

# 1. 概述

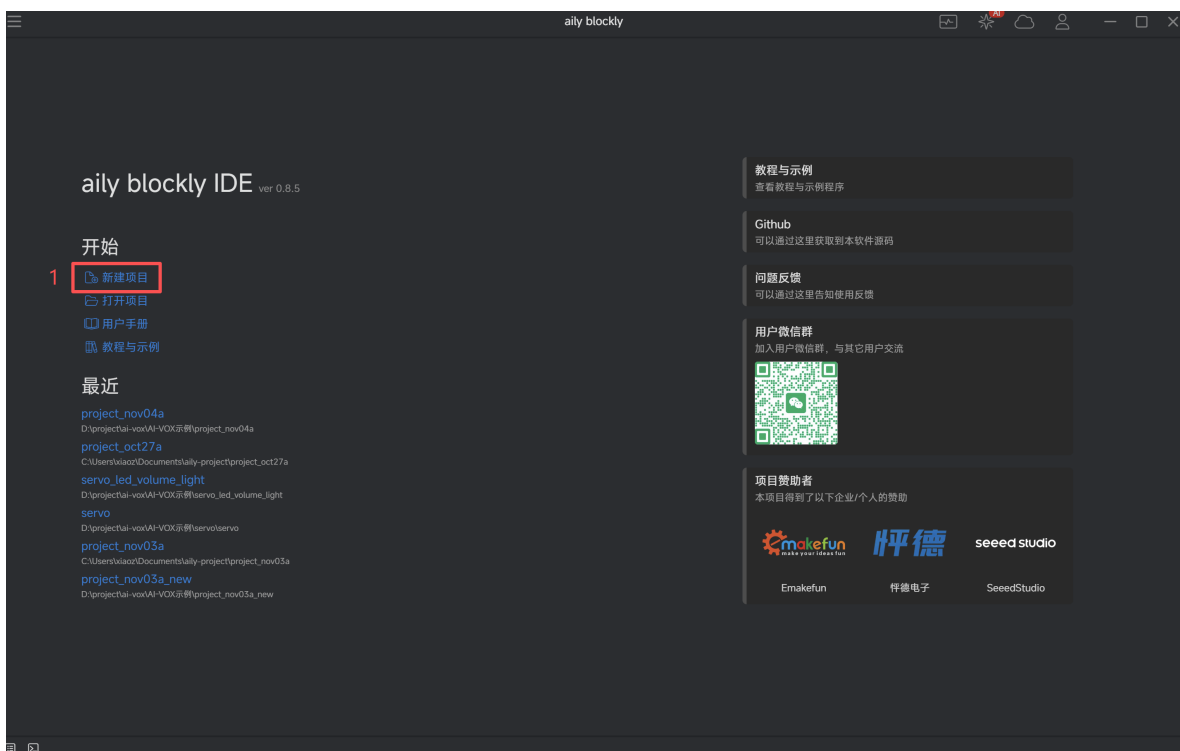
AI-VOX图形化库是采用新一代AI图形化编程平台AilyBlockly专门针对小智AI这类产品而开发的。它基于Arduino版本小智SDK（AI VOX Engine AI语音交互库）二次开发而成。该库适配AI-VOX3等搭载ESP32S3芯片的硬件平台，提供丰富而便捷的功能模块，包括手动配网、音量调节、屏幕显示内容（表情、状态、对话内容）设置、屏幕亮度调整、MCP事件注册，以及MCP外设查询与控制等。用户仅需通过拖拽图形化模块，即可轻松实现复杂的小智AI语音对话及外设控制功能。我们相信，创造的门槛不应高不可攀，别让想法止于想象，选择AI-VOX图形化库，让你彻底告别“从入门到放弃”，你的第一个语音AI交互项目，今天就能轻松开启！

## 2. 创建第一个AI图形化项目

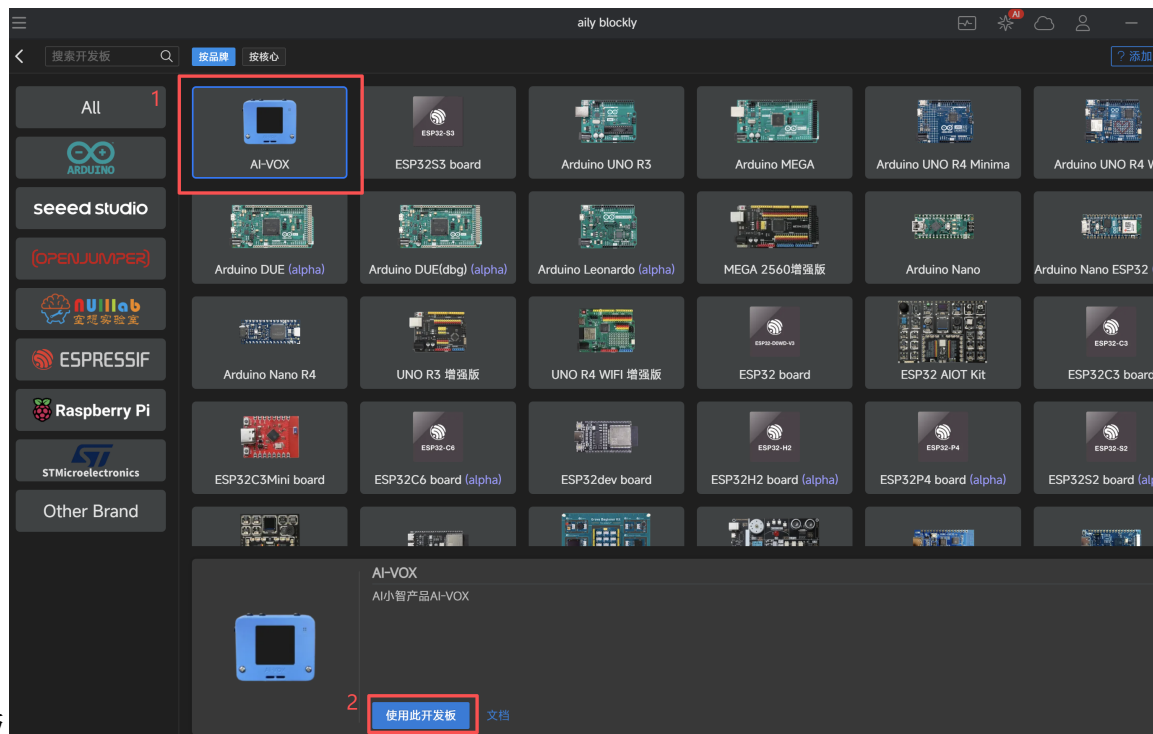
### 2.1 AilyBlockly软件下载

下载地址：<https://arduino.me/a/3093> 将软件安装好，并打开软件。

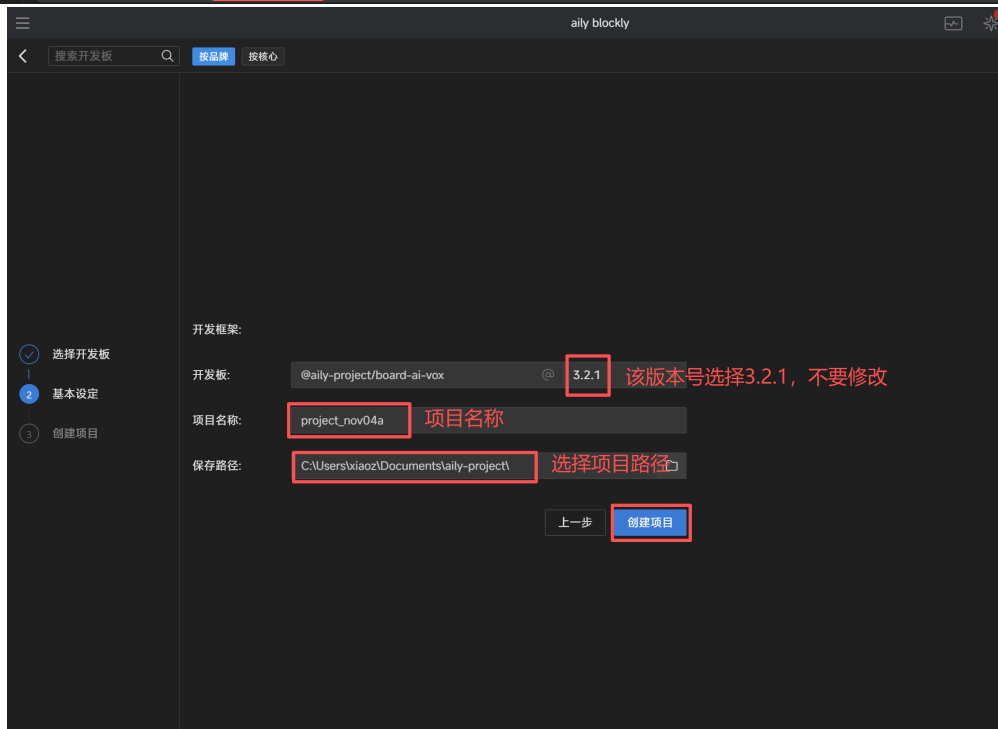
### 2.2 创建第一个项目



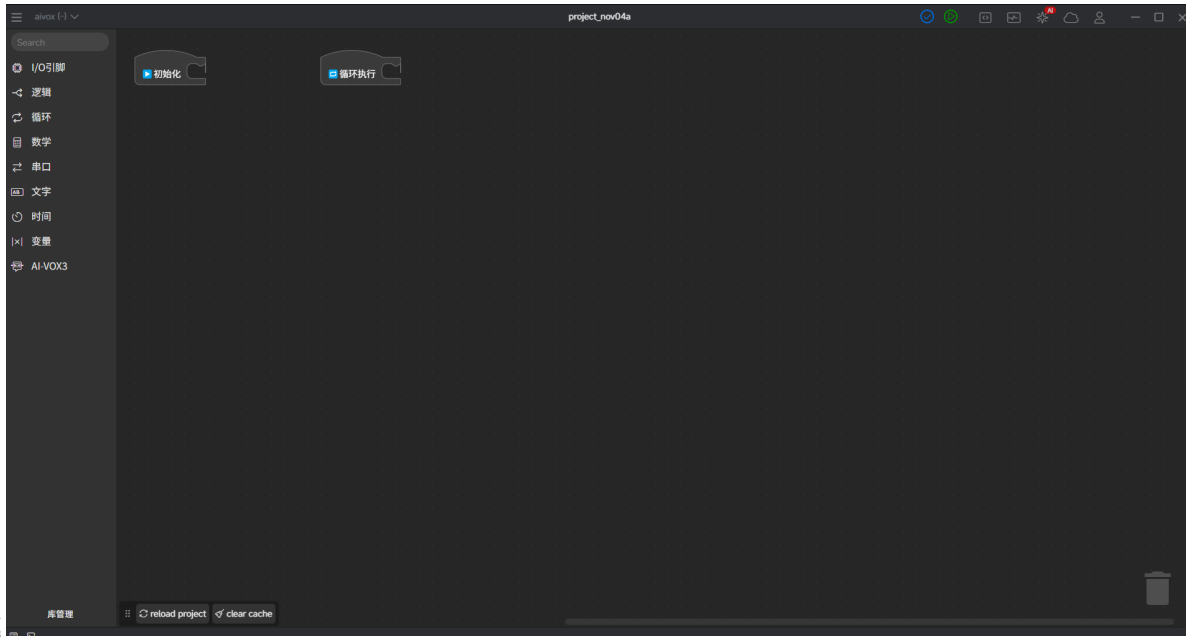
1. 选择新建项目



2. 选择AI-VOX开发板



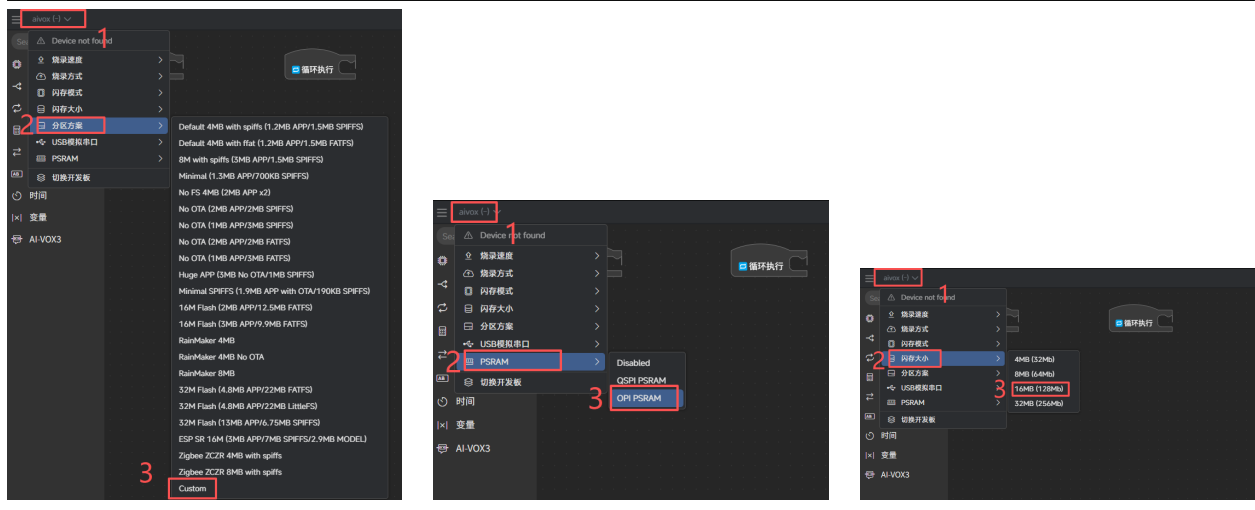
3. 选择项目路径, 版本号选择3.2.1



4. 项目创建完成

## 2.3 配置选择

AI-VOX图形化库项目要求硬件Flash大小(闪存大小)为16M, PSRAM为OPI PSARM, 分区表(Partition Scheme)选择Custom, 软件选择如下:



以上配置都选好了, 最后将AI-VOX的硬件通过Type-C转USB数据线连接好电脑, 按下AI-VOX的电源键, 点击选择对应的端口, 如下图, 就可以愉快的体验AI编程。

## 3. 图形化块介绍

选择工具栏的AI-VOX3菜单, 就会显示AI交互所需要块, 块分为五类, 分别为控制、初始化、事件、MCP、自定义; **控制**: 基本的AI-VOX硬件的控制, 包含音量、显示屏亮度、显示屏内容显示等常用块; **初始化**: 常用的网络、音频、显示的初始化; **事件**: AI大模型返回来的事件, 包含激活信息、表情、聊天

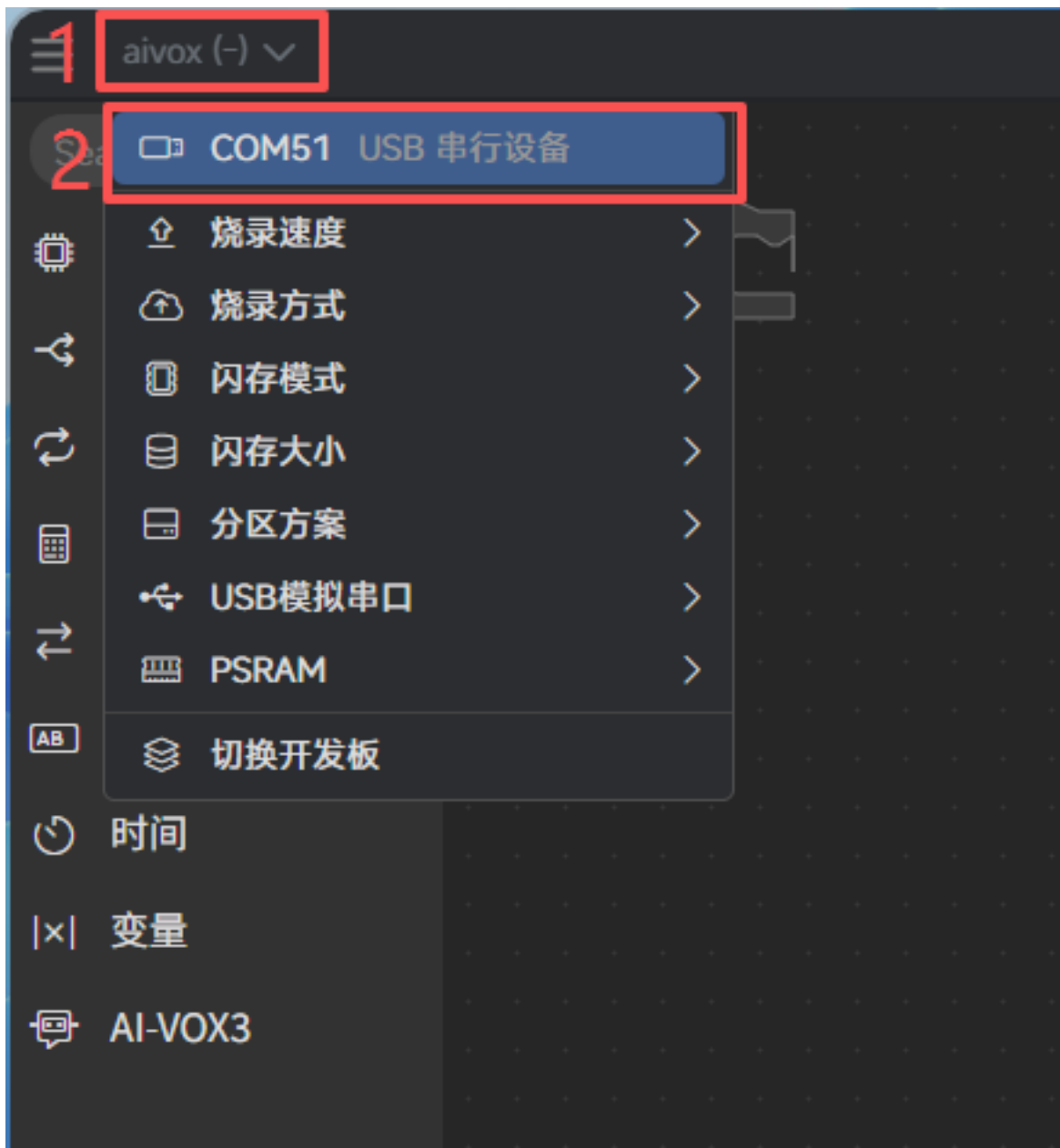





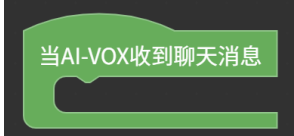

图 1: alt text

信息； **MCP**：MCP外设注册和对外设的控制； **自定义**：对OTA、WebSocket服务器的自定义； 接下来将一一讲解每个块的功能




分类	块	描述	功能
控制		启动AI-VOX AI交互引擎	对整个设备的启动，先初始化设备后，再使用启动块
控制		设置AI-VOX的音量	对音量进行设置，需要先初始化音频解码器后，再使用
控制		设置AI-VOX屏幕亮度	对屏幕亮度进行设置，需要先初始化显示屏后，再使用
控制		在显示屏的不同区域显示不同内容	“下拉框选择要选择的区域，区域包含表情框、状态框和内容框；表情框：屏幕左上角显示表情；状态框：屏幕正上方，紧连着表情框；内容框：屏幕中间内容区域，可显示文字、表情等”

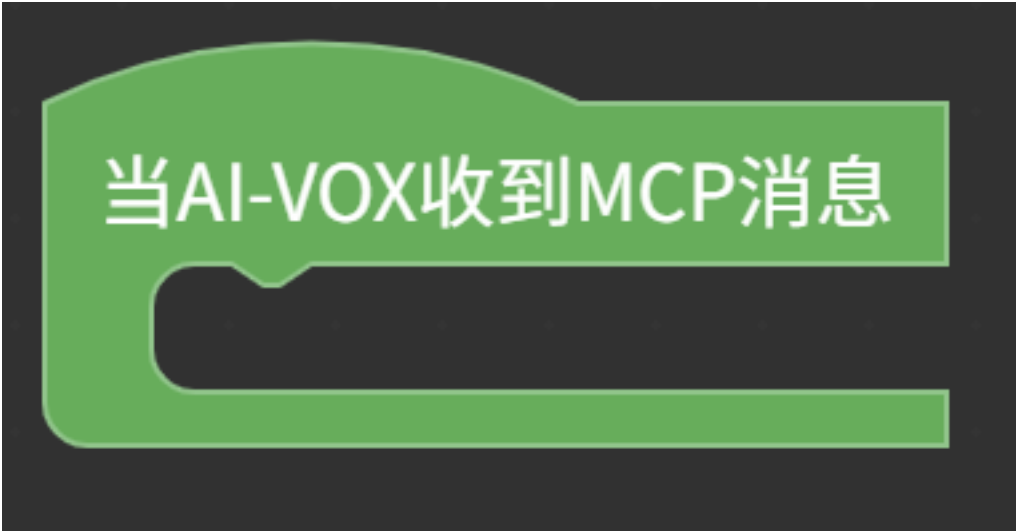
分类	块	描述	功能
控制		“AI-VOX AI交互引擎启动之后，设备的状态”	“AI-VOX AI引擎启动之后，设备会分为八个状态，分别为：空闲中：设备处于空闲状态，网络连接之后，未初始化AI引擎之前的状态；初始化完成：初始化AI引擎完成；加载协议中：加载后台服务器的协议；加载协议失败：协议加载失败，出现该状态，为连接后台服务器失败；需要保证网络畅通，重启硬件设备解决该问题；待命：设备准备就绪，可以通过唤醒词“你好，小智”唤醒设备，进行语音对话；连接中：唤醒设备后，短暂的设备连接服务器的过程；聆听中：设备聆听用户说话的过程；说话中：设备通过喇叭说话的过程；”
初始化		WIFI配置	“WIFI信息的配置；AI-VOX设备只支持2.4GHZ频段的网络；后期该块会新增smartconfig配网模式”
初始化		AI-VOX的网络连接状态	“AI-VOX设备配置网络信息之后，连接网络的状态，分别为：WIFI连接中和WIFI连接完成两个状态；当设备一直处在连接中的状态时，需要检查WIFI信息是否正确，WIFI频段是否为2.4GHZ频段”

分类	块	描述	功能
初始化		初始化ES8311音频编解码器	“AI-VOX3设备板载ES8311音频编解码器和3W音频放大器，引脚默认为AI-VOX设备连接的引脚，如果其他设备可以选择对应的引脚”
初始化		初始化标准的I2S音频编解码器设备	“是ES8311编解码器的一部分功能，AI-VOX3硬件不需要使用该块；AI-VOX硬件需要使用该模块”
初始化		初始化标准的I2S音频编解码器设备	“是ES8311编解码器的一部分功能，AI-VOX3硬件不需要使用该块；AI-VOX硬件需要使用该模块”
初始化		初始化显示屏	AI-VOX使用是以ST7789为驱动的1.54寸240*240分辨率SPI接口的LCD屏
初始化		初始化显示屏会话样式	“显示屏会话样式分两种：微信对话模式和普通模式；微信对话模式：和微信聊天功能一样，左右一问一答样式；普通模式：在显示屏中间显示，没有对话的样式；”

分类	块	描述	功能
事件		启动设备后，设备收到的激活信息	“该事件只会出现在未绑定后台服务器的时候触发；激活信息包含：激活码和激活网址；未绑定的设备需要将获取的激活码去激活网址绑定设备。”
事件		激活信息选项	包含激活码和激活网址两个信息，激活码和激活网址都是字符串类型。
事件		启动设备后，设备收到的表情信息	设备收到表情事件，可以将收到的表情码进行对应的操作。
事件		收到的表情码	事件返回的表情码，字符串类型。
事件		判断收的表情码是什么状态	该块为判断块，返回布尔值，判断当前的表情是什么。
事件		表情包列表	列举系统默认的表情。
事件		设备收到聊天消息时的逻辑处理	“当设备收到聊天事件时，包含两个信息，分别为聊天信息角色和聊天信息内容；角色包含助手和用户。”
事件		判断当前的消息是谁说的	该块为判断块，返回布尔值。
事件		当前的消息内容	当前的消息内容，字符串类型。



分类	块	描述	功能
MCP		注册MCP服务	“该块为MCP服务的基础，控制外设时，需要通过该块注册该硬件到服务器；可以通过点击+图案为该外设添加参数，一个服务可以有多个参数或者没有参数；当需要控制多个不同外设时，需要添加多个注册的块。”
MCP		注册MCP服务名称	“名称使用英文，描述使用中英文都可以，尽可能描述详细，越详细AI反馈越准确，参数也需要在描述中体现；结合上面块使用。”
MCP		注册MCP服务的参数	“参数是具体外设的属性，可以是灯光亮度、温湿度的温度或湿度、电机的速度或方向等；参数使用英文格式，参数类型有布尔型和数值型；数值型有最大值、最小值限制；同注册MCP服务块一起使用。”



MCP | 当AI-VOX设备收到MCP消息时 | 该块主要处理接收到MCP服务器传过来的内容进行对应的业务处理。 |



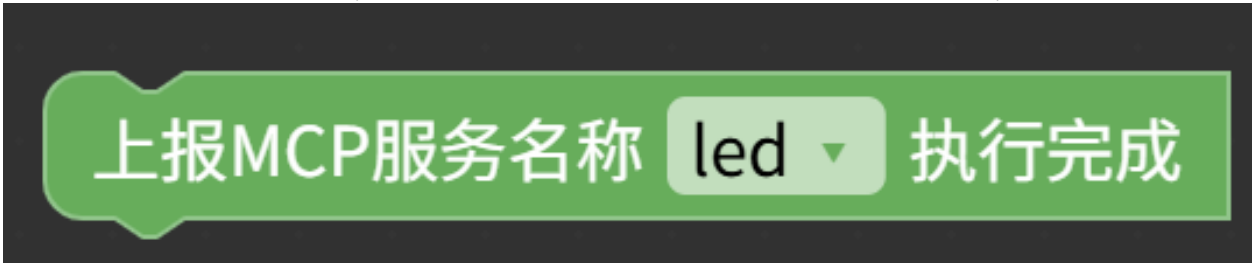
MCP | 判断MCP服务器返回来的服务名称 | 判断MCP服务器返回来的服务名称，块返回布尔值。 |



MCP | 获取对应的服务名称的参数结果 | “可以通过该块执行具体的动作；比如灯的亮灭，电机速度的调整等；块返回的数据类型由参数的类型决定。” |



MCP | 设置MCP服务器对应名称的参数值 | 通过该块设置MCP服务器对应服务的具体参数的值。 |



MCP | “上报给MCP服务器对应的名称的业务处理执行完成” | 告知MCP服务器该服务的业务处理已经执行完成。 |



自定义 | 设置自定义OTA服务的地址 | 自定义OTA服务的地址。 |



自定义 | 设置Websocket服务器的地址和请求头 | 设置Websocket服务器的地址和请求头。 |