SuperController 使用说明书

适用版本：v0.31.1 及以上 | 固件指令集：CT V2R05C01 | 更新日期：2026-11-11 | ComXim

SuperController 是 ComXim CT 全系设备（电动转盘 / 智能灯 / 程控转盘）的统一上位机控制软件， 支持 **多设备多页签**、**图形化控制**、**协议终端**、**工厂标定**，并内置 **AI 接入能力**——让 WorkBuddy、Codex、Claude、豆包等主流 AI 工具用自然语言直接操控硬件。

目录

1. 软件架构总览 | 2. 安装与启动 | 3. 连接设备 | 4. 转盘控制页 | 5. 灯光控制页 | 6. 工厂模式 | 7. AI 接入总览 | 8. WorkBuddy 接入 | 9. Codex CLI 接入 | 10. Claude Desktop / Cursor 接入 | 11. 豆包 / 扣子 / ChatGPT 等接入（HTTP 方式） | 12. AI 工具全清单（34 个） | 13. HTTP API 参考 | 14. 常见问题与错误码 | 15. 安全与权限

1. 软件架构总览

SuperController 采用「五层单向依赖」架构：界面与底层协议彻底分离，保证稳定性与可扩展性。 所有控制指令（CT+ 协议，ASCII 明文）只由协议层统一组装，传输层只负责搬运字节。

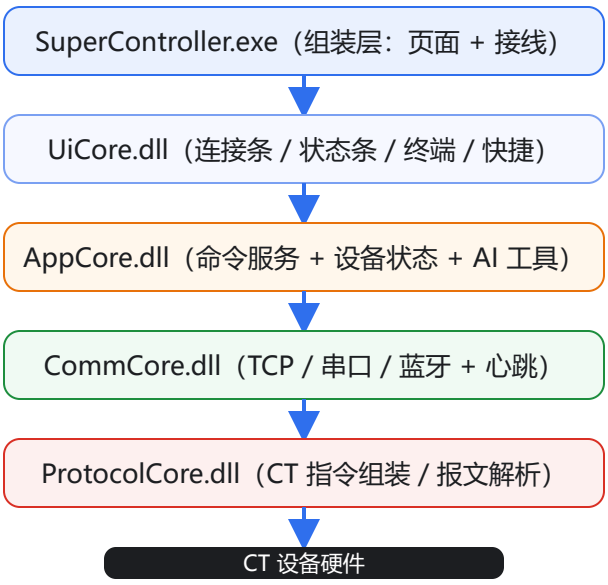

图 1-1 五层架构与指令流向（严格单向依赖，UI 不触碰协议与串口）

三条数据链路贯穿全系统：

链路	路径
指令流	UI 按钮 / AI 工具 → CommandService → CT 指令组装 → 链路管理 → 设备
数据流	设备字节 → 链路管理 → 解析 → 设备状态（单一数据源）→ 信号广播 → 界面刷新
日志流	每条收发报文 → 终端面板实时显示（图形操作与协议报文一一对应）

2. 安装与启动

2.1 安装

1. 双击安装包 `SuperController_Setup.exe`；
2. 选择安装目录（默认 `C:\Program Files\ComXim\SuperController\`）；
3. 安装完成后桌面与开始菜单生成快捷方式。安装包已内置全部运行库，**无需安装 Qt 或 VC++ 运行库。**

2.2 界面总览

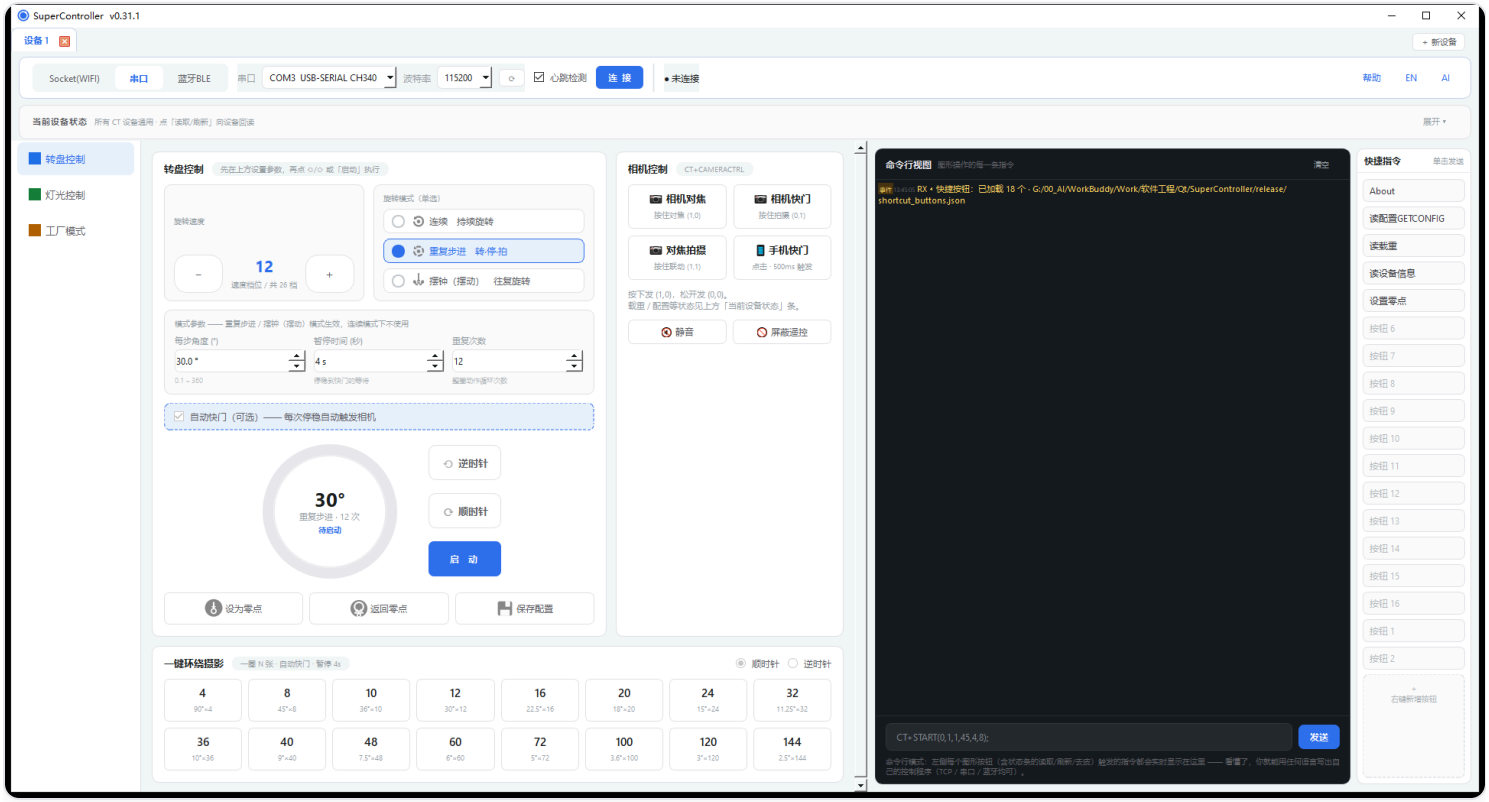

图 2-1 主界面：顶部页签（多设备）｜连接条（WiFi/串口/蓝牙）｜左侧导航（转盘控制 / 灯光控制 / 工厂模式）｜右侧命令行终端与快捷指令

- **多页签**：点右上角「+ 新设备」为每台设备开独立页签，互不干扰；页签可关闭（×）、双击重命名、右键管理、拖动排序。
- **终端面板**：所有图形操作对应的 CT 协议报文实时显示，图形操作 ↔ 协议报文一一对应，是学习 CT 协议的最佳窗口。
- **快捷指令**：常用动作一键直达。

3. 连接设备

通道	操作	说明
Socket(WiFi)	输入设备 IP + 端口	Wi-Fi 版设备；设备 IP 可用 `CT+SETIP` 修改（重启生效）
串口	下拉选择 COM 口 + 波特率	下拉自动枚举并显示驱动描述（如 COM3 USB-SERIAL CH340）
蓝牙 BLE	点扫描 → 选择设备名	BLE GATT 连接

> 💡 **心跳检测**：勾选后软件每秒与设备互发心跳，5 秒失联自动告警，状态胶囊实时显示"心跳 N"。多设备场景强烈建议开启。

4. 转盘控制页

4.1 三种旋转模式

模式	行为	典型用途
连续	匀速持续旋转	视频拍摄、展示
重复步进（转·停·拍）	转一个角度 → 停 → （自动快门）→ 再转，直到转完 N 次	360° 产品图拍摄
摆钟（摆动）	在 ±角度内往复摆动	氛围展示、动效拍摄

4.2 操作要点

- **参数区**：每步角度、暂停秒数、重复次数、自动快门——「重复步进 / 摆钟」模式生效；
- **方向按钮**：↺ 逆时针 / 顺时针 ↻，点击立即按当前参数启动；「启动」按钮复用记住的方向；
- **圆盘**：中央圆盘实时显示角度与进度（蓝=顺时针弧，橙=逆时针弧），下方显示模式与进度（如"重复步进 · 5/12 次"）；

- **速度**：圆形 ± 按钮升降档，或直接设定档位（0~上限）；
- **零点三键**：⊙ 设为零点（当前位记为零）｜↩ 返回零点｜💾 保存配置（写入设备 Flash）。

> 💡 一键环绕摄影：快捷指令里点「环绕 N 张」——自动按 360/N 设角度、开自动快门、暂停 4 秒、转 N 次，一条指令完成整套产品图拍摄流程。

5. 灯光控制页

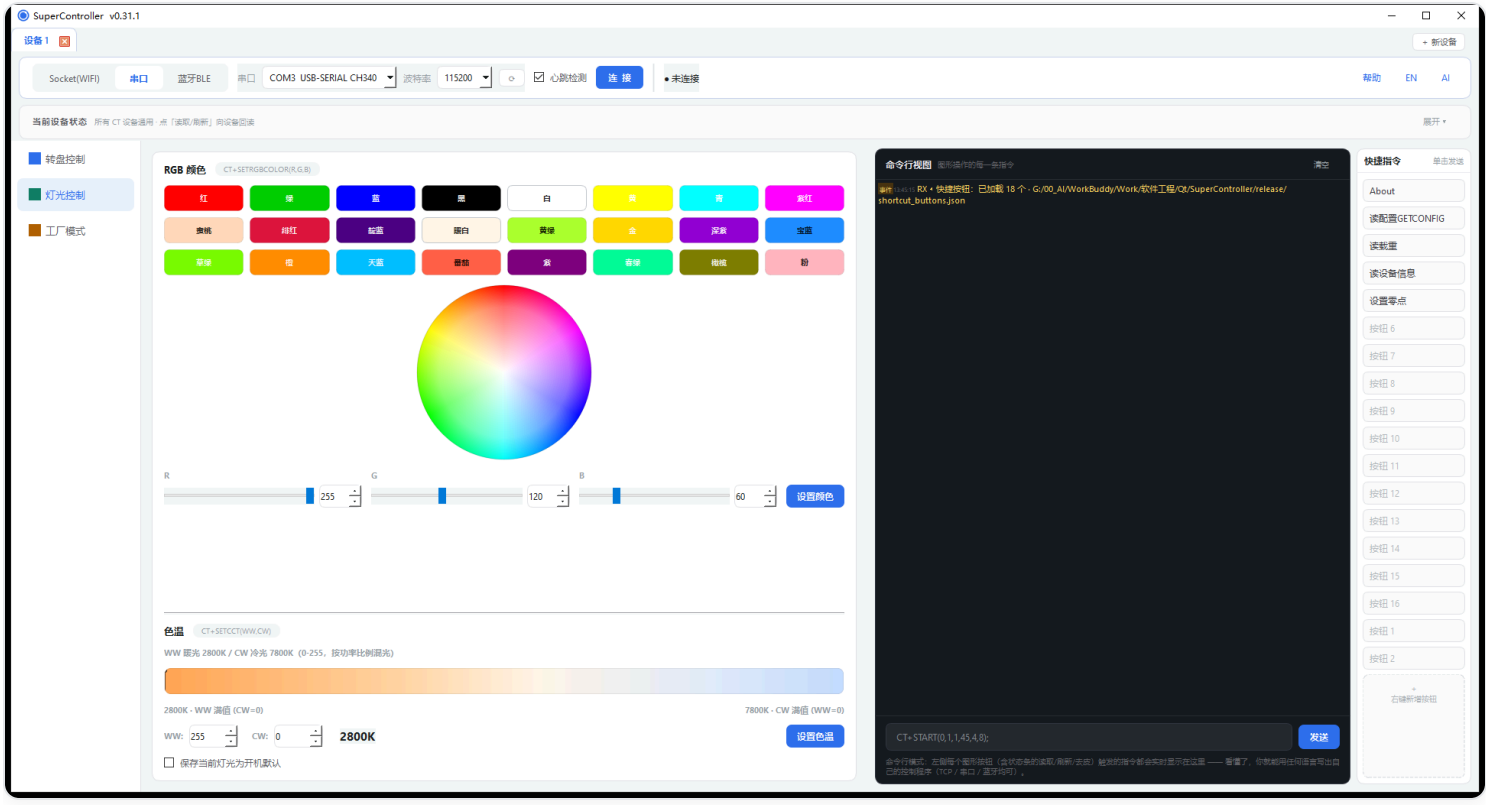

图 5-1 灯光控制页：RGB 预设色与调色、色温条、相机控制、附件开关

- **RGB 颜色**：预设色块一键直达，或用圆形调色盘取任意颜色，对应 `CT+SETRGBCOLOR`；
- **灯效**（支持的机型）：常亮 / 呼吸 / 脉冲 / 流彩，对应 `CT+SETRGBC`；
- **色温灯**：WW 暖光（2800K）/ CW 冷光（7800K）双通道 0~255 混光，渐变条拖动即调；
- **相机控制**：对焦 / 快门按住语义（按下触发、松开结束），手机快门配件一键触发；
- **附件开关**：蜂鸣静音、屏蔽红外遥控——防止现场遥控器误操作。

6. 工厂模式（密码保护）

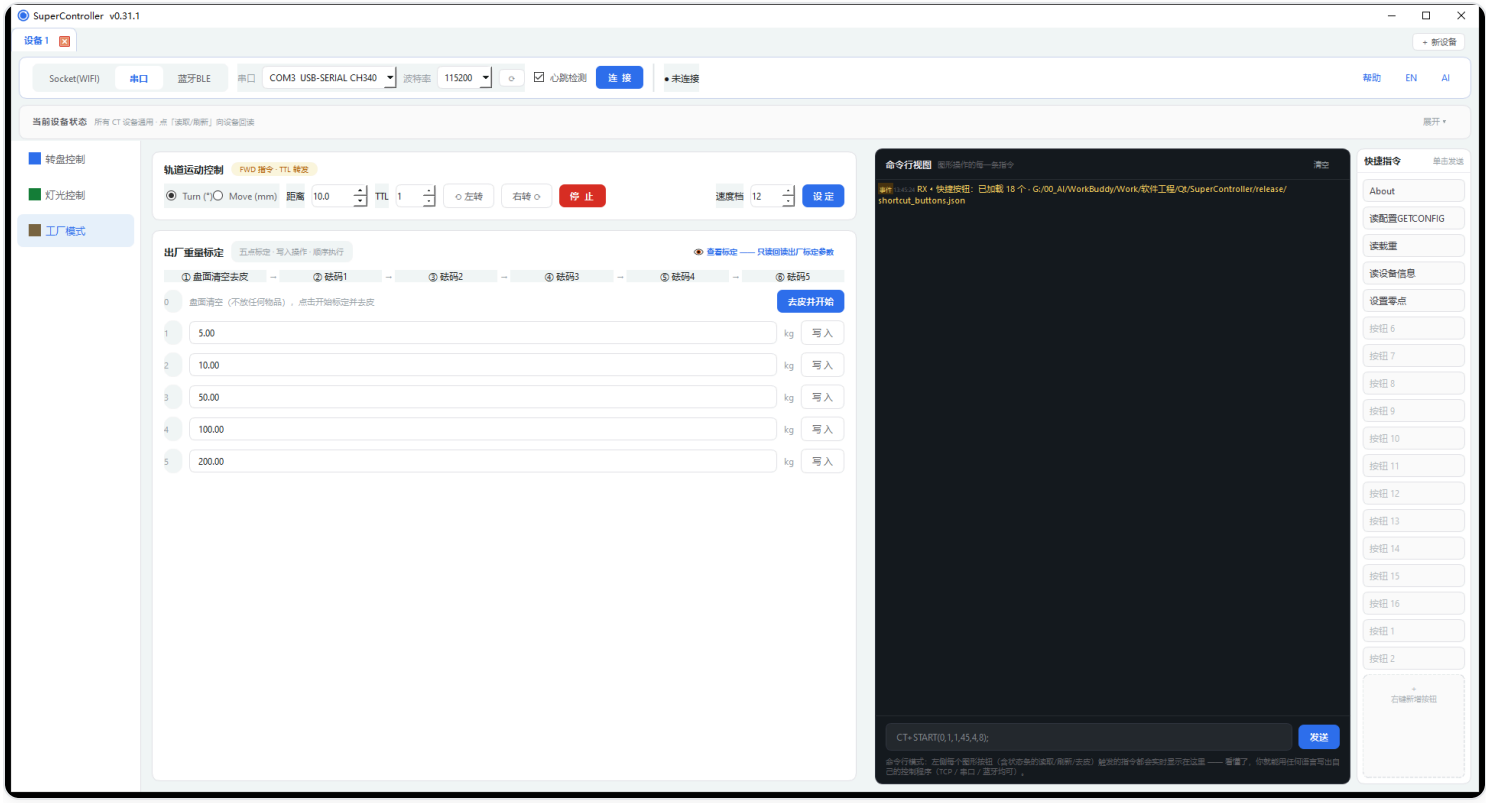

图 6-1 工厂模式：轨道运动控制与五点砝码标定

- **密码门**：默认锁定，输入工厂密码解锁（会话级，重启软件需重新输入）；
- **轨道运动**：FWD 指令族——轨道旋转（按角度）/ 轨道平移（按毫米）/ 直接设速度档 / 停止，供产线调试；
- **五点砝码标定**：①清空去皮 → ②~⑥ 依次放入标准砝码（默认预填 5/10/50/100/200 kg，可改）→ 点「写 入」。标定数据写入设备电子秤，是称重准确性的基准。

> ⚠ 标定数据一旦被覆盖，普通用户无法自行恢复（需重新走砝码物理标定流程）。这也是本软件 **默认不允许 AI 写标定** 的原因（见第 15 章）。

7. AI 接入总览

SuperController 内置 AI 接入能力：把设备的全部控制能力封装成 **34 个 AI 工具**， 主流 AI 工具通过两条路径接入后，即可用自然语言直接操控硬件。

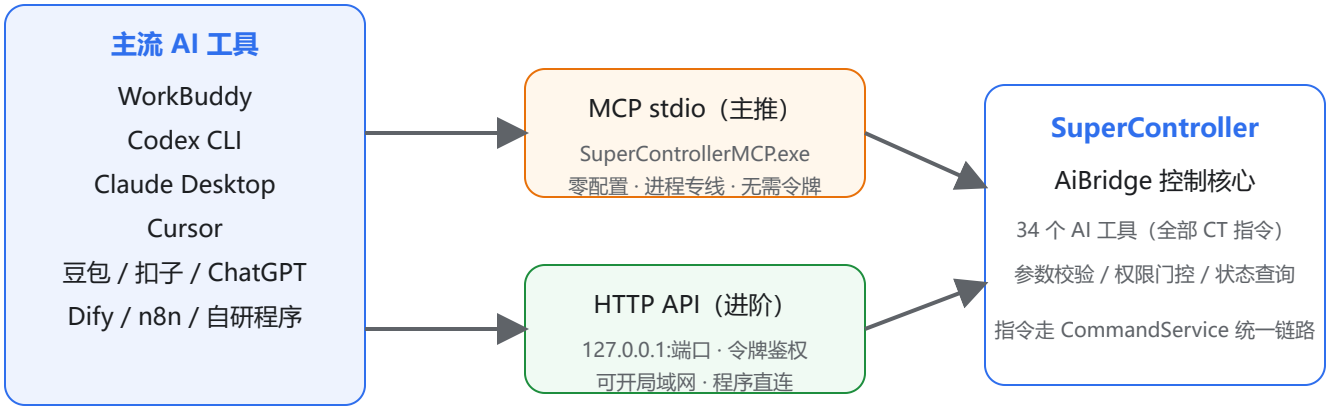

图 7-1 AI 接入架构：两条路径，一个控制核心，全部能力

7.1 两条路径怎么选

	MCP stdio（主推）	HTTP API（进阶）
配置成本	**零配置**：粘一段连接器 JSON 即可	需开网络开关，程序直连
令牌	不需要（进程专线）	需要（自动生成，HTTP 唯一门锁）
支持的工具	WorkBuddy / Codex / Claude Desktop / Cursor 等原生支持 MCP 的工具	豆包 / 扣子 / ChatGPT Actions / Dify / n8n 等一切能发 HTTP 的系统
硬件所有权	UI 在运行→代理转发；UI 没开→MCP 进程自动接管硬件（自动连上次串口）	由 SuperController 进程服务

7.2 「AI 控制」设置

页签右上角 **AI 按钮** 打开设置：

设置项	默认	说明
允许 AI 控制（MCP 接口）	开	总开关，关闭后 AI 一切调用返回"已禁用"
复制 MCP 配置	-	一键复制连接器 JSON，粘贴到 AI 工具即可

控制端口	19840	1024~65535 可改（高级选项）
监听范围	仅本机	可切「局域网 (0.0.0.0)」——同网段其他设备可连（高级选项）
访问令牌	自动生成	HTTP 调用的"门锁"，可重置（高级选项）
允许 AI 写标定数据	**关**	覆盖出厂标定不可自行恢复，仅在产线/售后场景开启（高级选项）

⚠ **令牌**是一串自动生成的 32 位十六进制"访问密码"（如 `96a02ac1787846fe9c50344f967f7ca6`）： 凡通过 HTTP 方式调用控制端口的程序都必须在请求头 `X-SC-Token` 中携带它，否则返回 401。 MCP 方式走进程专线不需要令牌。开启局域网监听后令牌是唯一门锁，请妥善保管；点「重置」后需同步更新所有调用方。

8. WorkBuddy 接入

1. 启动 SuperController，连接设备；点 **AI** 按钮确认「允许 AI 控制」已开启，点**「复制 MCP 配置」**；
2. WorkBuddy 左下角 **「+」→ 连接器 → MCP → 自定义服务器**，粘贴刚才复制的 JSON：

```
{
  "mcpServers": {
    "SuperController": {
      "command": "C:\\Program Files\\ComXim\\SuperController\\SuperControllerMCP.exe"
    }
  }
}
```

3. 保存后在连接器列表中**点「信任」启用**（新 MCP 不会自动生效）；
4. 回到对话，直接说人话：

对 WorkBuddy 说	AI 自动调用
"看一下转盘现在什么状态"	get_status
"顺时针间歇模式，每 30° 停 4 秒拍一张，转一整圈"	start_rotation(angle=30, mode=repeat_step, shutter=true, times=12)
"转快一点" / "停" / "回到零点"	change_speed / stop_rotation / return_to_zero
"把灯调成紫色呼吸效果"	set_rgb_effect(128,0,128,effect=1)
"转盘现在偏了多少度"	get_offset_angle

> 💡 运行模式自动适配：SuperController 开着 → AI 指令走 UI 进程（终端面板可实时看到报文）；没开 → MCP 进程自动无界面接管硬件并连上上次使用的串口。

9. Codex CLI 接入

编辑 `~/.codex/config.toml`，加入：

```
[mcp_servers.SuperController]
command = "C:\\Program Files\\ComXim\\SuperController\\SuperControllerMCP.exe"
```

重启 Codex CLI 后，用 `/mcp` 命令可看到 SuperController 及其 34 个工具； 对话中直接下达控制指令即可。

10. Claude Desktop / Cursor 接入

两者均使用 `mcp.json` / `claude_desktop_config.json`，格式与第 8 章完全相同：

工具	配置文件位置
Claude Desktop	设置 → 开发者 → 编辑配置（`%APPDATA%\Claude\claude_desktop_config.json`）
Cursor	设置 → MCP → Add MCP Server（粘贴同一份 JSON）

保存重启后，工具列表出现 SuperController 的 34 个工具即接入成功。

11. 豆包 / 扣子 / ChatGPT 等接入（HTTP 方式）

凡是**能发 HTTP 请求**的 AI 平台都走这条路径（需要开启高级选项中的网络控制）：

平台	接入方式
扣子 Coze / Dify / n8n	工作流里加「HTTP 请求」节点，POST 到 `http://<电脑IP>:19840/api/call`，头带 `X-SC-Token`，体 `{"name":"start_rotation","arguments":{...}}`
ChatGPT（Actions）	按 OpenAPI 规范描述 `/api/tools` 与 `/api/call` 两个端点导入 Action
豆包桌面版	支持 MCP 的版本按第 8 章 JSON 配置；其余场景用 HTTP API + 智能体编排

任意自研程序	直接按第 13 章 HTTP API 规范调用

⚠ 局域网模式 = 把控制权开放给同网段设备。请确保：① 家庭/办公网络可信；② 妥善保管令牌；③ 不需要时切回"仅本机"。

12. AI 工具全清单（34 个，覆盖 CT V2R05C01 全部指令）

12.1 状态查询（只读）

工具	参数	说明
get_status	-	连接/身份/进度/档位/载重 一次性快照
get_device_info	-	产品名 / 电子条码 / 固件版本
get_weight	-	当前盘面载重 kg
get_offset_angle	-	当前偏离零点角度（±xx.xx°）

12.2 旋转控制

工具	参数	说明
start_rotation	angle, mode, direction, times, pause_seconds, shutter	一键启动（六参数齐发）
stop_rotation	-	立即停止
turn_single	direction, angle	单步转一个角度
run_rotation	-	按已设参数启动（配合下面四个 set 工具）
set_direction	direction	只设方向
set_rotation_mode	mode	只设模式
set_step_angle	angle	只设每步角度
set_pause_time	seconds	只设暂停秒数

set_repeat_times	times	只设重复次数
change_speed	up	升/降一档
set_speed_grade	grade (0~127)	直接设定档位

12.3 零点与标定模式

工具	参数	说明
set_zero_point	-	当前位设为零点
return_to_zero	-	回零点
goto_hardware_zero	-	回硬件零点并设为新零点
set_autocalibrate	mode (0~2)	自动标定模式（关/标定表/自适应）

12.4 灯光与拍摄

工具	参数	说明
set_rgb_color	r, g, b	RGB 颜色
set_rgb_effect	r, g, b, effect	RGB 颜色 + 灯效（0常亮/1呼吸/2脉冲/3流彩）
set_color_temperature	ww, cw	色温双通道混光
set_led_brightness	light_id (1~6), brightness (0~100)	单灯亮度
set_power_led	led_num, grade	电源灯亮度（0=全部）
set_color_led	led1, led2, led3	三路彩色 LED 开关
set_knob_disable	on	屏蔽旋钮调光（仅 CT 控制）
phone_shutter	-	触发手机快门
set_auto_shutter	on	自动快门开关
set_shutter_timing	timing (0/1), duration_ms (100~2000)	快门时机与时长（需兼容相机模块）

12.5 协议与设备管理

工具	参数	说明
save_config	-	保存配置到设备 Flash
set_heartbeat	on	心跳机制开关
set_ack_report	on	CR+OK 应答开关
set_event_report	on	事件上报开关（TB_PAUSE/TB_END）
set_autosleep	minutes (0~30)	自动休眠时间
set_device_ip 危险	ip	设置设备 IP（Wi-Fi 版，重启生效）
factory_reset 危险	-	恢复出厂配置
tare 危险	-	去皮（启动标定流程）
write_calibration 危险	n (1~5), kg (0.01~200)	写第 n 点标定重量

⚠ 标有 危险 的工具默认不对 AI 开放（tare / write_calibration / factory_reset / set_device_ip 由「允许 AI 写标定与高级配置」开关统一门控）。开启前请确认使用场景。

13. HTTP API 参考

```
GET  /api/tools        工具清单（含参数 schema）
POST /api/call         执行工具调用  体: {"name":"...","arguments":{...}}
鉴权: 请求头 X-SC-Token: <令牌>（设置 → AI → 高级选项 查看/重置）
```

curl 示例：

```
curl -X POST http://127.0.0.1:19840/api/call ^
  -H "X-SC-Token: 96a02ac1787846fe9c50344f967f7ca6" ^
  -H "Content-Type: application/json" ^
  -d "{\"name\":\"start_rotation\",\"arguments\":{\"angle\":30,\"mode\":\"repeat_step\"}}"
```

返回为 MCP content 格式：`{"content":[{"type":"text","text":"…"}],"isError":true?}`。

14. 常见问题与错误码

14.1 FAQ

现象	处理
AI 调用返回 "AI control is disabled"	AI 设置里勾选「允许 AI 控制」
返回 "SuperController UI is not running"	属无界面模式自动接管的正常提示；若希望走 UI 代理，先启动 SuperController
AI 看不到标定写入类工具	默认关闭；到高级选项打开「允许 AI 写标定」
端口 19840 被占用	已有实例拥有硬件，MCP 自动进代理模式；或在设置里改端口
查询返回 "--"	该字段尚未查询过，说一句"查设备信息"即可回填
设备无响应	检查串口/WiFi 连接与心跳；终端面板查看 TX/RX 报文定位

14.2 设备错误码（CR+ERR=）

码	含义	码	含义
31	串口超时	51	指令名过长
40	功能不支持（机型无此硬件）	52	指令名无效
41	指令过短（<5 字符）	61	参数为空
42	指令头无效（须 CT）	62	参数过长
43	连接符无效（须 +）	63	参数个数超限
45	参数起始符无效（须 "("）	64	参数个数不匹配
46	参数结束符无效（须 ")"）	70~79	第 N 参数非数字
48	结束符无效（须 ";"）	80~89	第 N 参数超范围

15. 安全与权限

- **默认最小权限**：AI 可查询与基础控制；**写标定 / 恢复出厂 / 改设备 IP 默认禁止**，须在设置高级选项显式开启；
- **本机隔离**：控制端口默认只绑 127.0.0.1，外部设备无法访问；
- **局域网模式**：开启后同网段可达，令牌是唯一门锁——请在可信网络使用并妥善保管令牌；
- **令牌重置**：怀疑泄露时一键重置，旧令牌立即失效；
- **全链路可观测**：AI 发出的每一条指令都走统一链路，终端面板完整可见、可审计。
