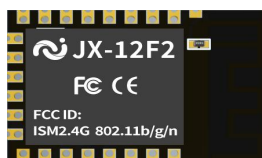


# JX-12F Wi-Fi 模块 技术开发手册



深圳市机芯智能有限公司

[www.aimachip.com](http://www.aimachip.com)

[www.smartpi.cn](http://www.smartpi.cn)

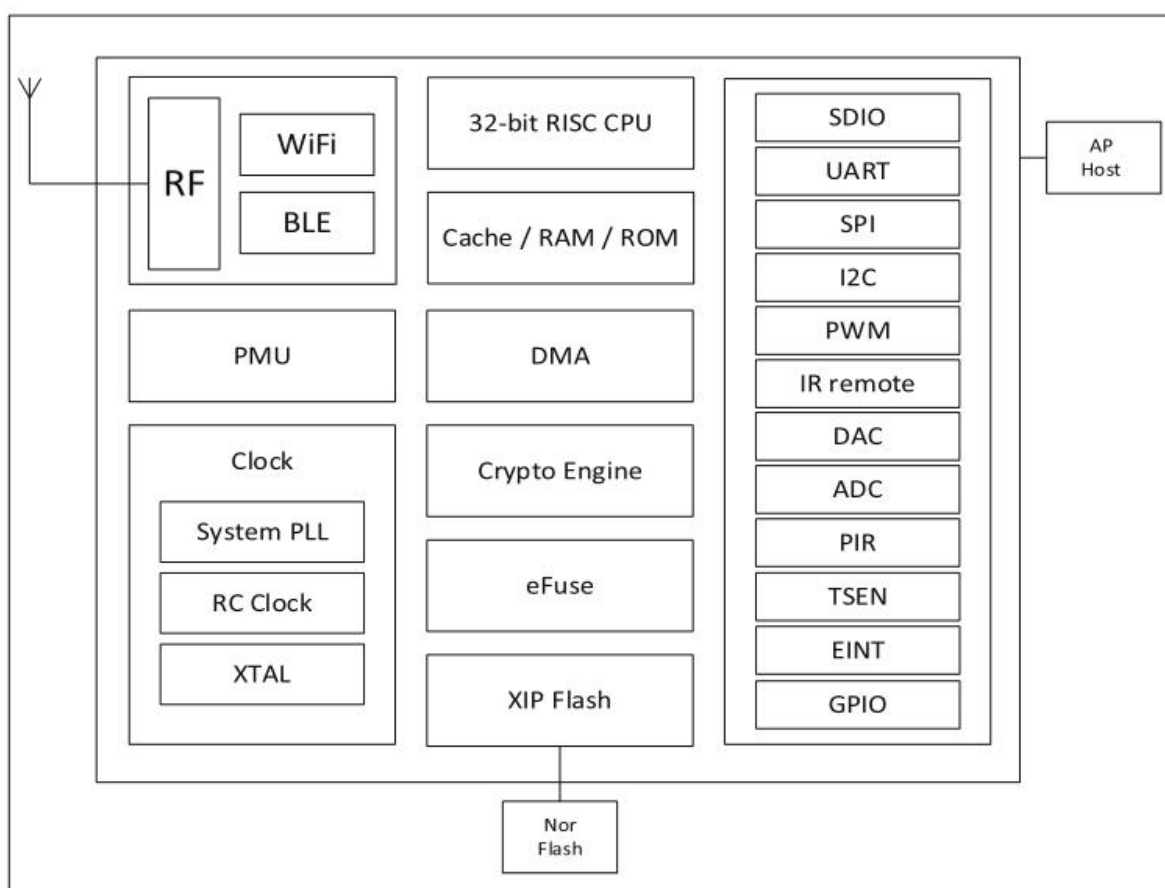
## 修订记录

[illegible]

# 1. 概述

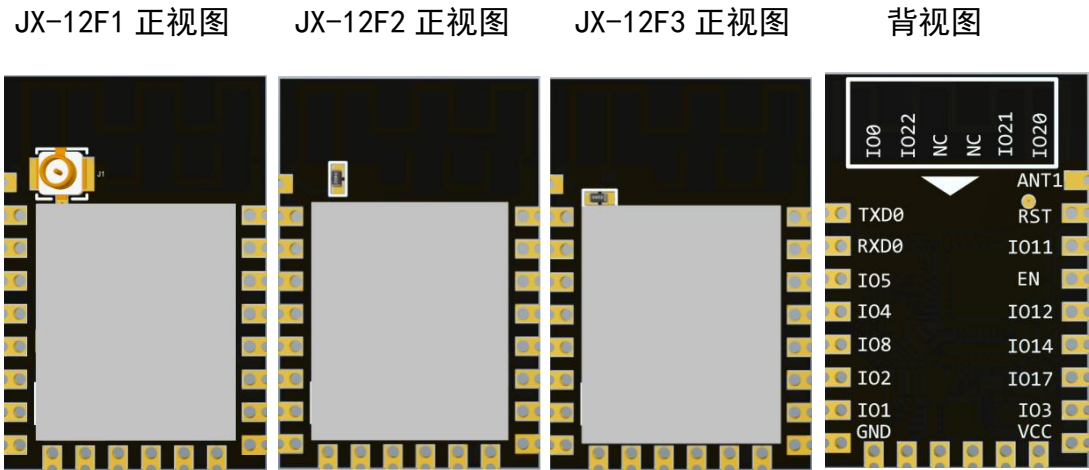
JX-12F 是一款基于 WiFi+BLE 单芯片 SoC 为主控的无线模组，它可以满足低功耗和高性能的 IOT 应用开发，该模组核心处理器 JX-1001 集成了 2.4G Wi-Fi（802.11b/g/n）和 BLE 5.0 的基带及 MAC 设计。其微控制器子系统包含一个低功耗的 32 位 RISC CPU，高速缓存和存储器。具有先进的电源管理单元，支持多种低功耗模式。外围接口包括 UART，GPIO，ADC，DAC，PWM，I2C，SDIO，SPI，IR 远程等。

JX-1001 功能框图：



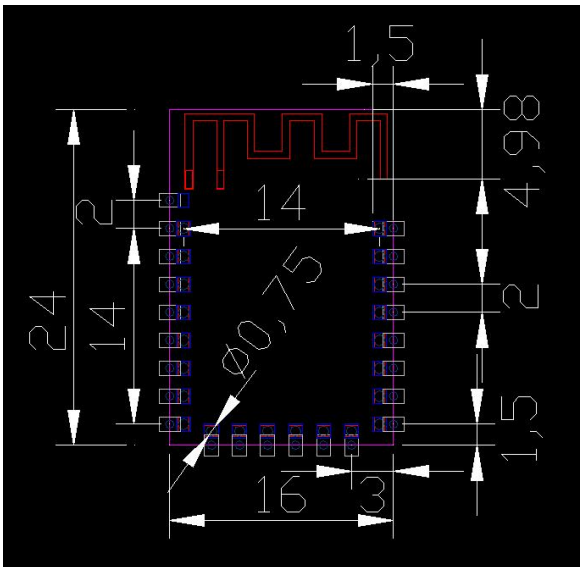
1.1. 外观尺寸

1.1.1 PCB 外观图

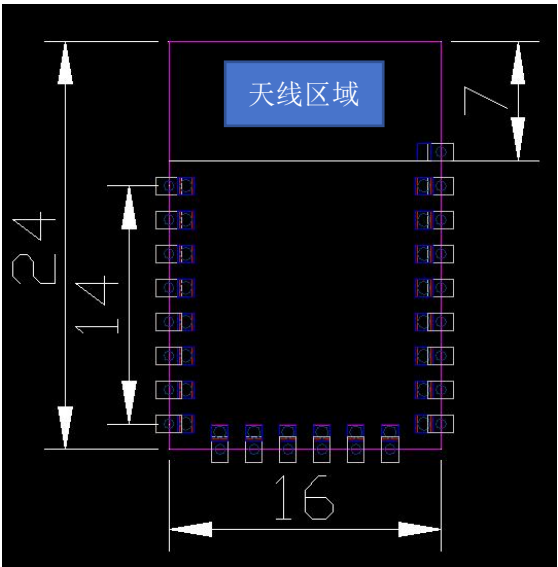


1.1.2 PCB 尺寸图 (单位 mm)

正视



后视



## 1.2. 特性

### 1.2.1. 无线

- IEEE 802.11b/g/n, 1x1 SISO 2.4GHz
- Bluetooth® 低功耗 5.0  
支持 BLE 5.0 通道选择 # 2
- Wi-Fi 20MHz 带宽
- Wi-Fi 安全 WPS/WEP/WPA/WPA2 Personal/  
WPA2 Enterprise/WPA3
- STA, SoftAP 和 Sniffer 模式
- Wi-Fi 和 BLE 共存, BLE 协助实现 Wi-Fi 快速连接
- 集成 Balun, PA/LNA
- 支持串口本地升级和远程固件升级 (FOTA)
- 通用 AT 指令可快速上手
- 支持二次开发, 集成了 Windows、Linux 开发环境

### 1.2.2. MCU 子系统

- 带 FPU (浮点单元) 的 32 位 RISC CPU
- 一个 RTC 计时器一年更新
- 两个 32 位通用定时器
- 四个 DMA 通道
- DFS (动态频率缩放) 从 1MHz 到 192MHz
- JTAG 开发支持

- XIP QSPI Flash 具有硬件加密支持

### 1.2.3. 内存

- 276KB RAM
- 128KB ROM
- 1Kb eFuse
- 嵌入式 Flash 2M

### 1.2.4. 安全机制

- QSPI Flash 即时 AES 解密 (OTFAD) - AES - 128, CTR 模式
- 支持 AES 128/192/256 位加密引擎
- 支持 SHA-1/224/256
- 真实随机数发生器 (TRNG)
- 公钥加速器 (PKA)

1.2.5. 主要参数

表 1.1 主要参数说明

模块型号	JX-12F
封装	SMD23/焊盘可焊接插针
尺寸	24*16*3.1（±0.2）mm
认证	TBD
Flash	2M Byte
支持接口	UART/GPIO/ADC/DAC/PWM/ I2C/SDIO/SPI/IR 远程
IO 口	16
串口速率	2400/4800/9600/19200/38400/115200/921600 bps ,最高 5Mbps
频谱范围	2400 ~2483.5MHz
天线形式	IPEX-2 代座(JX-12F1)/板载天线(JX-12F2)/外置天线(JX-12F3)
安全性	WPS/WEP/WPA/WPA2 Personal/WPA2 Enterprise/WPA3
供电范围	供电电压 3.0V ~ 3.6V， 供电电流 >500mA
工作温度	-30℃ ~ +85℃
存储环境	-40℃ ~ +90℃

1.3.管脚定义

JX-12F 模组共接出 23 个接口;如图 2.1， 外设包括 16 个 GPIO, 1 个电源输入, 1 个 GND,1 个 RST (复位脚) , 一个 EN (使能脚) , 1 个邮票孔外置天线接口和 2 个预留的空脚组成。

图 2.1 JX-12F1 管脚示意图

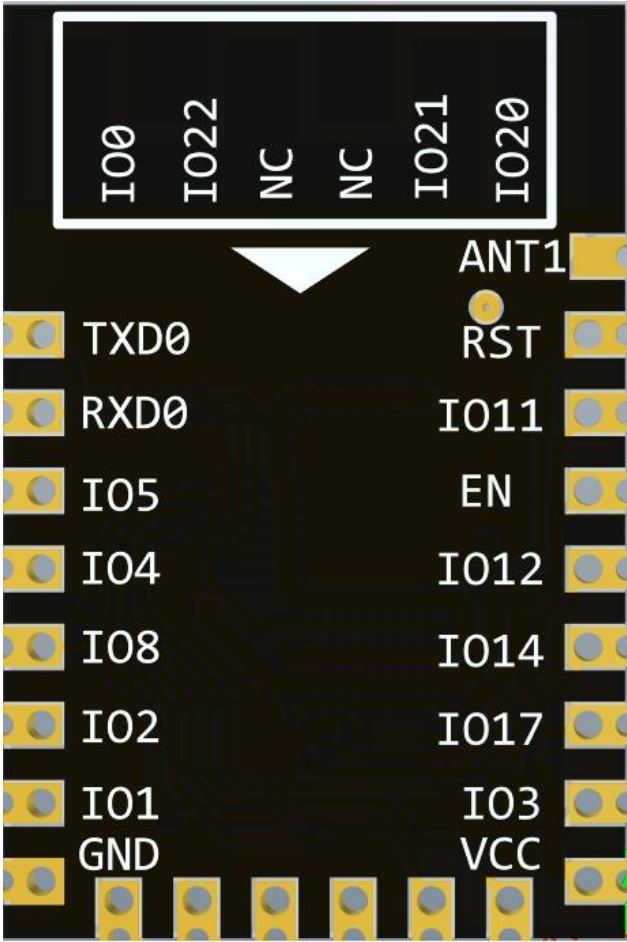


表 2.1 管脚功能定义

序号	Pin 脚名称	功能说明
1	ANT1/NC	JX-12F3 版本的 Wi-Fi/BLE 天线，天线位置需要避空，不能有金属或者铜皮/其它版本为悬空



2	RST	默认悬空（可定位为复位引脚，低电平有效，如有使用，请联系机芯智能
3	IO11	SCLK /SDA /UART/PWM_CH1/FEM3/ ADC_CH10 /IRTX / SWGPIO11/TDO/TDI /IRRX (ir_rx_gpio_sel=1)
4	EN	默认作为芯片使能，高电平有效，与 RST 不能同时使用
5	IO12	MOSI /MISO/SCL/UART/PWM_CH2/ FEM0/ ADC_CH0/ SWGPIO12/ TMS/TCK/ IRRX (ir_rx_gpio_sel=2)
6	IO14	SS/ SCL/ UART/ PWM_CH4/ FEM2/ ADC_CH2 /DAC_B/ SWGPIO14/ TCK/TMS
7	IO17	SF1_D3/MISO /MOSI/ SDA/ UART/ PWM_CH2/ FEM1/ SWGPIO17/ TDI/TDO
8	IO3	DAT1/ SCLK/ SDA/ UART/ PWM_CH3/ FEM3/ SWGPIO3/ TDO/TDI
9	VCC	3.3V 供电，外部电源的供给能力建议大于 500mA
10	IO20	SF1_D0/&SF2_D0/ MOSI /MISO/ SCL/ UART/ PWM_CH0/ FEM0/ SWGPIO20/ TMS/TCK
11	IO21	SF1_CS &SF2_CS/ MISO &MOSI /SDA/ UART/ PWM_CH1 FEM1/ SWGPIO21/ TDI/TDO

12	NC	悬空
13	NC	悬空
14	IO22	SF1_CLK &SF2_CLK/SS/ SCL/ UART/ PWM_CH2/ FEM2/ SWGPIO22/ TCK/TMS
15	IO0	CLK/ SF2_D1/ MOSI &MISO/ SCL/ UART/ PWM_CH0/ FEM0/ SWGPIO0/ TMS/TCK
16	GND	接地
17	IO1	CMD/ SF2_D2/ MISO&MOSI/SDA/ UART/ PWM_CH1/ FEM1/ SWGPIO1/ TDI/TDO
18	IO2	DAT0/SF2_D3/SS/SCL/UART/ PWM_CH2/ FEM2/ SWGPIO2/ TCK/TMS
19	IO8	MOSI &MISO/ SCL/UART/ PWM_CH3/ FEM0 /SWGPIO8 TMS/TCK(该引脚作为 <b>Bootstrap</b> 功能，上电瞬间为高电平时，模组进入烧录模式，上电瞬间为低电平时，模组正常启动)。
20	IO4	DAT2/ MOSI &MISO/SCL/UART/ PWM_CH4/FEM0/ ADC_CH1/ SWGPIO4/ TMS/TCK
21	IO5	DAT3/MISO&MOSI/ SDA/UART/PWM_CH0/ FEM1/ ADC_CH4/ SWGPIO5/ TDI/TDO

22	RXD0	SCLK/SDA/UART/ PWM_CH2/FEM3/ SWGPIO7/ TDO/TDI 默认为模组的烧录口 RX.
23	TXD0	MOSI &MISO/SCL/ UART/ PWM_CH1/ FEM0/ SWGPIO16/ TMS/TCK 默认为模组的烧录口 TX

## 2. 快速上手

智能公元平台（<http://www.smartpi.cn>）已支持配置并生成 JX-12F 模块的固件，在智能公元平台的加持下，JX-12F 具有如下主要特点：

- ✧ 零代码开发 Wi-Fi 产品：在智能公元平台，可快速配置并生成 JX-12 模块的固件，实现远程通讯和远程控制的产品。
- ✧ 支持二进制/JSON/文本协议自定义：串口和通讯接口上的协议可以通过平台配置出来，并可以在多个产品间共享协议，平台具有根据协议配置，自动生成二进制格式、JSON 格式、文本格式协议的代码的能力。具体可参考文档《智能公元平台协议自定义》
- ✧ 拖拽式设计小程序：平台提供了各种按钮、方向盘、进度条、功能列表等多种小程序控件，可拖拽编辑，可设置大小、颜色、图片等风格，所见即所得，可快速完成小程序界面功能的开发。
- ✧ 免费接入机芯智能公元小程序：机芯智能公元小程序，是机芯智能公司为 Wi-Fi/蓝牙产品提供的公版小程序，平台生成的固件，都可以免费接入这个小程序，设备可低成本、低门槛的接入云。
- ✧ 支持自有品牌小程序：如果客户不想使用公版小程序，也可以在智能公元的小程序平台上开通自有品牌的小程序，平台为小程序注册、审核、上线提供了各种功能和便利支持。
- ✧ 小程序支持商城：平台提供了内置的小程序商城功能，客户自有品牌的小程序，可选择开通小程序商城功能，这样小程序除了设备控制功能外，还可以实现销售商品的功能，实现对自有用户群的持续销售和运营。
- ✧ 支持设备管理：平台提供了小程序管理端，客户自有品牌的小程序，可以在小程序管理端完成设备管理，支持设备的接入、删除和升级管理。
- ✧ 支持 OTA：平台的小程序管理端，支持对批量设备进行 OTA 升级；在小程序端，支持通过扫描平台生成固件的二维码，实现单个设备的 OTA 升级。
- ✧ 支持嵌入 C 代码：平台支持嵌入 C 程序代码，实现客户需要的一些灵活控制功能。

### 2.1. 硬件连接

使用 JX-12F 模块开发产品时，可以使用模块，也可以使用开发板，一般建议是使用开发板。使用开发板操作更方便，运行也更稳定，不容易因为连接问题、供电不稳问题影响模块的正常运行，使开发过程更为方便和高效。

#### 2.1.1. 使用模块时的硬件连接

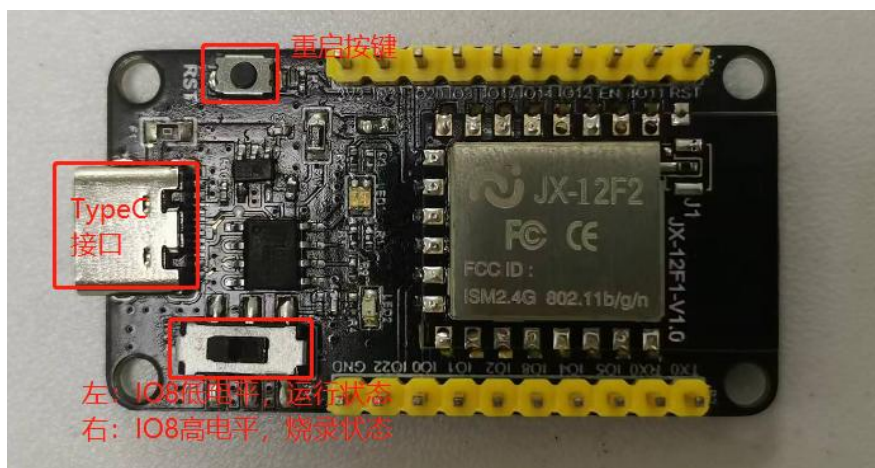
使用模块时，首先要注意的是 JX-12F 模块是 3.3V 供电，如果不小心使用 5V 供电，可能会损坏模块。模块在 Wi-Fi 联网过程中，瞬间功率会变高，要求 3.3V 供电要比较稳定，建议电源线使用焊接方式，不要使用插针方式，插针容易松动导致接触不良，在连接 Wi-Fi 路由器时，会由于供电不稳导致模块重启。下面是常用的模块引脚的连接要求：

- ✧ 模块 VCC：连接 3.3V 供电，要求连接牢固不松动，供电稳定
- ✧ 模块 GND：连接地线，要求连接牢固不松动

- ✧ 模块 TXD0: 连接串口工具的 RX, 用于烧录固件和运行时输出日志
- ✧ 模块 RXD0: 连接串口工具的 TX, 用于烧录固件和运行时输出日志
- ✧ 模块 IO8: 烧录/运行状态切换, 当烧录固件时, 需要将 IO8 连接 3.3V 高电平, 然后重启模块, 模块进入烧录状态; 当模块运行时, 需要将 IO8 悬空或接地, 然后启动模块, 模块将进入运行状态
- ✧ 模块 RST: 模块服务引脚, RST 拉低或接触一下 GND, 模块将会重新启动

## 2.1.2. 使用开发板时的硬件连接

JX-12F 开发板如下图所示:



- ✧ Type C 接口: 连接电脑 USB 接口, 给开发板供电, 同时连接模块 TXD0/RXD0 串口
- ✧ 重启按键 RST: 按一下, 模块将复位重启
- ✧ 烧录/运行模式切换 (跳线或开关): 当 IO8 连接 H (高电平) 时, 按一下“重启按键 RST”, 模块将进入烧录状态; 当 IO8 连接 L (低电平) 时, 按一下“重启按键 RST”, 模块将进入运行状态

## 2.2. 平台配置并生成产品固件

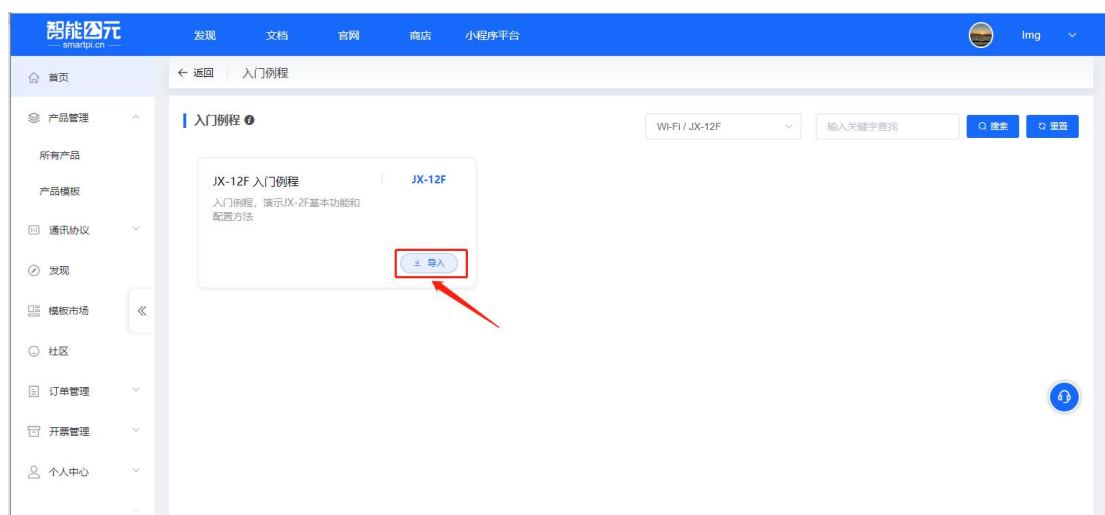
智能公元平台上, 对 Wi-Fi 产品的配置项目比较多, 刚开始使用 JX-12F 模块时, 建议直接导入《入门例程 JX-12F.json》文件, 先了解基本的功能和用法后, 再配置自己的产品, 或者在入门例程的基础上修改, 配置自己的产品。

### 2.2.1. 导入入门例程

打开并登录智能公元平台 (<http://www.smartpi.cn>), 点击“首页”, 找到“JX-12F”模块, 鼠标放这模块上面时, 会显示“入门例程”按钮, 点击这个按钮。可参考下图:



在显示的例程列表中，选择入门例程并点击“导入”按钮，可参考下图：



导入后会自动跳转到产品列表，最上面一行就是新导入的产品，具体参考下图：



## 2.2.2. 继承一个新的版本

在产品列表中，点击“入门例程”的“查看详情”，可进入版本列表，可参考下图：

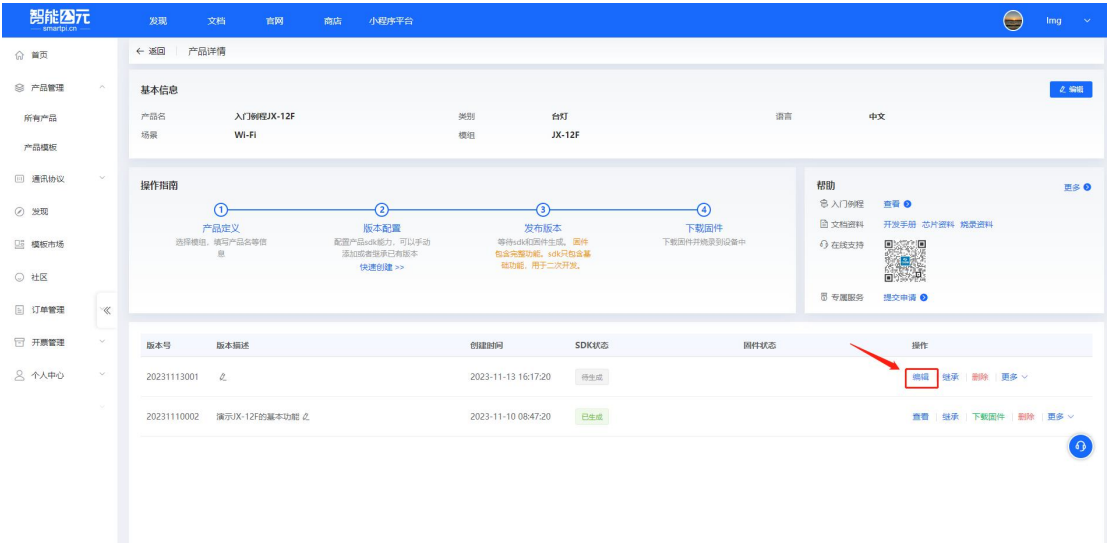


在产品列表中，点击“继承”，并确定后，会继承一个新的版本，新的版本允许编辑并保存，可参考下图：

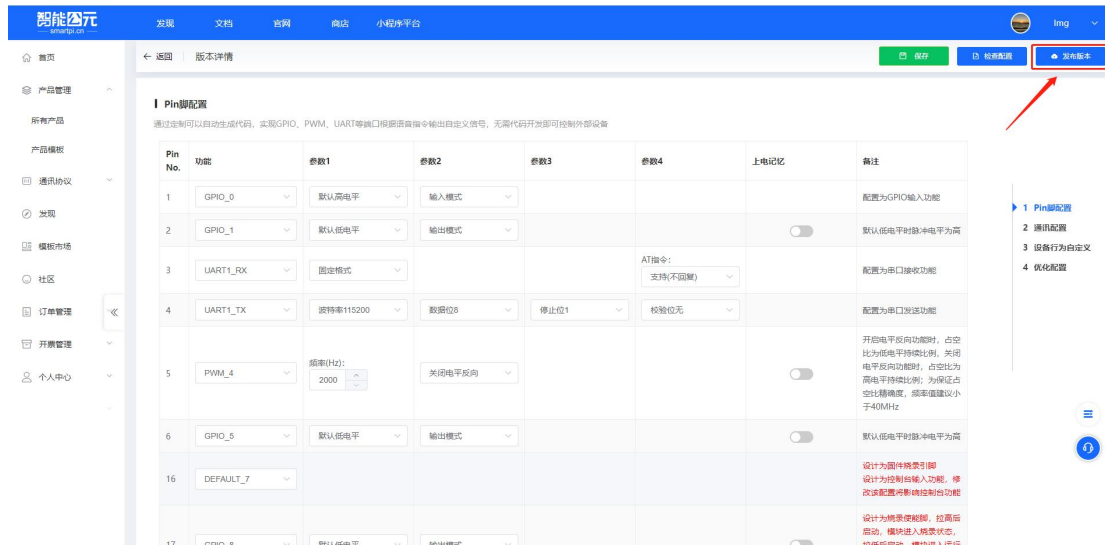


### 2.2.3. 生成新版本固件

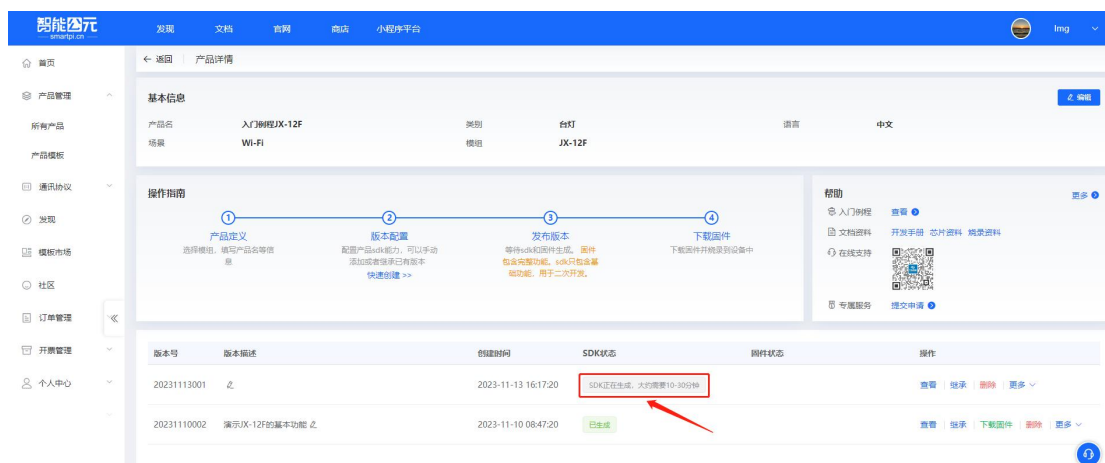
点击新版本上的“编辑”，可进入新版本的配置，可参考下图：



在产品配置界面，可查看并修改配置，修改完成，可点击右上角“检查配置”按钮，检查是否有配置错误或告警，最后点击“发布版本”按钮，生成固件。可参考下图：

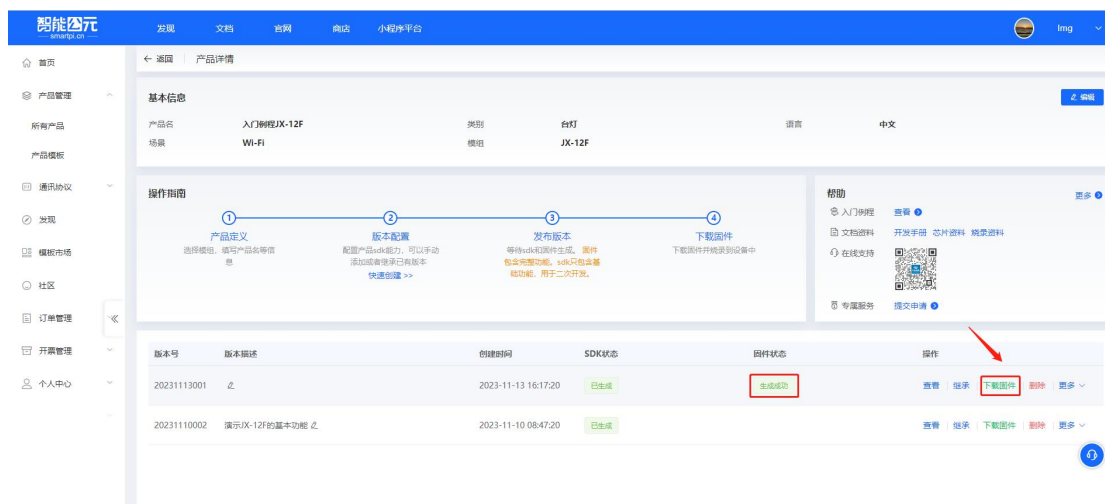


等待平台生成固件，可参考下图：



## 2.2.4. 下载固件包文件

固件生成成功，点击“下载固件”，可将固件包下载到本地。可参考下图：





## 2.3. 使用串口烧录固件

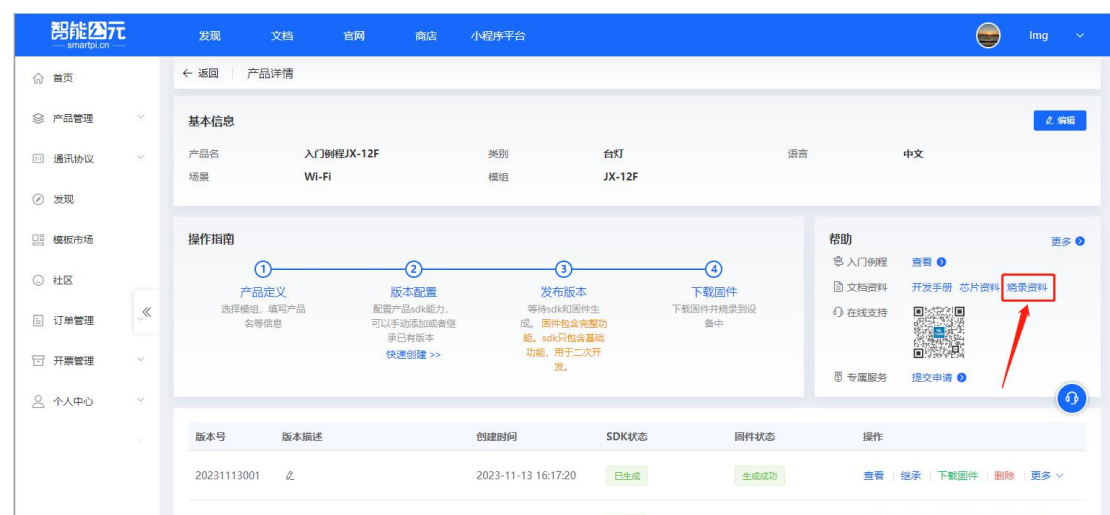
固件包下载到本地的文件名一般为 jx\_firm.tar.gz，是一个压缩包文件，需要用 WinRAR 或其它工具解压缩，解压后的文件结构如下：

名称	修改日期	类型	大小
uart	2023/11/13 16:29	文件夹	
chip_factory_params_IoTKitA_40M.dts	2023/11/13 16:24	DTS 文件	11 KB
chipsp_boot2_release.bin	2023/11/13 16:24	BIN 文件	46 KB
jx_12f_firmware.bin	2023/11/13 16:24	BIN 文件	663 KB
jx_12f_firmware.bin.ota	2023/11/13 16:24	OTA 文件	668 KB
jx_iot_protocol.docx	2023/11/13 16:24	DOCX 文档	26 KB
ota_cfg.json	2023/11/13 16:24	JSON 源文件	1 KB
partition_cfg_2M_ota.toml	2023/11/13 16:24	Toml 源文件	2 KB
properties_action_map.txt	2023/11/13 16:24	文本文档	1 KB
readme.txt	2023/11/13 16:24	文本文档	2 KB

文件说明如下：

- ✧ jx\_12f\_firmware.bin：通过串口烧录的固件文件
- ✧ chip\_factory\_params\_IoTKitA\_40M.dts：通过串口烧录的设备树文件
- ✧ chipsp\_boot2\_release.bin：通过串口烧录的启动文件
- ✧ partition\_cfg\_2M\_ota.toml：通过串口烧录的分区文件
- ✧ jx\_12f\_firmware.bin.ota：OTA 升级时使用的固件文件
- ✧ ota\_cfg.json：在小程序管理端进行 OTA 升级时，需要上传的 OTA 配置文件
- ✧ readme.txt：固件包说明文件
- ✧ jx\_iot\_protocol.docx：机芯 IOT 协议说明文件
- ✧ properties\_action\_map.txt：IOT 属性和行为/变量的绑定关系文件

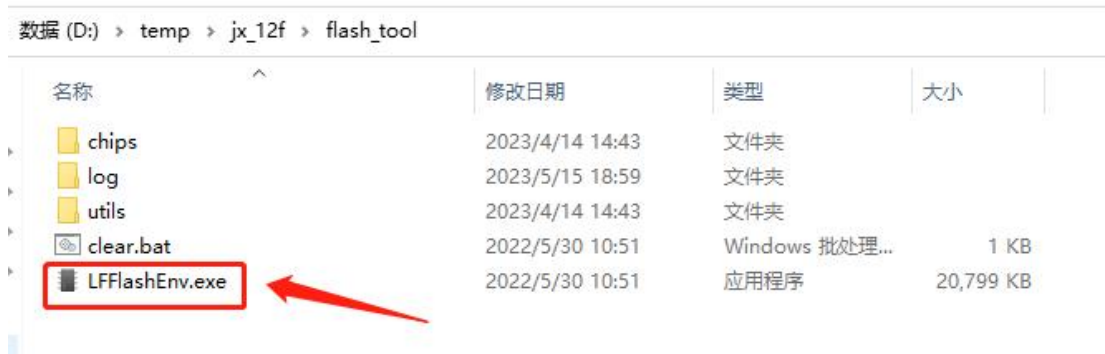
首次串口烧录固件时，需要下载固件烧录工具，下载入口参考下图：



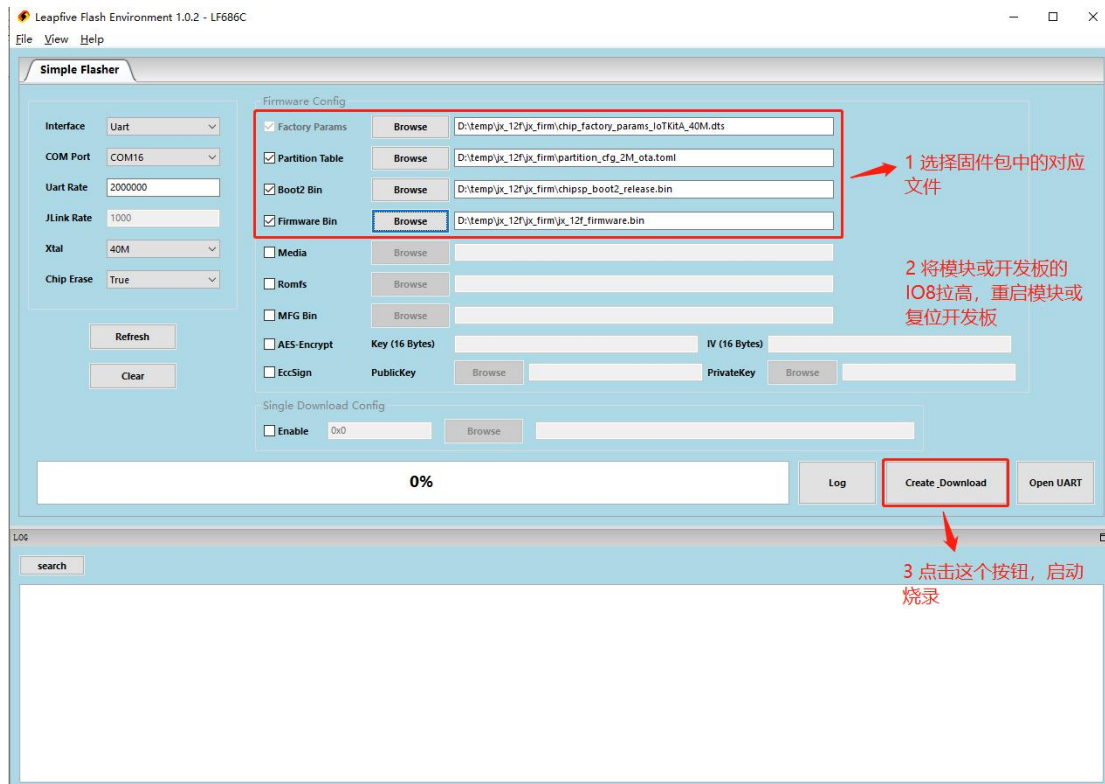
参考下图操作，下载烧录工具：



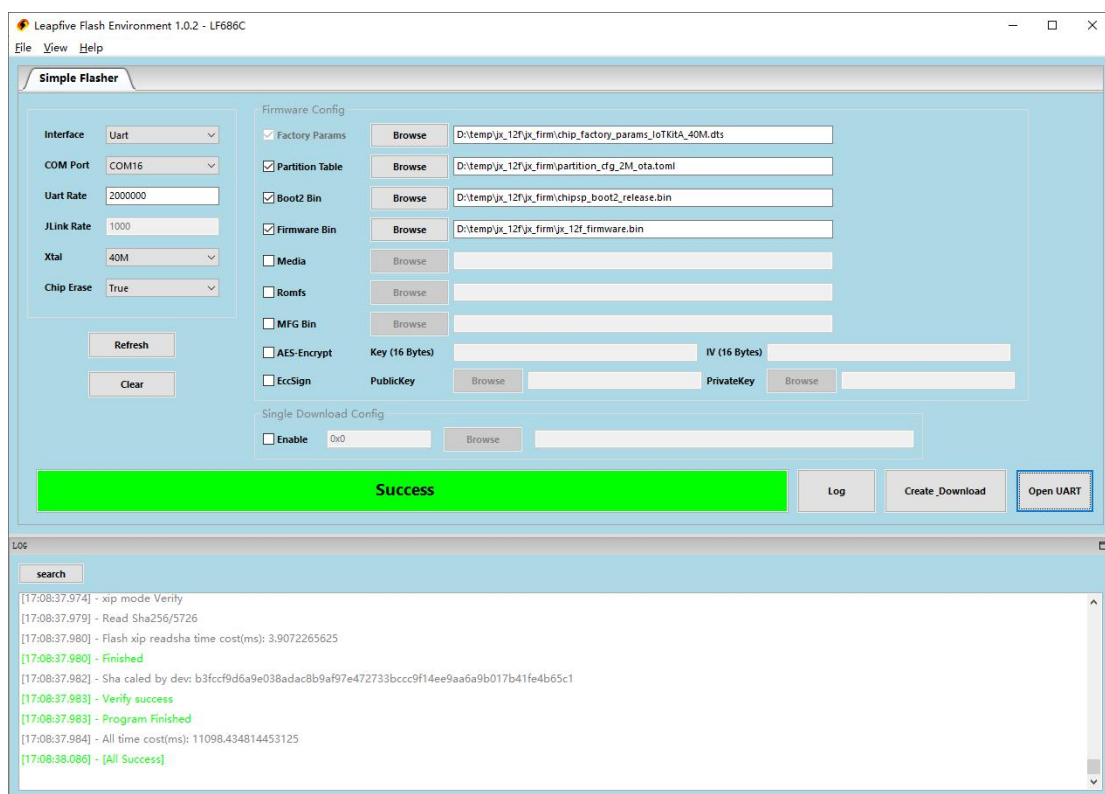
将下载的烧录工具包解压缩，启动烧录工具，参考下图：



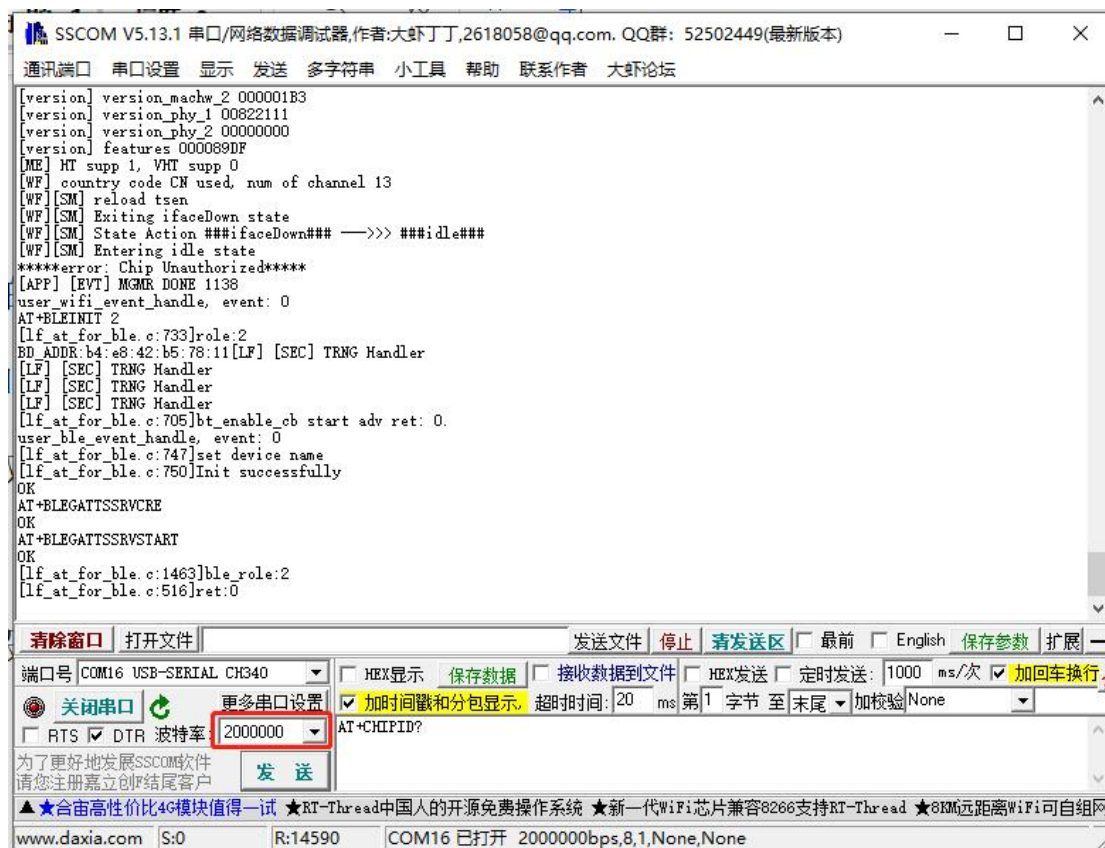
参考下图选择烧录文件（注意正确选择下载的固件包中的对应文件），并按图中步骤操作，进行固件烧录：



烧录成功后，显示如下：



固件烧录完成后，将 IO8 拉低或接地，重新启动模块或开发板，Wi-Fi 模块将进入运行状态。在模块的 TXD0/RXD0 上，用串口工具打开，波特率为 2000000，可以观察到模块的运行日志，参考下图：



## 2.4. 使用小程序配网并添加设备

先在手机上打开蓝牙开关，然后在手机微信中，搜索“机芯智能公元”小程序，或者用微信扫描下面的小程序二维码，进入智能公元小程序：



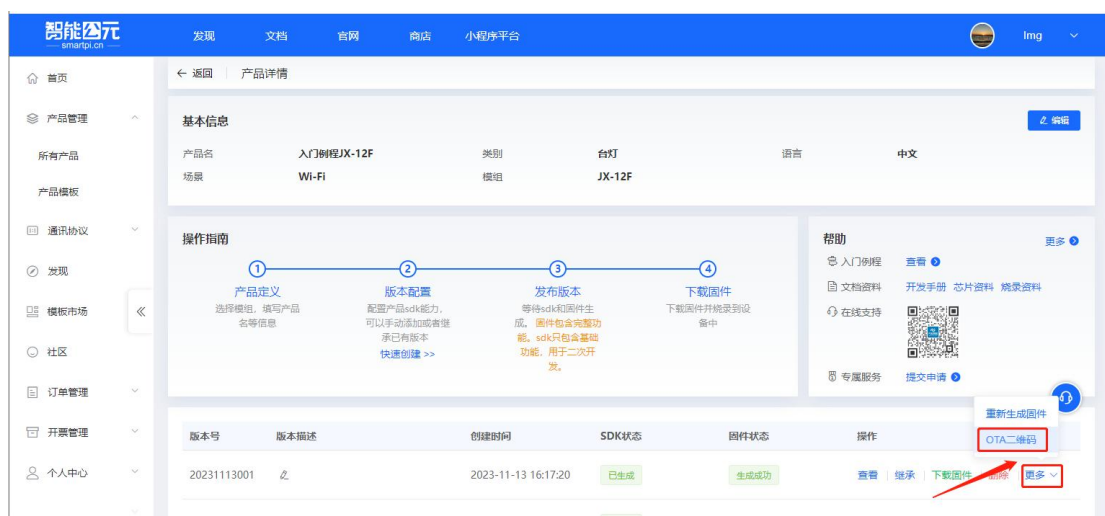
进入小程序后，需要进行必要的注册和授权。完成授权后，点击“首页”中的“+”，可参考下图过程，完成设备配网和设备添加：

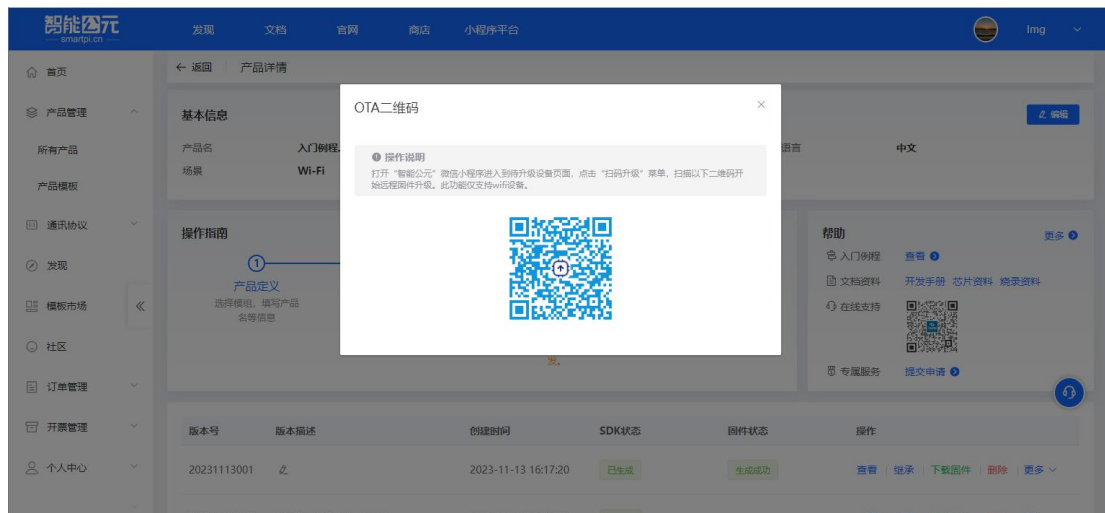




## 2.5. 使用小程序扫码 OTA 升级

首先要求设备在小程序中已经完成配网并添加设备，然后在智能公元平台上，进入产品详情，在需要 OTA 的版本上，点“更多”，“OTA 二维码”，可以打开已生成固件的版本的 OTA 二维码，可参考下图：





打开小程序，进入设备控制界面，点击设备名字，在弹出的菜单中，选择“扫码升级”，扫描平台上的 OTA 二维码，可进行升级。参考下图操作过程：



OTA 扫码升级可以在产品开发和测试过程中使用，提高开发过程的效率。



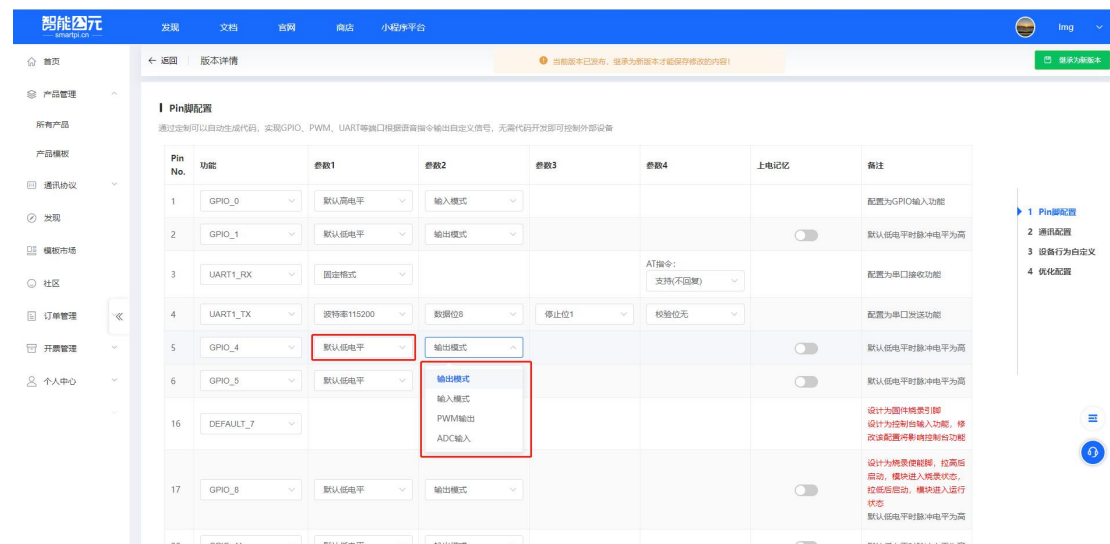
## 3. PIN 脚配置

### 3.1. GPIO 配置

JX-12F 模块支持最多配置 16 个 GPIO，GPIO 可以配置成默认是高电平还是低电平，还可以指定工作模式：

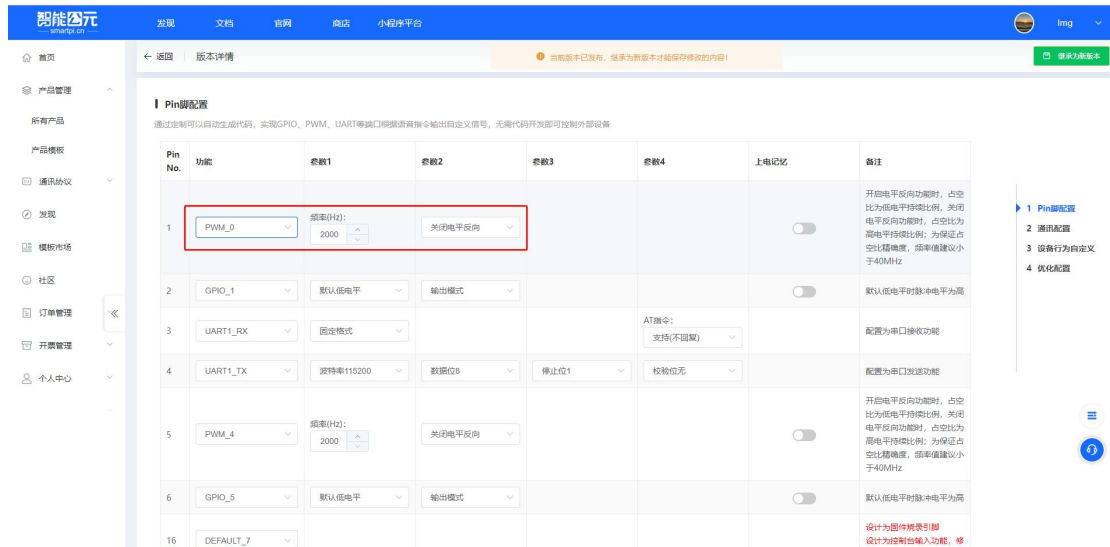
- ✧ 输出模式：可以控制输出高电平或低电平，可以控制输出脉冲
- ✧ 输入模式：可以配合“GPIO 输入”触发行为，实现 GPIO 口电平从高到低或从低到高切换时执行指定的控制动作
- ✧ PWM 输出：模拟 PWM 输出，当硬件 PWM 数量不够时，可以用 GPIO 口来模拟 PWM，但支持的频率比较低，建议频率不要超过 1000
- ✧ ADC 输入：可以配合“ADC 输入”触发行为，当 IO 口电压切换到指定的取值附近时，执行指定的控制动作。电压 0 到 3.3V，对应的 ADC 取值为 0 到 4096。

GPIO 配置可参考下图：



### 3.2. PWM 配置

JX-12F 最多支持 PWM\_0/PWM\_1/PWM\_2/PWM\_3/PWM\_4 共 5 路 PWM，PWM 的配置参考下图：



### 3.3. 串口配置

JX\_12F 最多支持 2 个串口 UART0 和 UART1，其中 UART0 一般用作烧录口，运行时作为控制台，用于日志输出和命令输入，默认配置为 DEFAULT\_7 和 DEFAULT\_16，不建议修改为其它功能，以免影响控制台功能。UART1 可选配置到多个引脚上。

串口支持接收如下格式的消息：

- ✧ 固定格式：由机芯智能定义的消息格式（16 进制数）：AA 55 x1 xn 55 AA，其中 AA 55 为帧头，55 AA 为帧尾，x1 为消息号，xn 为参数，参数可以有多个。
- ✧ 配置格式：和固定格式的区别是允许用户设置帧头和帧尾（也可不要帧尾），其它和固定格式一样
- ✧ 自选协议：使用用户在智能公元平台上自定义的协议，具体参考文档《智能公元平台协议自定义》

串口上也可以配置 AT 指令的如下方式：

- ✧ 不支持：不支持串口执行 AT 指令
- ✧ 支持（不带回复）：支持串口接收并执行 AT 指令，但不回复执行的结果
- ✧ 支持（带回复）：支持串口接收并执行 AT 指令，并回复执行的结果

串口配置可参考下图：



智能图元

发现 文档 官网 商店 小程序平台

img

首页

产品管理

所有产品

产品模板

通信协议

发现

模拟市场

社区

订单管理

开票管理

个人中心

返回 版本详情

当前版本已发布，请求为新版本才能保存修改的内容！

保存为新版本

4	GPIO_3	默认低电平	输出模式				默认低电平时脉冲电平为高
5	UART1_RX	固定格式			AT指令: 不支持		配置为串口接收功能
6	UART1_TX	固定格式					配置为串口发送功能
16	DEFAULT_7	配置格式	数据位8	停止位1	校验位无		设计为固件烧录引脚，设计为控制台输入功能，修改该配置将影响控制台功能
17	GPIO_8	默认低电平	输出模式				设计为情景唤醒脚，拉高后启动，模块进入烧录状态，拉低后启动，模块进入运行状态
20	GPIO_11	默认低电平	输出模式				默认低电平时脉冲电平为高
21	PWM_2	频率(Hz): 2000	关闭电平反向				开电平反向功能时，占空比为低电平持续比例，关闭电平反向功能时，占空比为高电平持续比例；为保证占空比精度，频率值建议小于40KHz
22	GPIO_14	默认低电平	输出模式				默认低电平时脉冲电平为高
27	DEFAULT_16	配置格式					设计为固件烧录引脚，设计为控制台输出功能，修改该配置将影响控制台功能
28	GPIO_17	默认低电平	输出模式				默认低电平时脉冲电平为高

1 Pin脚配置

2 通信配置

3 设备行为自定义

4 优化配置

UART1支持接收的消息格式配置

UART1支持AT指令的方式配置

不建议修改这个配置

不建议修改这个配置

## 4. 通讯配置

### 4.1. 通讯接口配置

平台支持配置下面类型的通讯接口：

- ✧ TCP
- ✧ UDP
- ✧ MQTT
- ✧ BLE

通讯接口支持接收如下格式的消息：

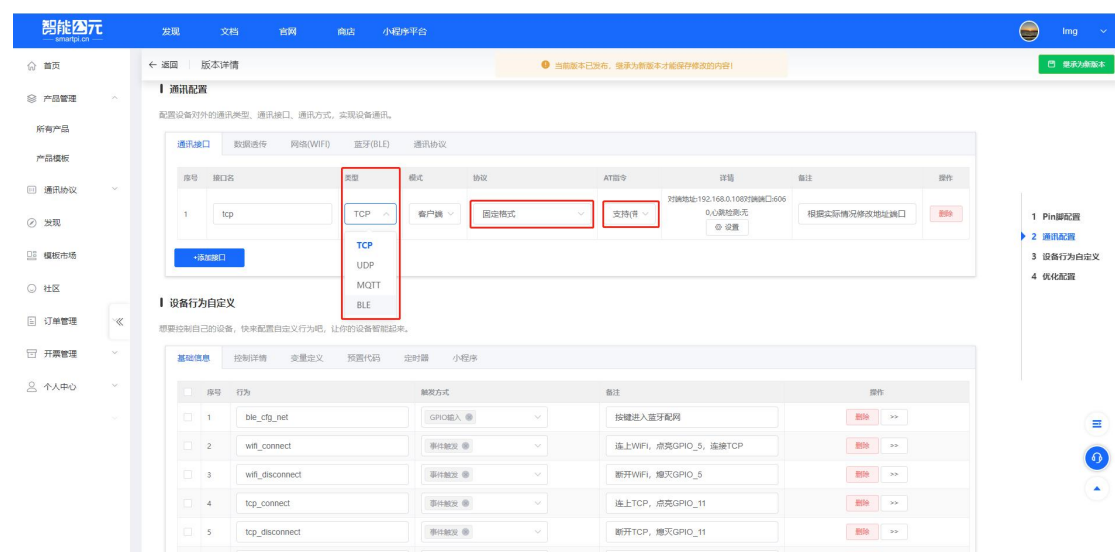
- ✧ 固定格式：由机芯智能定义的消息格式（16 进制数）：AA 55 x1 xn 55 AA，其中 AA 55 为帧头，55 AA 为帧尾，x1 为消息号，xn 为参数，参数可以有多个。
- ✧ 配置格式：和固定格式的区别是允许用户设置帧头和帧尾（也可不要帧尾），其它和固定格式一样
- ✧ 自选协议：使用用户在智能公元平台上自定义的协议，具体参考文档《智能公元平台协议自定义》

通讯接口可以配置 AT 指令的如下支持方式：

- ✧ 不支持：不支持串口执行 AT 指令
- ✧ 支持（不带回复）：支持串口接收并执行 AT 指令，但不回复执行的结果
- ✧ 支持（带回复）：支持串口接收并执行 AT 指令，并回复执行的结果

如果配置中打开了支持微信小程序的开关，平台会自动打开和小程序通讯的接口，不需要在通讯接口中配置和小程序通讯的接口。

通讯接口配置参考下图：

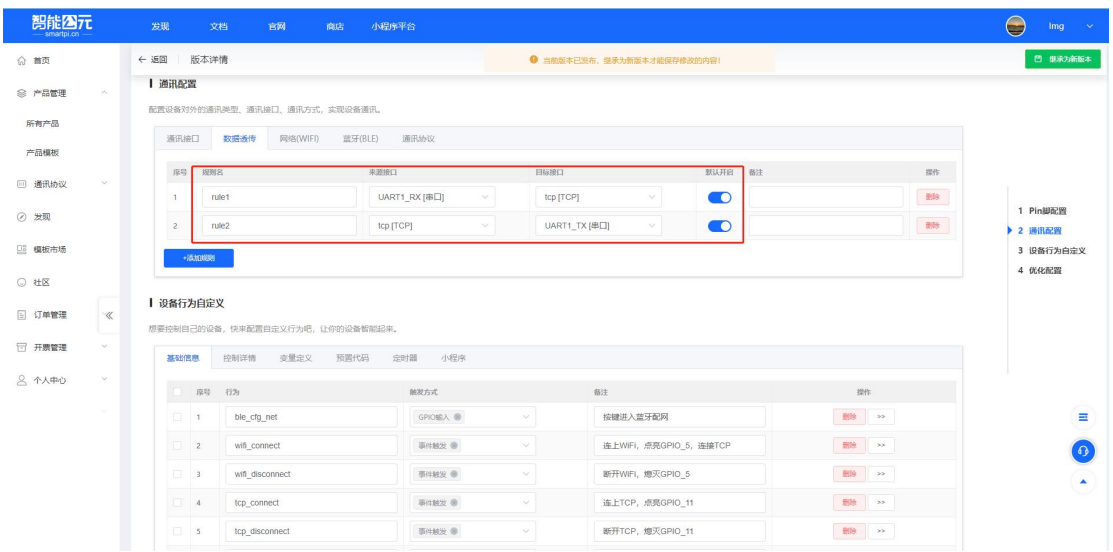


## 4.2. 数据透传配置

平台支持在通讯接口之间，通讯接口和串口之间配置数据透传规则，数据透传规则可以在行为控制中动态开启和关闭。通讯接口和串口收到消息后，一般会按如下过程处理：

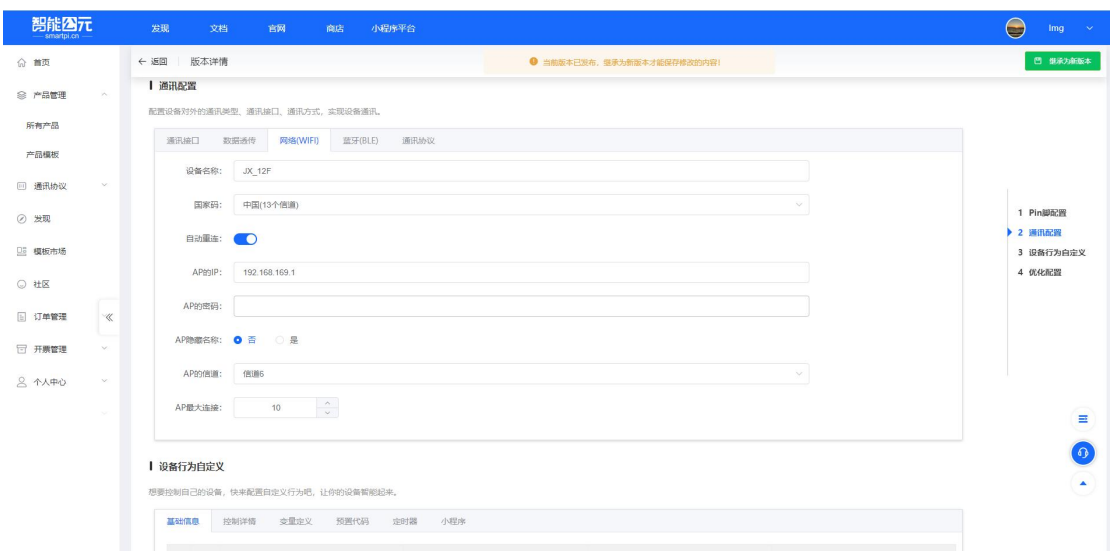
- 判断通讯接口或串口是否支持 AT 指令，如果支持 AT 指令，分析消息是否 AT 指令格式，如果是 AT 指令，则执行 AT 指令，处理过程结束。如果非 AT 指令格式，转下一步处理。
- 判断消息是否匹配“通讯输入”触发或“串口输入”触发的行为，如果匹配，则执行匹配的行为控制，处理过程结束。如果不匹配，转下一步处理。
- 判断是否存在该消息来源接口的数据透传规则，如果存在并且规则启用，则将消息转发到数据透传的目标接口，处理过程结束。

数据透传配置参考下图：



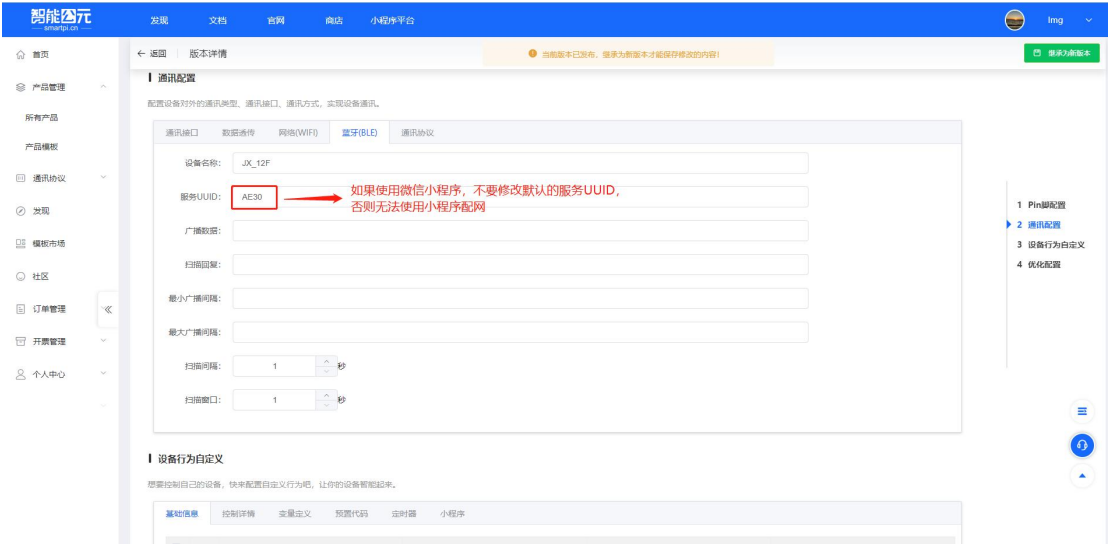
## 4.3. 网络（WIFI）配置

网络（WIFI）配置参考下图：



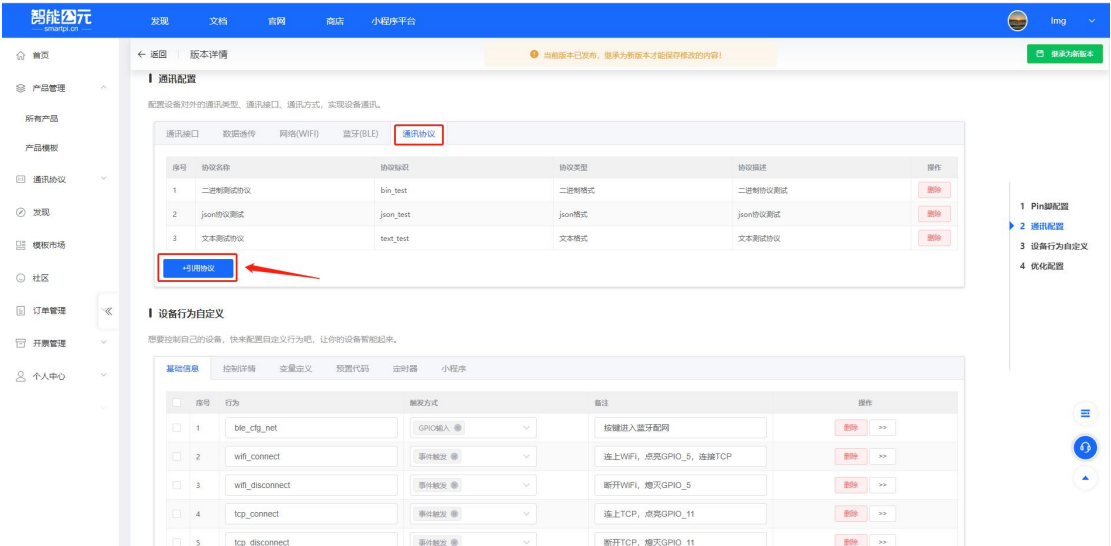
## 4.4. 蓝牙（BLE）配置

蓝牙（BLE）配置中，如果使用微信小程序，则不要修改“服务 UUID”的默认值，否则无法通过小程序配网。可参考下图：



## 4.5. 通讯协议配置

如果产品需要使用自定义协议，则需要在“通讯协议配置”中，引入在平台上已经配置好的协议，具体可参考文档《智能公元平台协议自定义》。可参考下图：



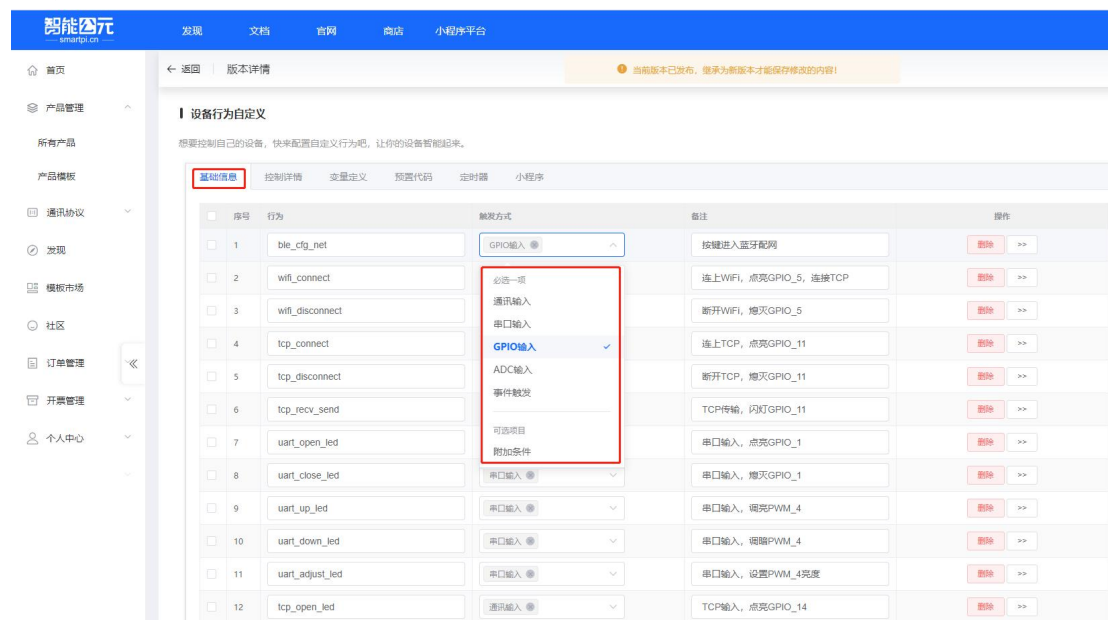
## 5. 行为触发配置

平台支持如下行为触发方式：

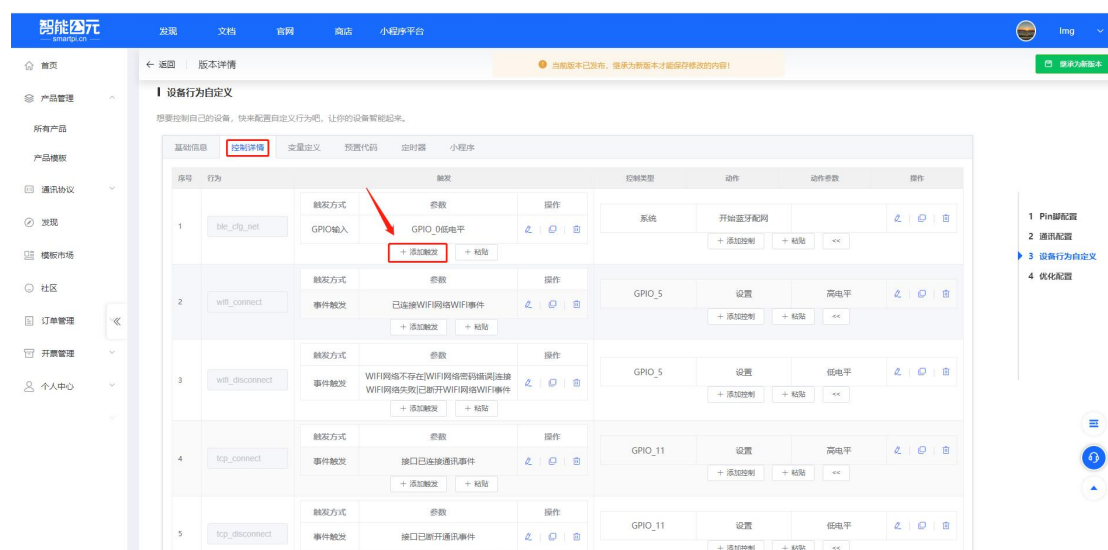
- ✧ 通讯输入：通讯接口上接收并匹配到预期消息，触发对应的行为
- ✧ 串口输入：串口上接收并匹配到预期消息，触发对应的行为
- ✧ GPIO 输入：输入模式的 GPIO 口发生电平变化，触发对应的行为
- ✧ ADC 输入：ADC 输入模式的 GPIO 口的电压（0 到 3.3V）变化到目标值附近（0 到 3.3V 对应取值 0 到 4096），触发对应的行为
- ✧ 事件触发：系统内部发生某些事件时，比如联网或断网事件等，触发对应的行为

同时支持行为触发时“附近条件”，只要满足附加条件要求才能触发行为，附加条件可以是变量值，也可以是 IO 口状态，或者它们的组合。

在“设备行为自定义”中，在“基础消息”中添加行为后，需要指定行为的触发方式，参考下图：



然后在“控制详情”中，可以在行为上，点击“添加触发”按钮，设置触发的内容：



# 5.1. 通讯输入触发

如果通讯接口的协议为“固定格式”或“配置格式”，则添加触发的界面如下图，可手工添加参数，测试消息可以用于接口模拟测试：

添加触发

行为: tcp\_open\_led

触发方式: 通讯输入

选择接口: tcp [TCP]

消息编号: 1

输入参数:

参数名	类型	测试值	操作
arg1	char	1	删除

添加

测试消息: AA 55 01 01 55 AA

16进制数，AA 55为帧头，55 AA为帧尾

取消

确定

如果通讯接口的协议为“自选协议”，则添加触发的界面如下图，可选择“协议”、“消息”、“模板”，并根据模板自动生成参数列表：

添加触发

×

行为: tcp\_open\_led

触发方式:

通讯输入

▼

选择接口:

tcp [TCP]

▼

选择协议:

二进制测试协议

▼

选择消息:

msg1

▼

选择模板:

tpl\_test1

▼

模板参数:

参数名	类型	测试值
MsgNo	char	<input type="text"/>
ArgInt	int	<input type="text"/>

取消

确定

在行为的控制中，可以通过“变量设置”将接收到的参数值传递给变量，变量可以用于其它控制。

## 5.2. 串口输入触发

如果串口的协议为“固定格式”或“配置格式”，则添加触发的界面如下图，可手工添加参数，测试消息可以用于接口模拟测试：

添加触发

×

行为: uart\_open\_led

触发方式: 串口输入

选择串口: UART1\_RX

消息编号: 1

输入参数:

参数名	类型	测试值	操作
arg1	char	1	删除

添加

测试消息: AA 55 01 01 55 AA  
16进制数, AA 55为帧头, 55 AA为帧尾

取消

确定

如果串口的协议为“自选协议”，则添加触发的界面如下图，可选择“协议”、“消息”、“模板”，并根据模板自动生成参数列表：

添加触发

×

行为: uart\_open\_led

触发方式: 串口输入

选择串口: UART1\_RX

选择协议: 二进制测试协议

选择消息: msg1

选择模板: tpl\_test1

模板参数:

参数名	类型	测试值
MsgNo	char	
ArgInt	int	

取消

确定



在行为的控制中，可以通过“变量设置”将接收到的参数值传递给变量，变量可以用于其它控制。

### 5.3. GPIO 输入触发

在添加“GPIO 输入”触发时，需要有 GPIO 口设置为输入模式，可选“低电平”或“高电平”触发，参考下图：



### 5.4. ADC 输入触发

在添加“ADC 输入”触发时，需要有 GPIO 口设置为 ADC 输入模式，0 到 3.3V 电压对应取值 0 到 4096，当电压对应取值切换到目标取值附近时，会触发相应行为。参考下图：



## 5.5. 事件触发

事件触发时，支持下列类型的触发事件：

✧ 系统启动，事件原因包括：

- 上电启动
- 软件重启
- 看门狗重启

✧ 定时器超时

✧ 蓝牙事件，事件原因包括：

- 蓝牙广播开始
- 蓝牙广播停止
- 蓝牙扫描开始
- 蓝牙扫描停止

✧ WIFI 事件，事件原因包括：

- WIFI 初始化完成
- 已连接 WIFI 网络
- 已断开 WIFI 网络
- 未配置 WIFI 网络
- WIFI 网络不存在
- WIFI 网络密码错误
- 连接 WIFI 网络失败
- AP 已启动
- AP 已停止
- AP 接入设备
- AP 断开设备

✧ 通讯事件，事件原因包括：

- 接口已连接
- 接口已断开
- 接口连接失败
- 接口发送数据
- 接口接收数据

事件原因可以多选，如果不选事件原因，则任何一种事件原因都会触发。

参考下图设置：



## 5.6. 触发附加条件

触发附加条件可以是 IO 口的状态，也可以是变量取值，或者是两者的组合，可参考下图设置：



## 6. 行为控制配置

在“控制详情”中，可以在行为上“添加控制”，一个行为的控制条数没有限制。控制方式包括：

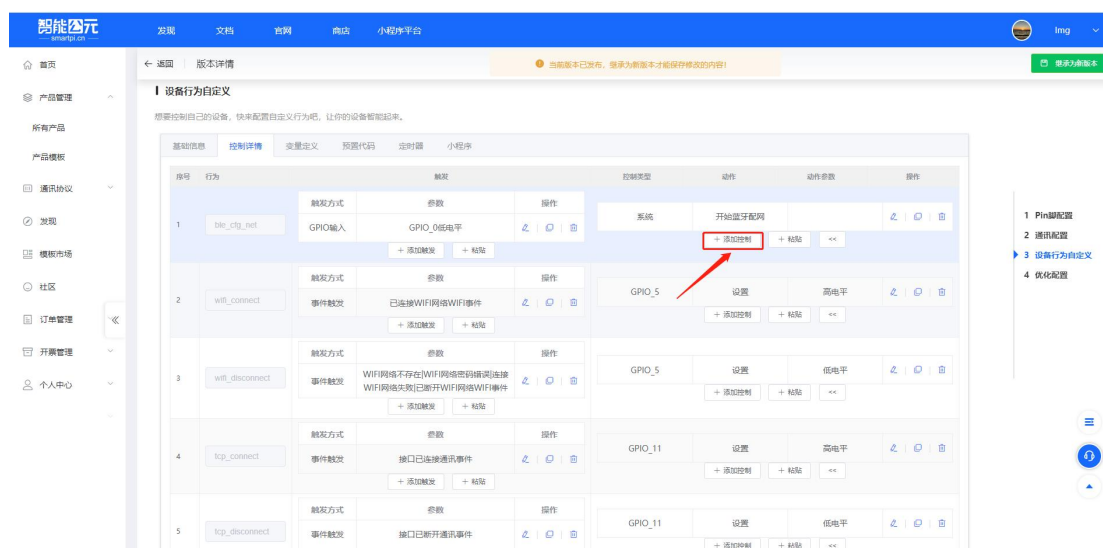
- ✧ 端口输出：控制 GPIO、PWM、串口输出
- ✧ 系统设置：包括 WIFI、蓝牙、定时器、AT 指令、重启、内部控制等
- ✧ 变量设置：变量的设置、增加、减少
- ✧ 通讯控制：连接接口、发送消息、关闭接口、开始透传、停止透传等

行为控制支持设置“执行条件”，如果设置了条件，只有满足条件，控制才会被执行。

行为控制支持设置“延时执行”，如果设置了延时，所有延时控制的开始延时时间，都是从触发这个行为开始，而不是从上一个控制执行完开始，这一点需要注意。

部分类型的行为控制，支持设置“延时恢复”，延时恢复的开始延时时间，是从这个控制执行完开始算的。

添加控制可参考下图：



常用的设置参考下图：



## 6.1. GPIO 端口输出

添加控制时，选择控制方式为“端口输出”，选择对应的 GPIO 口，可以执行控制：

✧ 高电平

- ✧ 低电平
  - ✧ 反电平：将 GPIO 设成当前电平相反的电平
  - ✧ 脉冲输出
- 可参考下图：

添加控制

行为: tcp\_disconnect

是否条件执行: ☐ 是 ☒ 否

控制方式: 端口输出

控制类型: GPIO\_11

是否延时输出: ☐ 是 ☒ 否

动作: 设置电平

参数: 低电平

延时电平翻转: ☐ 是 ☒ 否

取消 确定

## 6.2. PWM 端口输出

添加控制时，选择控制方式为“端口输出”，选择对应的硬件 PWM 口或者模拟 PWM 口，可以执行控制：

- ✧ 设置占空比
- ✧ 增加占空比
- ✧ 减少占空比

可参考下图：

添加控制

行为: uart\_adjust\_led

是否条件执行: ☐ 是 ☒ 否

控制方式: 端口输出

控制类型: PWM\_4

是否延时输出: ☐ 是 ☒ 否

动作: 设置占空比

方式: 变量

变量: pwm4\_duty

延时恢复: ☐ 是 ☒ 否

取消 确定

## 6.3. 串口输出

添加控制时，选择控制方式为“端口输出”，选择对应的串口，可以执行控制：

- ✧ 发送 16 进制数：直接填写 16 进制的数，进行发送
  - ✧ 发送协议消息：如果在通讯配置中引用了自定义的协议，则可以发送自定义的协议消息，具体可参考《智能公元平台协议自定义》
- 发送 16 进制数可参考下图：

添加控制

行为: wifi\_disconnect

是否条件执行: ☐ 是 ☒ 否

控制方式: 端口输出

控制类型: UART1\_TX

是否延时输出: ☐ 是 ☒ 否

动作: 发送16进制数

参数: 11 22 33

格式为十六进制0-9和A-F, 最大长度32字节

不能出现连贯多字符连接, 如010809

两个字符一组, 中间为空格 例如: 98 A3 0B FE

取消

确定

发送协议消息时, 会根据选择的协议、消息、模板自动生产参数列表, 参数取值可以填数值, 也可以填变量。发送协议消息可参考下图:

添加控制

行为: wifi\_disconnect

是否条件执行: ☐ 是 ☒ 否

控制方式: 端口输出

控制类型: UART1\_TX

是否延时输出: ☐ 是 ☒ 否

动作: 发送协议消息

协议: 二进制测试协议

消息: msg1

模板: tpl\_test1

参数:

参数名	类型	模式	取值
MsgNo	char	数值	1
ArgInt	int	数值	2

取消

确定

## 6.4. 通讯控制输出

添加控制时，选择控制方式为“通讯输出”，可以执行如下控制：

- ✧ 连接接口：连接到指定的接口
  - ✧ 发送 16 进制数：直接填写 16 进制的数，进行发送
  - ✧ 发送协议消息：如果在通讯配置中引用了自定义的协议，则可以发送自定义的协议消息，具体可参考《智能公元平台协议自定义》
  - ✧ 关闭接口：关闭指定的接口
  - ✧ 开始透传：打开指定的数据透传规则
  - ✧ 停止透传：停止指定的数据透传规则
- 发送 16 进制数可参考下图：



添加控制

行为: wifi\_disconnect

是否条件执行: ☐ 是 ☒ 否

控制方式: 通讯控制

是否延时设置: ☐ 是 ☒ 否

操作: 发送16进制数

通讯接口: tcp [TCP]

发送数据: 11 22 33

格式为十六进制0-9和A-F, 最大长度32字节

不能出现连贯多字符连接, 如010809

两个字符一组, 中间为空格 例如: 98 A3 0B FE

取消

确定

发送协议消息时, 会根据选择的协议、消息、模板自动生产参数列表, 参数取值可以填数值, 也可以填变量。发送协议消息可参考下图:

添加控制

行为: wifi\_disconnect

是否条件执行: ☐ 是 ☒ 否

控制方式: 通讯控制

是否延时设置: ☐ 是 ☒ 否

操作: 发送协议消息

通讯接口: tcp [TCP]

协议: 二进制测试协议

消息: msg1

模板: tpl\_test1

参数:
 

参数名	类型	模式	取值
MsgNo	char	数值	1
ArgInt	int	数值	2

取消

确定

## 6.5. 系统设置

添加控制时，选择控制方式为“系统设置”，可以执行如下类别的控制：

- ✧ WIFI 控制，包括：
  - 连接 WIFI 网络(默认)
  - 连接 WIFI 网络(指定)
  - 断开 WIFI 网络
  - 启动 AP 模式(默认)
  - 启动 AP 模式(指定)
  - 关闭 AP 模式
- ✧ 蓝牙控制，包括：
  - 开始蓝牙配网
  - 停止蓝牙配网
  - 启动蓝牙服务端
  - 停止蓝牙服务端
  - 开始蓝牙广播(默认)
  - 开始蓝牙广播(指定)
  - 停止蓝牙广播

- ✧ AT 指令：执行 AT 指令
- ✧ 定时器控制，包括：
  - 启动定时器
  - 停止定时器
- ✧ 重启控制：软件重启模块
- ✧ 内部控制，包括：
  - 跳转行为：调用执行其它行为
  - 清空密码：如果用户设置了连接密码，可以清空密码
  - 执行代码：可以执行用户输入的 C 代码

可参考下图：

添加控制

行为: wifi\_disconnect

是否条件执行: ☐ 是 ☒ 否

控制方式: 系统设置

是否延时设置: ☐ 是 ☒ 否

类别: 内部控制

操作: WIFI控制  
蓝牙控制  
AT指令  
定时器控制  
重启控制  
内部控制

延时恢复:

取消 确定

## 6.6. 变量设置

添加控制时，选择控制方式为“变量设置”，可以执行如下控制：

- ✧ 设置变量值
- ✧ 增加变量值
- ✧ 减少变量值

可参考下图：

## 添加控制



行为: wifi\_disconnect

是否条件执行: ☐ 是 ☒ 否

控制方式: 变量设置

是否延时设置: ☐ 是 ☒ 否

变量: pwm2\_duty

操作: 请选择

方式: 设置  
增加  
数值: 减少

延时恢复: ☐ 是 ☒ 否

取消

确定

## 7. 小程序配置

点击“小程序”页签，可以配置微信小程序。将“开启”开关打开，可以启用微信小程序。在小程序控件区，提供了如下可用控件：

- ✧ 开关：左右切换开关
- ✧ 按钮：点击按钮
- ✧ 复杂按钮：可提供点击、按下、松开、长按等更多控制
- ✧ 按钮组：多个点击按钮的组合
- ✧ 状态按钮：按下弹起式按钮
- ✧ 数值输入：通过输入数值的功能
- ✧ 数值按键：通过增大、减小、最大、最小按键的方式输入数值
- ✧ 方向按钮：提供左、中、右、上、下布局的五个点击按钮
- ✧ 复杂方向按钮：提供左、中、右、上、下布局的五个复杂按钮
- ✧ 方向盘：提供方向盘布局的五个点击按钮
- ✧ 滑块：通过弹出式滑动条（占半行）和平铺式滑动条（占整行）
- ✧ 功能列表：从屏幕下面弹出的菜单列表
- ✧ 音乐播放：播放音乐风格的按钮组合
- ✧ 圆形进度条：圆形风格的进度条
- ✧ 文本描述：输入静态文本并显示在指定位置

点击控件区的控件，会自动在小程序设计区新增一个控件，在控件设置区可以设置控件的名称、关联的行为或变量、颜色、图片、大小等。可参考下图：



### 7.1. 设备基本信息配置

开启微信小程序后，小程序设计区有默认的“设备页”控件（在设计区顶部），点击“设备页”控件，在右边的控件设置区，可以设置设备的基本信息：

- ✧ 设备名称：小程序上显示的设备名字，设备用户在小程序中也可以改名字
- ✧ 设备 ID：配网时显示的设备名称前缀

- ✧ 对接小程序：如果客户开通了自己的小程序，可以将小程序的 **appid** 填入，设备可接入该小程序。对接的小程序支持多个，可以点下面的“+”号增加
- ✧ 智能公元开关：打开将允许设备免费接入智能公元公版小程序，关闭将不允许设备接入智能公元小程序
- ✧ 设备密码，有两种模式：
  - 后台管理密码：密码在小程序管理后台管理，由管理员设置或修改，适用于需要统一控制设备访问权限的场景
  - 设备本地密码：密码保存在设备 Flash 中，由用户设置，适用于用户自行管理设备访问权限的场景。这种方式通常需要提供密码清空功能，避免用户忘记密码后无法使用。典型的做法是设备提供一个按键，按键触发后使用“系统设置”/“内部控制”/“清空密码”控制功能
- ✧ 设备图标：小程序中显示的设备头像
- ✧ 其它风格设置：颜色、背景图片等可以灵活配置

可参考下图：



## 7.2. 控件关联行为

大部分控件都可以关联行为，简单的只涉及状态切换的控件，比如开关、按钮能，只能关联的是无参数的行为；对于和数值输入相关的控件，比如滑块、数值输入、数值按键等，只能关联的是带有一个参数的行为。

关联行为可参考下图：



## 8. 嵌入 C 代码

平台支持 JX-12F 模块嵌入 C 代码，开发者可以用这种方式实现一些平台没有提供的功能，或者用这种方式来简化平台设置复杂的功能逻辑。提供两种支持方式：

- ✧ 预置代码：开发者可以在预置代码中定义一些函数和变量，这些函数和变量可以在后面执行代码中使用
- ✧ 执行代码：在行为控制中，开发者可以使用自己在预置代码中定义的函数和变量，或者使用平台开放的函数

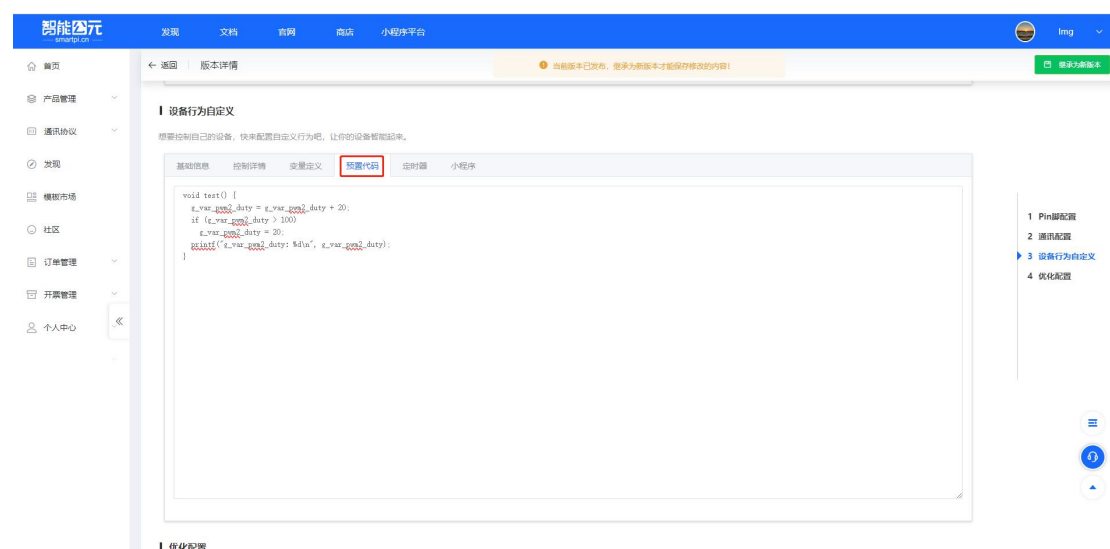
开发者在平台的“变量定义”中定义的变量，在嵌入 C 代码中，可以用“g\_var\_变量名”的方式来使用，比如“变量定义”中的变量名为 `duty`，则在嵌入 C 代码中的变量名为 `g_var_duty`。

平台开放的函数列表另行提供。

嵌入 C 代码功能要小心使用，如果输入的代码有错误，会导致固件生成失败。

### 8.1. 预置代码

在“设备行为自定义”中，选择“预置代码”页签，可以输入 C 代码。参考下图：



### 8.2. 执行代码

在行为中添加控制，选择“系统设置”、“内部控制”、“执行代码”，可以输入要执行的代码。可参考下图：



## 添加控制



行为: ble\_cfg\_net

是否条件执行: ☐ 是 ☒ 否

控制方式: 系统设置

是否延时设置: ☐ 是 ☒ 否

类别: 内部控制

操作: 执行代码

参数: test();

取消

确定