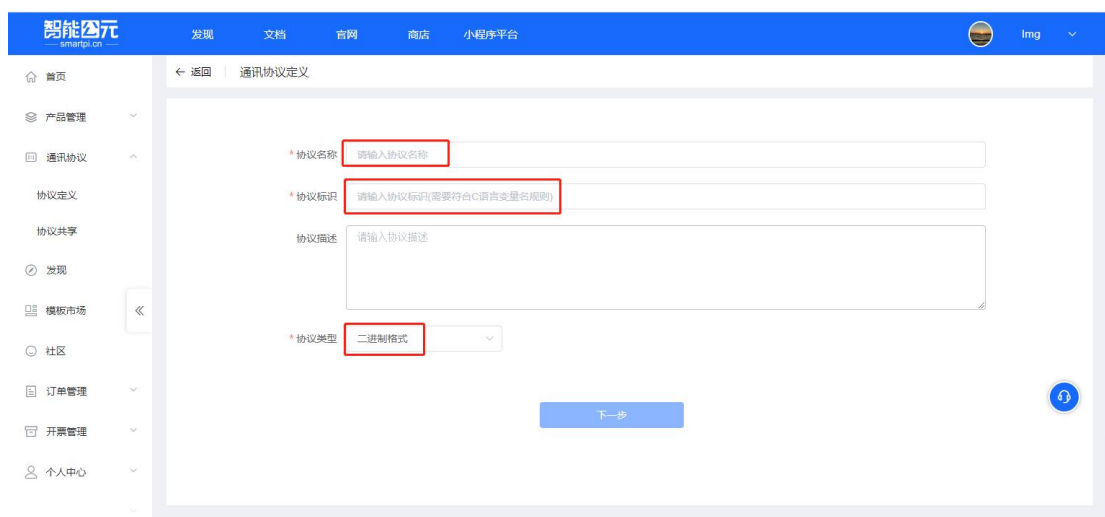
3. 二进制协议定义

3.1. 创建二进制协议

登录智能公元平台（<http://www.smartpi.cn>），按下图顺序操作：



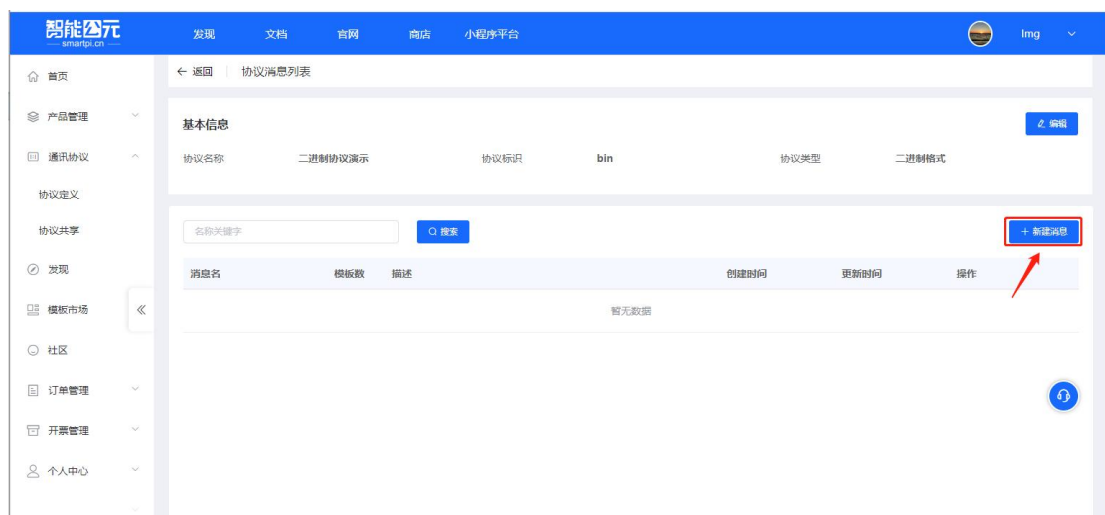
在下图中输入协议名称、协议标识、选择“二进制格式”，点击“下一步”按钮，完成协议创建：



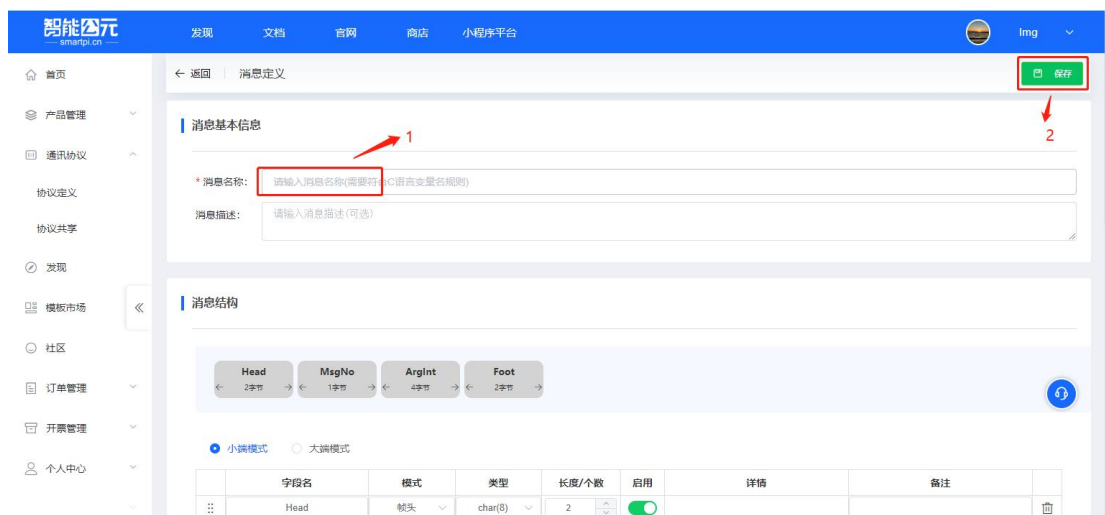
说明：需要填写符合 C 语言变量名规则的协议标识，协议标识是平台生成算法的要求。

3.2. 创建二进制消息

进入协议后，在下图中，点击“新建消息”：

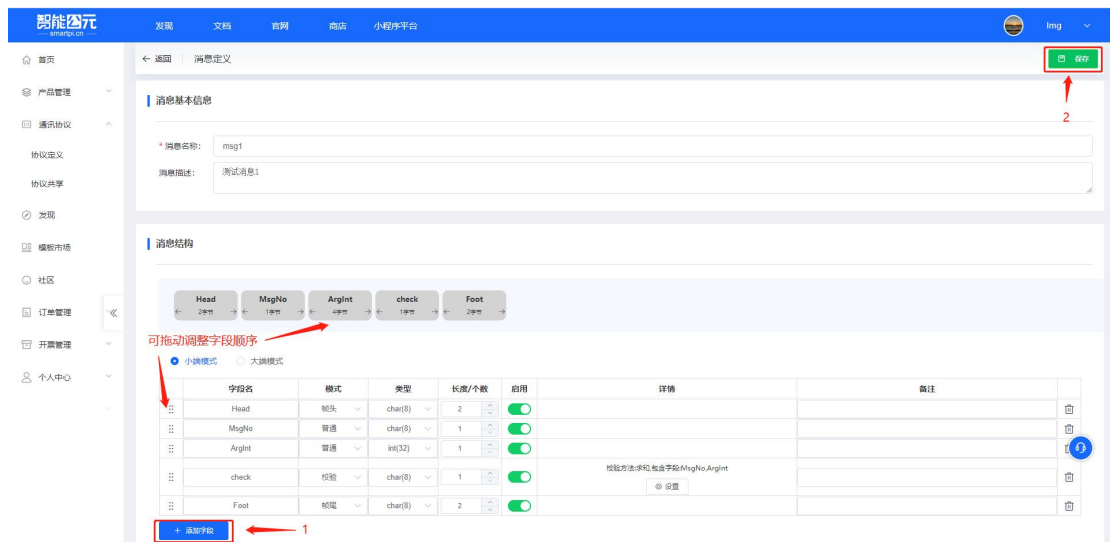


在下图中输入“消息名称”，点击“保存”按钮，完成消息创建：



3.3. 配置二进制消息结构

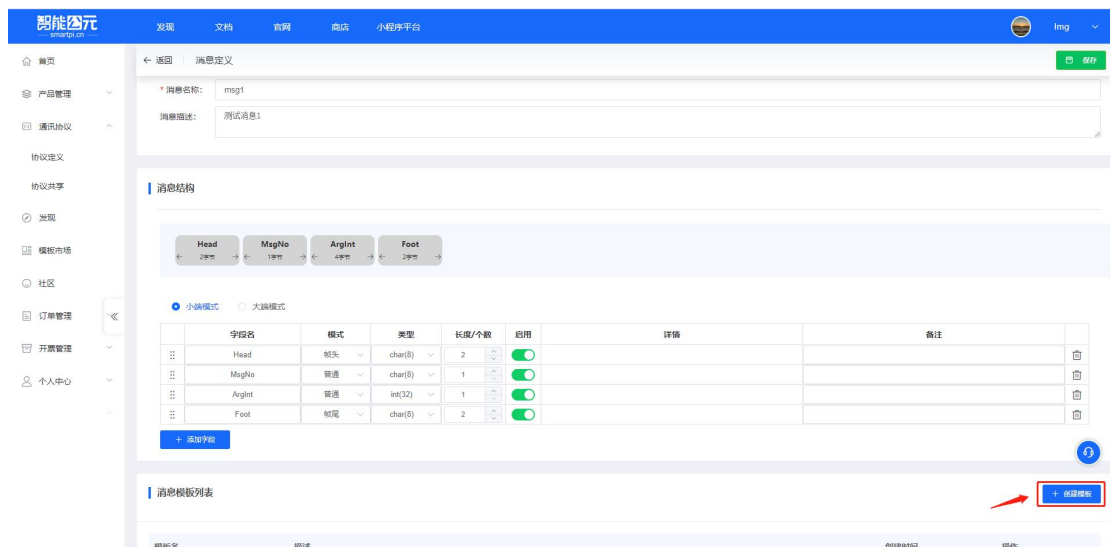
在下图中，可点击“添加字段”按钮增加消息字段，并可配置字段的模式、类型、长度等，配置完后点击“保存”按钮：



说明：消息字段可以拖动调整字段顺序；支持配置校验字段，校验算法目前支持“求和”、“异或”两种，校验字段需要在详情中设置哪些字段需要参与检验。

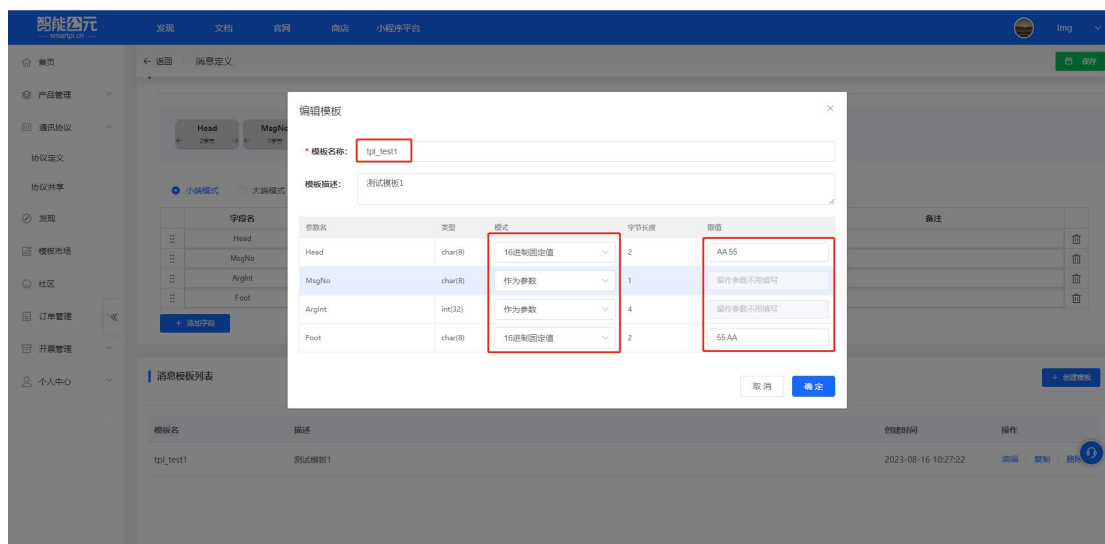
3.4. 创建二进制消息模板

在下图中点击“创建模板”按钮：



3.5. 配置二进制消息模板

在下图中输入“模板名称”、选择模板字段的“模式”，输入模板字段的“取值”，点击“确定”按钮：



说明：

模板字段的模式为“固定值”，需要在“取值”中按格式要求填写，发送消息时，这个字段会使用这里填写的“取值”来填充，接收消息时，会要求这个字段的值是这里填写的“取值”；

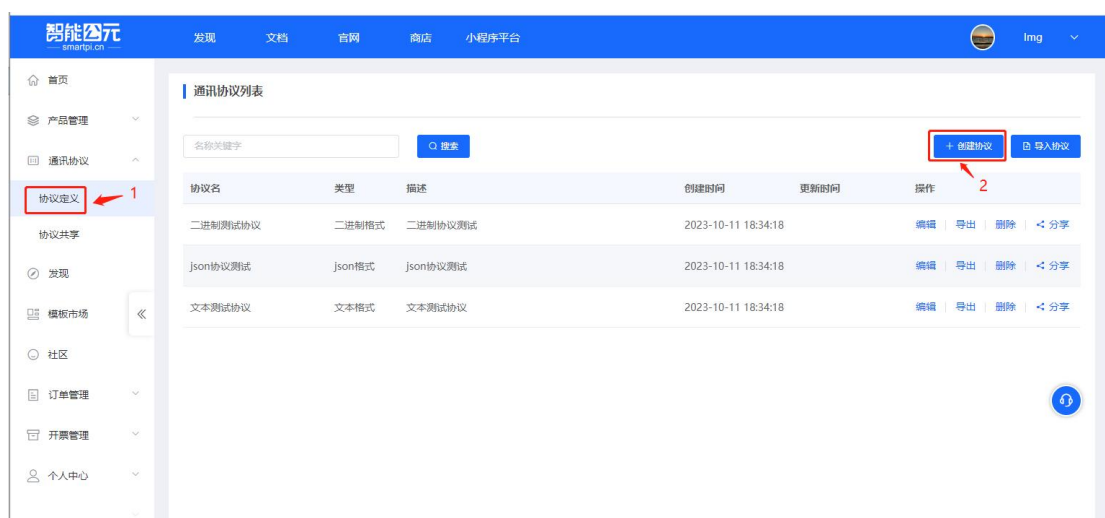
模板字段的模式为“默认留空”，发送消息时，这个字段将填 0，接收消息时，不关心这个字段的值；

模板字段的模式为“作为参数”，这个字段就是作为待定值，发送消息时再填入具体值，接收消息时，可以将这个字段的值解析并传出使用。

4. JSON 协议定义

4.1. 创建 JSON 协议

登录智能公元平台（<http://www.smartpi.cn>），按下图顺序操作：



在下图中输入协议名称、协议标识、选择“json 格式”，点击“下一步”按钮，完成协

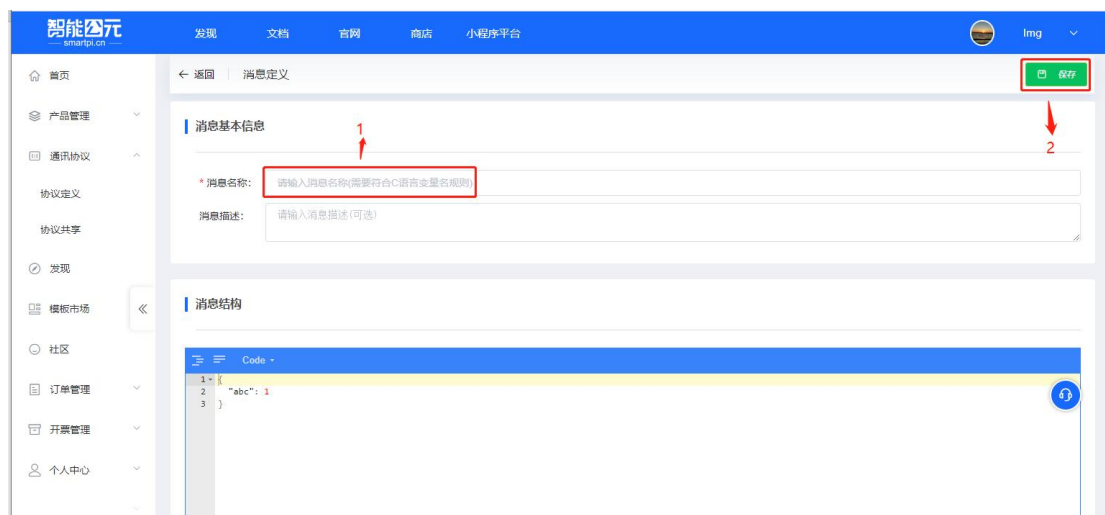
议创建:

说明：需要填写符合 C 语言变量名规则的协议标识，协议标识在平台生成算法的要求。

4.2. 创建 JSON 消息

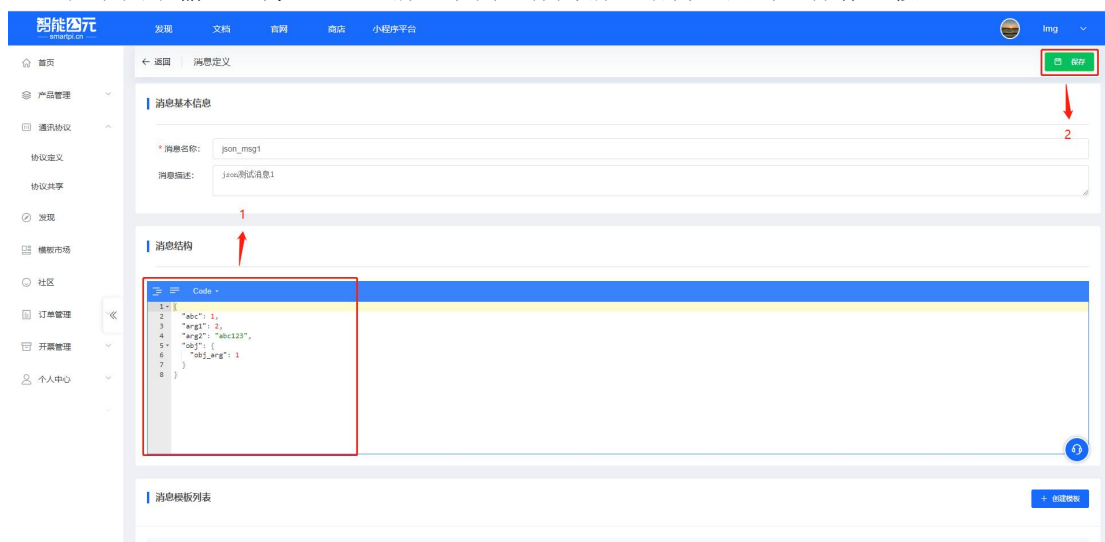
进入协议后，在下图中，点击“新建消息”：

在下图中输入“消息名称”，点击“保存”按钮，完成消息创建：



4.3. 配置 JSON 消息结构

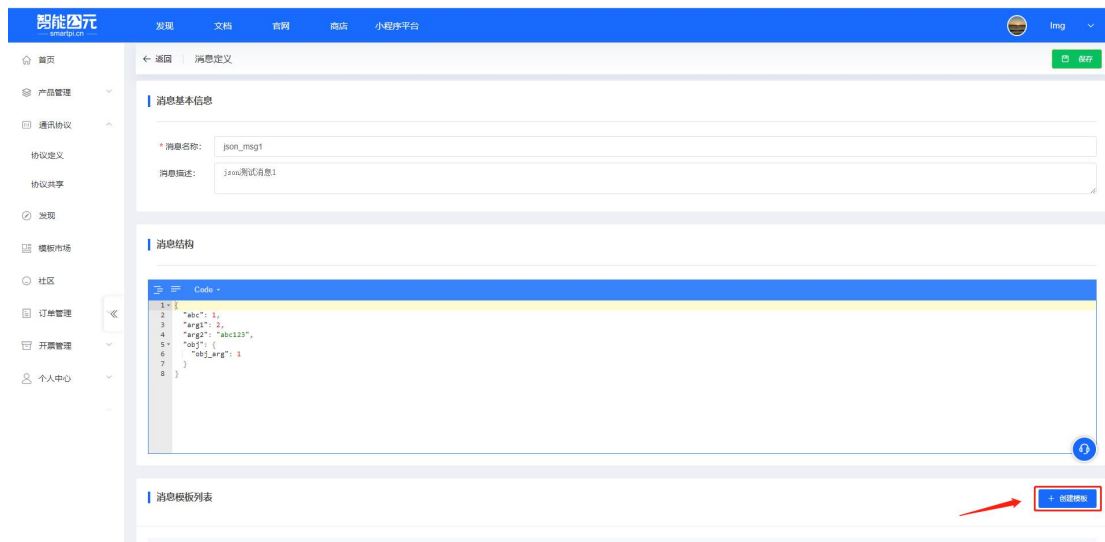
在下图中输入或拷入 JSON 消息示例，作为消息结构，点击“保存”按钮：



说明：如果输入的 JSON 消息不合格式要求，会有错误提示。

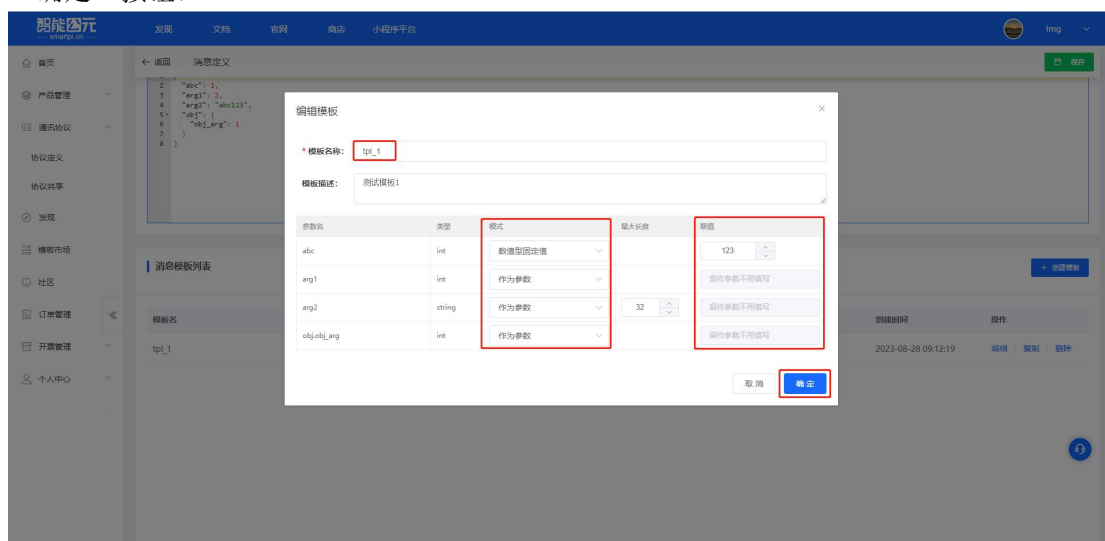
4.4. 创建 JSON 消息模板

在下图中点击“创建模板”按钮：



4.5. 配置 JSON 消息模板

在下图中输入“模板名称”、选择模板字段的“模式”，输入模板字段的“取值”，点击“确定”按钮：



说明：

模板字段的模式为“固定值”，需要在“取值”中按格式要求填写，发送消息时，这个字段会使用这里填写的“取值”来填充，接收消息时，会要求这个字段的值是这里填写的“取值”；

模板字段的模式为“默认留空”，发送消息时，这个字段将填 0，接收消息时，不关心这个字段的值；

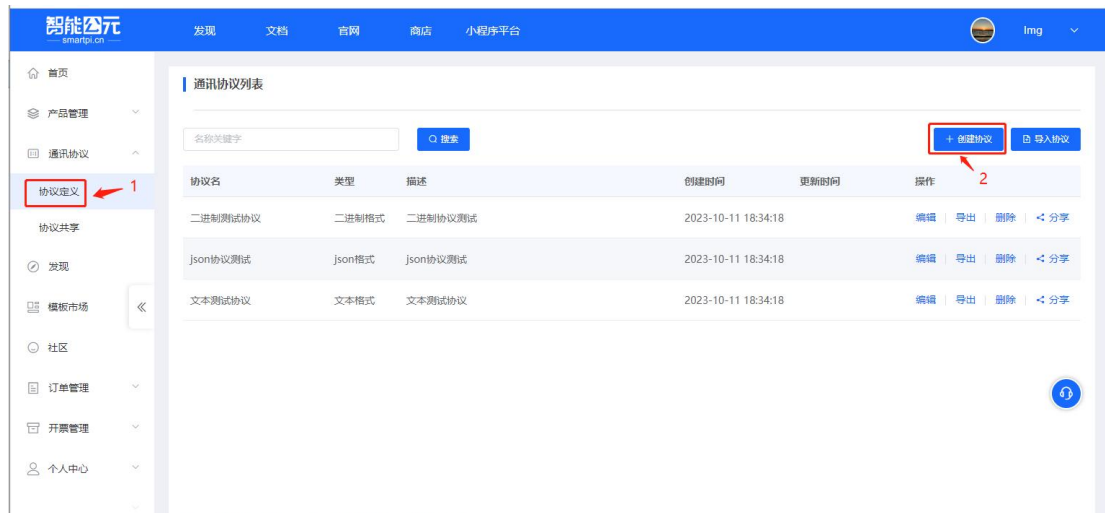
模板字段的模式为“作为参数”，这个字段就是作为待定值，发送消息时再填入具体值，接收消息时，可以将这个字段的值解析并传出使用。

当字段类型为 string 且“作为参数”时，需要指定其最大长度，保证平台算法有地址存放其内容。

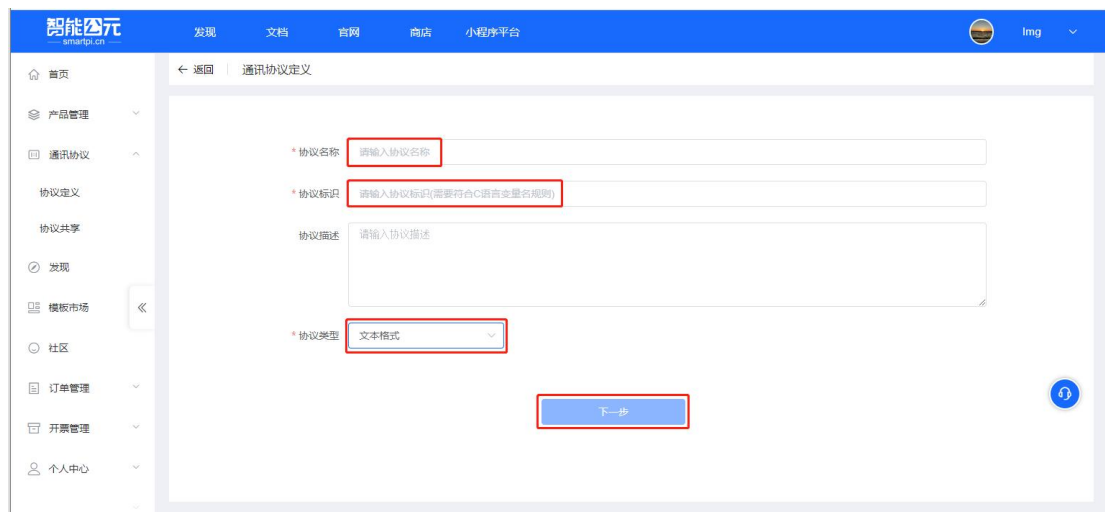
5. 文本协议定义

5.1. 创建文本协议

登录智能公元平台（<http://www.smartpi.cn>），按下图顺序操作：



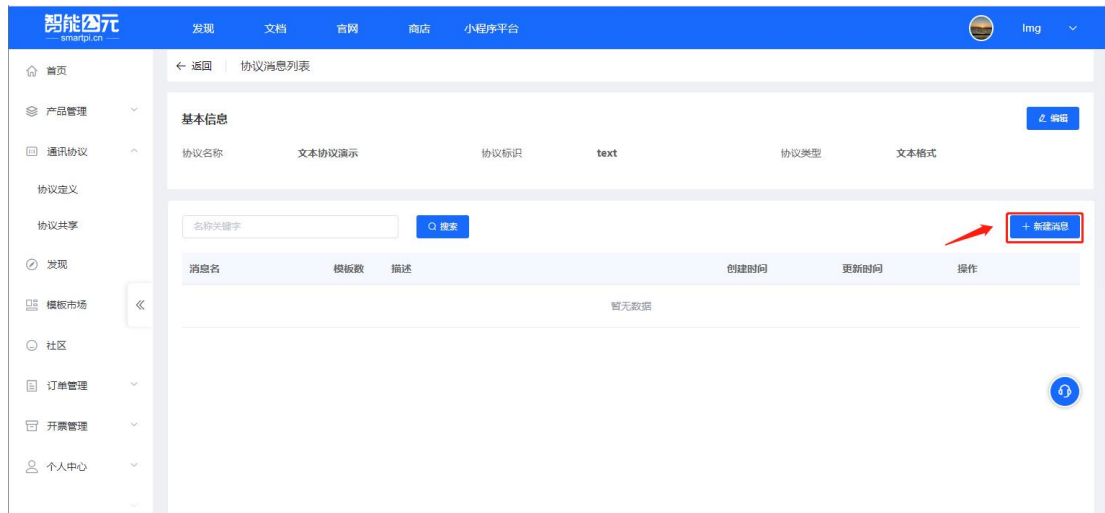
在下图中输入协议名称、协议标识、选择“文本格式”，点击“下一步”按钮，完成协议创建：



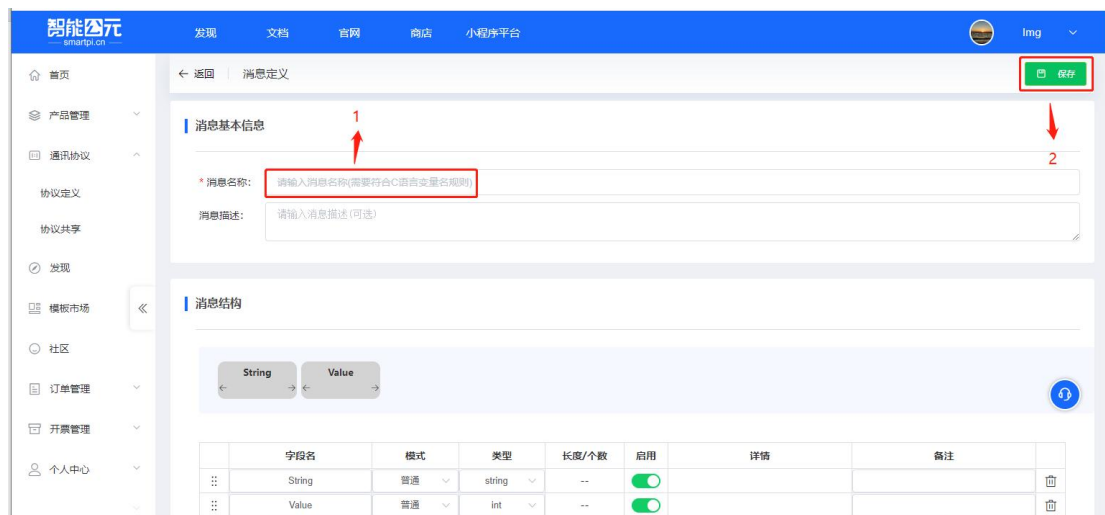
说明：需要填写符合 C 语言变量名规则的协议标识，协议标识在平台生成算法的要求。

5.2. 创建文本消息

进入协议后，在下图中，点击“新建消息”：

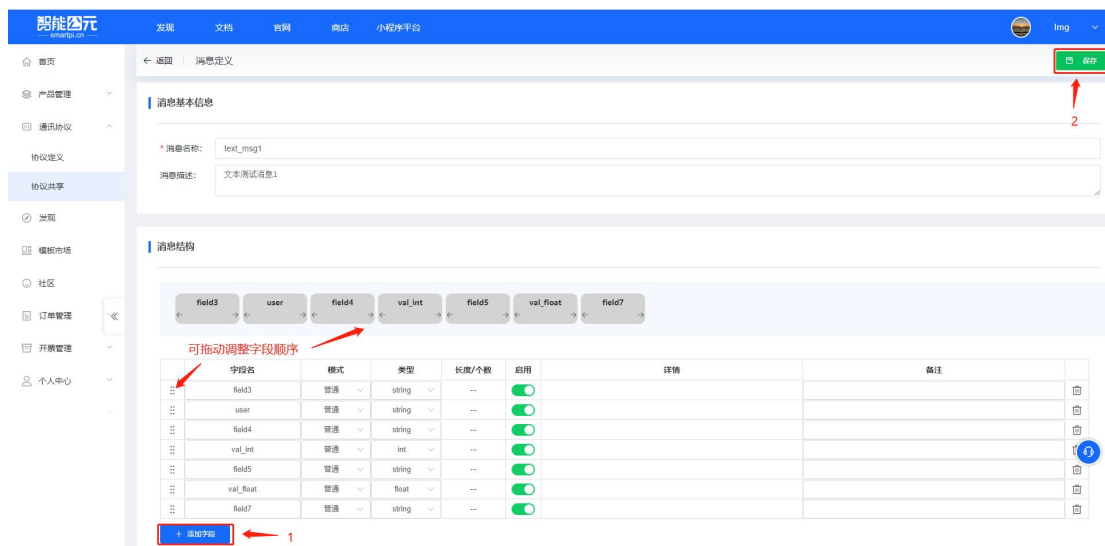


在下图中输入“消息名称”，点击“保存”按钮，完成消息创建：



5.3. 配置文本消息结构

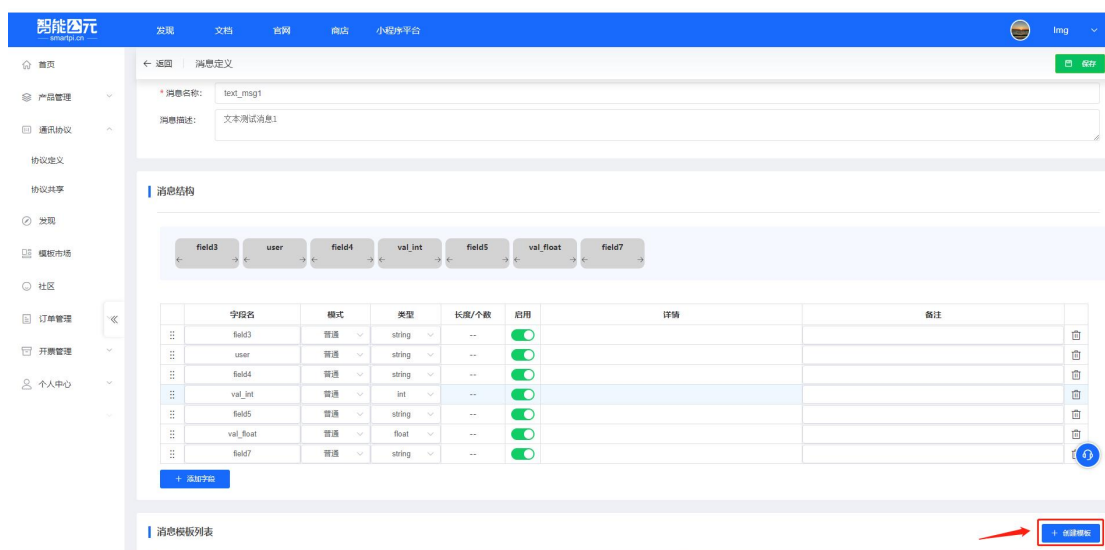
在下图中，可点击“添加字段”按钮增加消息字段，并可配置字段的模式、类型、长度等，配置完后点击“保存”按钮：



说明：消息字段可以拖动调整字段顺序。

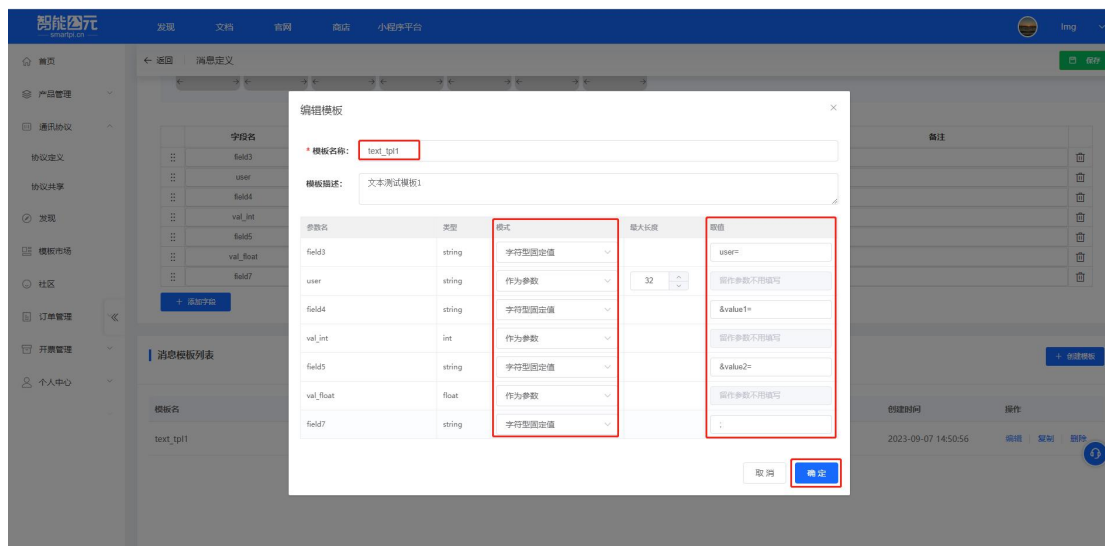
5.4. 创建文本消息模板

在下图中点击“创建模板”按钮：



5.5. 配置文本消息模板

在下图中输入“模板名称”、选择模板字段的“模式”，输入模板字段的“取值”，点击“确定”按钮：



说明：

模板字段的模式为“固定值”，需要在“取值”中按格式要求填写，发送消息时，这个字段会使用这里填写的“取值”来填充，接收消息时，会要求这个字段的值是这里填写的“取值”；

模板字段的模式为“默认留空”，发送消息时，这个字段将填 0，接收消息时，不关心这个字段的值；

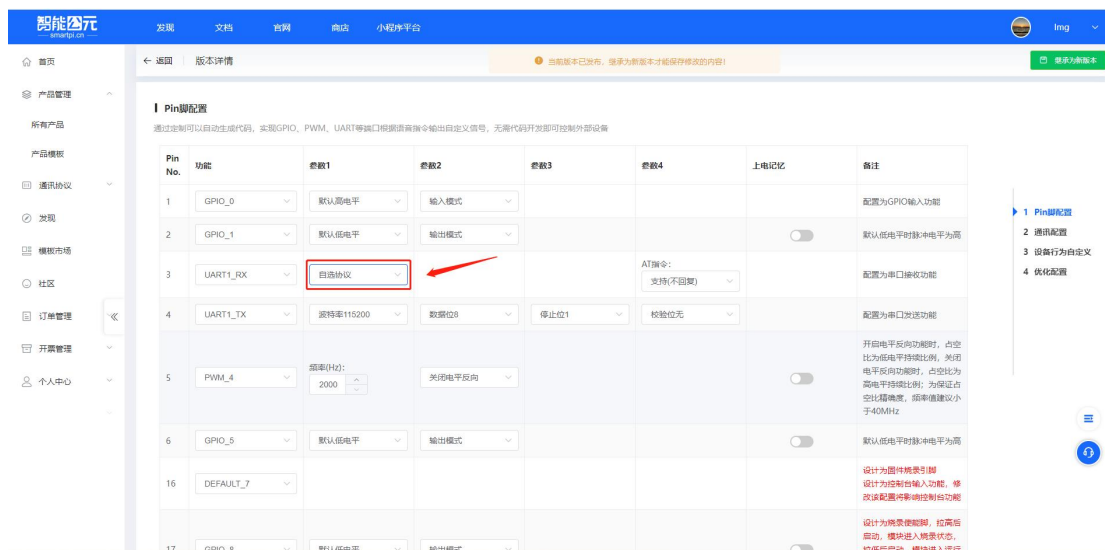
模板字段的模式为“作为参数”，这个字段就是作为待定值，发送消息时再填入具体值，接收消息时，可以将这个字段的值解析并传出使用。

当字段类型为 string 且“作为参数”时，需要指定其最大长度，保证平台算法有地址存放其内容。

6. 使用协议

6.1. 在串口上使用自定义协议

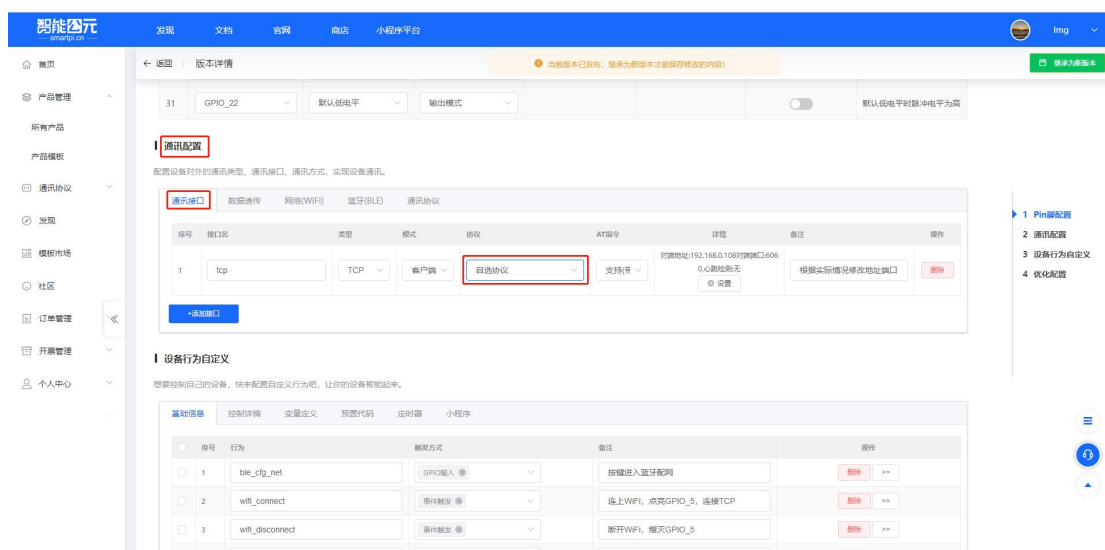
在产品的“PIN 脚配置”中，在串口的 RX 上，选择“自选协议”，参考下图：



说明：如果没有“自选协议”选项，则是该产品目前暂未支持自定义协议。

6.2. 在通讯接口上使用自定义协议

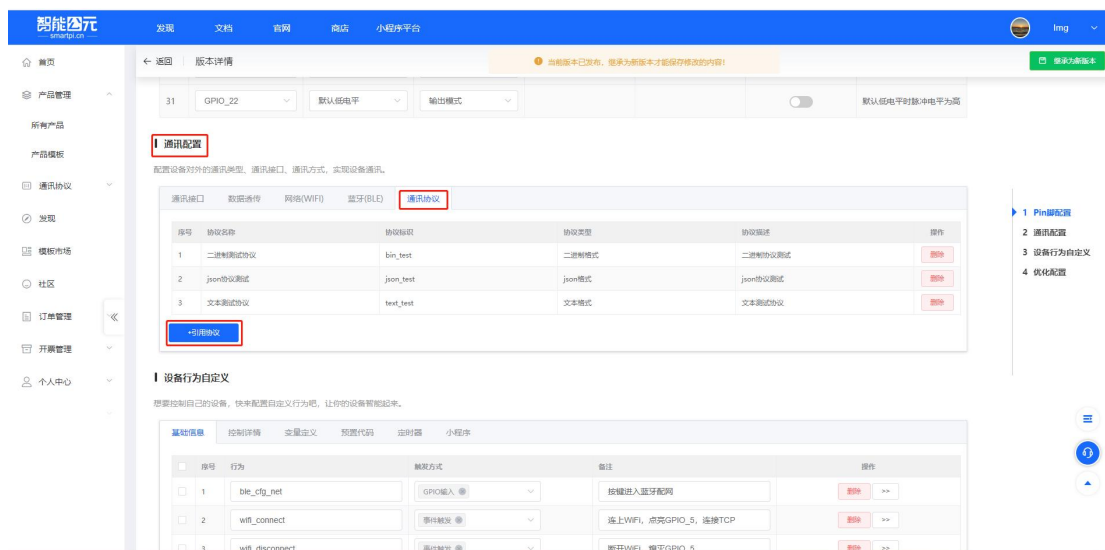
在产品的“通讯配置”中，在“通讯接口”上，选择“自选协议”，参考下图：



说明：如果没有“自选协议”选项，则是该产品目前暂未支持自定义协议。

6.3. 在产品配置中引用协议

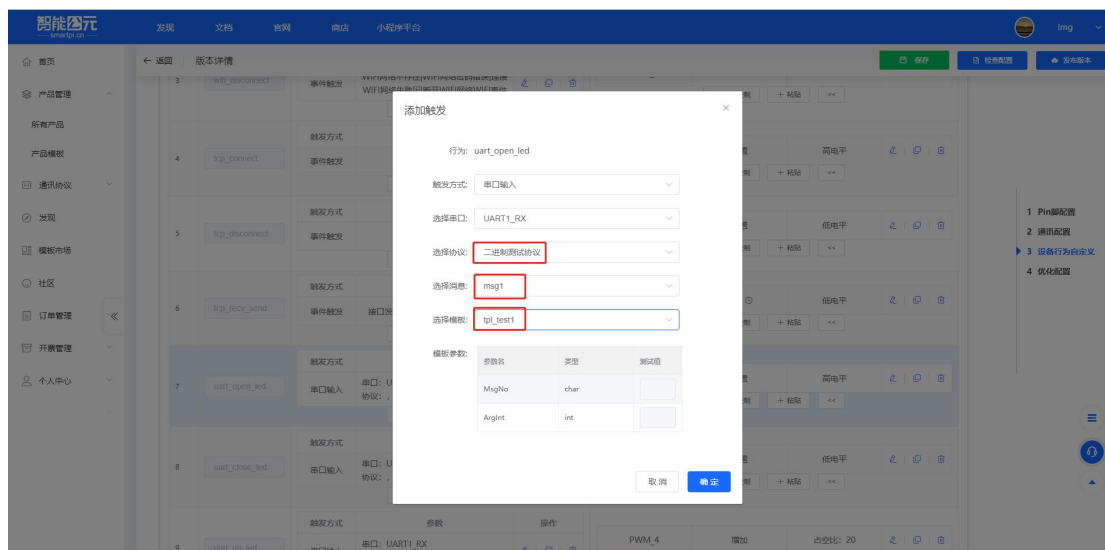
在产品的“通讯配置”中，在“通讯协议”上，点击“引用协议”，参考下图：



说明：自定义协议只有在引入产品配置后，才能在该产品中使用。

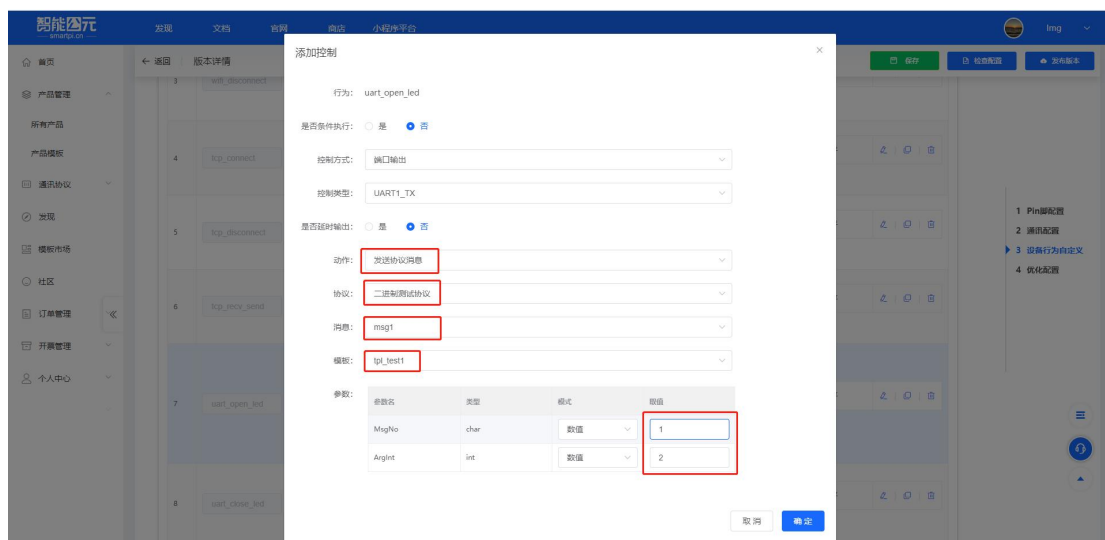
6.4. 在串口接收中使用自定义协议

当产品行为的触发方式为“串口输入”时，在“控制详情”中对该行为“添加触发”，就可以选择协议、消息、模板，根据模板自动生成消息接收参数，可参考下图：



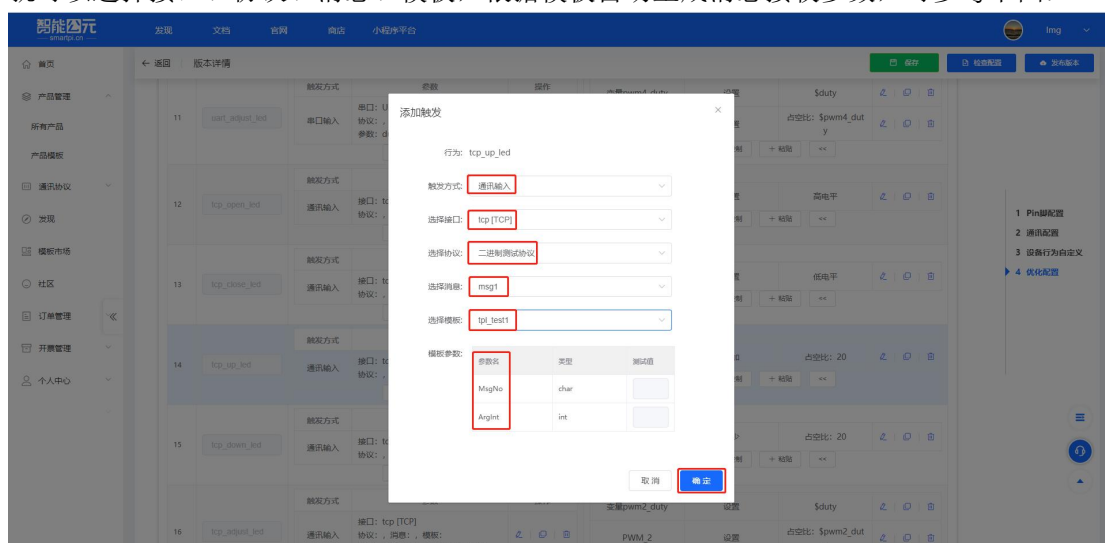
6.5. 在串口发送中使用自定义协议

在“控制详情”中，点击“添加控制”，选择控制方式为“端口输出”，控制类型为“UARTx_TX”，动作为“发送协议消息”，就可以选择协议、消息、模板，并根据模板生成参数列表，需要填写参数的“取值”，可参考下图：



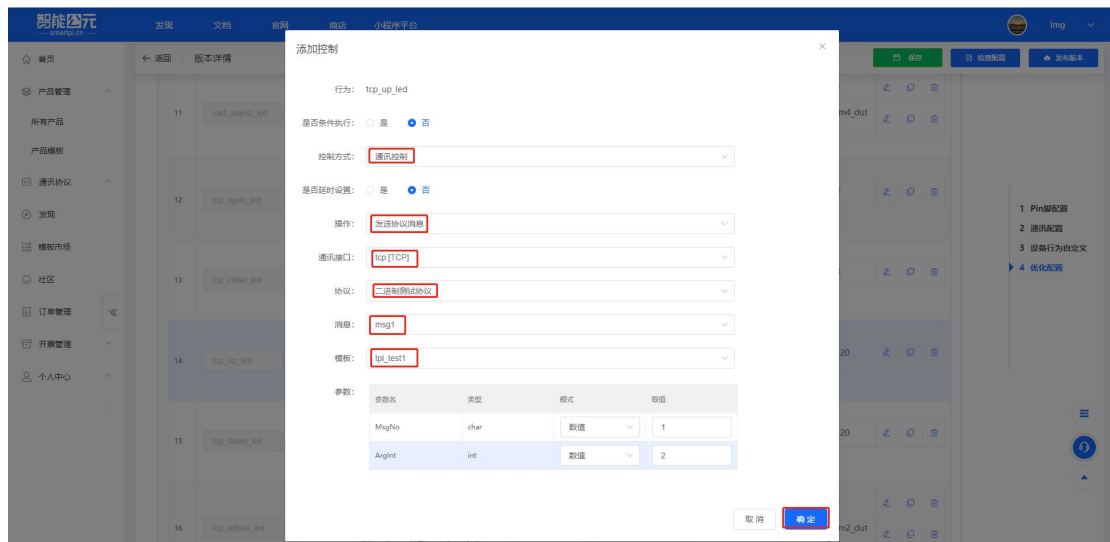
6.6. 在通讯接口接收中使用自定义协议

当产品行为的触发方式为“通讯输入”时，在“控制详情”中对该行为“添加触发”，就可以选择接口、协议、消息、模板，根据模板自动生成消息接收参数，可参考下图：



6.7. 在通讯接口发送中使用自定义协议

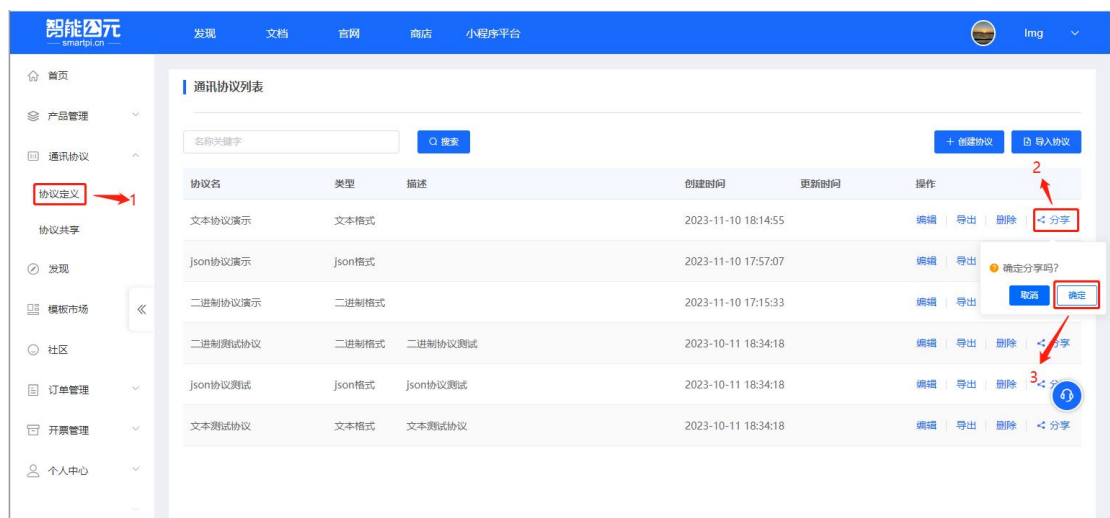
在“控制详情”中，点击“添加控制”，选择控制方式为“通讯控制”，选择“通讯接口”，动作为“发送协议消息”，就可以选择协议、消息、模板，并根据模板生成参数列表，需要填写参数的“取值”，可参考下图：



7. 共享协议

7.1. 将协议分享给其他人

已定义好的协议，可以分享给其他人，分享的协议，在“协议共享”中可以看到。分享的操作步骤参考下图：



7.2. 导入其他人共享的协议

在智能公元平台中，点击“协议共享”，可以看到其他人分享的协议，点击“导入”，可以将共享协议导入到“协议定义”中，就可以在产品中使用该协议了。可参考下图：

