

OpenClaw 绑定 GLM-5.2 完整教程

目标：在 Windows 电脑上安装 OpenClaw，并绑定智谱 GLM-5.2 大模型，实现"手机发指令 → 本地 AI 自动执行（创建文档、生成内容、跑代码测试）"。

记录日期：2026-08-14

一、整体思路（先看懂再动手）

两种路线

路线	说明	适用
A. 本地全能（本教程采用）	电脑装 OpenClaw = 大脑 + 执行端，手机直接连	✅ 最稳，无需公网
B. 云端大脑 + 本地执行端	云端 AI 写代码，传本地跑，结果回传	需要内网穿透/公网，成本高

本教程走路线 A：你的电脑本身就是完整的 AI 智能体，手机发消息就能指挥它。

关于硬件（常见疑问）

- DeepSeek / Kimi / GLM 都是云端模型，真正的"运算"在对方服务器
- 你的电脑只负责"收发消息 + 操作本地文件/软件"
- 不需要高性能显卡/CPU，4GB 内存的老电脑都够
- 唯一瓶颈是网速

二、环境准备

- Windows 10 (20H2+) / Windows 11
- 一个可用的聊天软件（后续接手机用）：微信 / 钉钉 / Telegram 等
- 一个智谱开放平台账号 + API Key

获取智谱 API Key

1. 打开 <https://open.bigmodel.cn> 注册登录（新用户有免费额度）
2. 进入「API Keys」页面：<https://bigmodel.cn/usercenter/proj-mgmt/apikeys>
3. 点击「创建 API Key」，复制保存（只显示一次）
4. 智谱 Key 格式：**数字.字母数字** 开头（如 **e57461...**），**不是 sk-** 开头

三、安装 OpenClaw (Windows)

方法：PowerShell 命令行安装（本教程采用）

1. 右键「开始」→ 打开 **Windows PowerShell (管理员)**
2. 执行安装命令：

```
powershell iwr -useb https://openclaw.ai/install.ps1 | iex
```
3. 安装器会自动：检测系统 → 安装 Node.js → 安装 Git → 安装 OpenClaw

⚠ 常见坑：Git 下载超时导致安装失败

现象：提示 `Portable Git bootstrap failed: 操作超时`，然后 `OpenClaw installation failed`

解决：

1. 手动去 <https://git-scm.com/download/win> 下载 Git，安装（一路 Next）
2. 关掉当前 PowerShell，重新打开（让系统识别 Git）
3. 重新执行安装命令，这次会跳过 Git 直接装 OpenClaw

首次配置 (onboarding)

安装完成后会进入配置向导，按以下回答：

- 问"要不要帮你设好基础配置" → 回答 **yes**
- 问"配置模型提供商 (model provider) ?" → 回答 **no**（模型我们后面单独配）

四、配置模型：绑定 GLM-5.2

第 1 步：设置 API 地址 (baseUrl)

```
openclaw config set models.providers.openai.baseUrl "https://open.bigmodel.cn/api/paas/v4"
```

关键概念：baseUrl 填到 `/v4` 就停，不要带 `/chat/completions`（OpenClaw 会自动补全，填完整地址会重复拼接报错）。

第 2 步：查询账号可用模型（避免模型名不对）

```
curl.exe -s "https://open.bigmodel.cn/api/paas/v4/models" -H "Authorization: Bearer 你的真实Key"
```

返回列表里确认 `glm-5.2` 存在即可。

第 3 步：注册模型（注意 --merge）

```
openclaw config set models.providers.openai.models '[{"id":"glm-5.2","name":"glm-5.2"}]' --strict-json --merge
```

坑 1：PowerShell 双引号转义 — JSON 里的每个 `"` 前都要加反斜杠 `\`，否则 PowerShell 会吃掉引号导致解析失败。

坑 2：--merge 参数 — 如果 models 里已有其他模型（比如之前的），不加 `--merge` 会被拒绝（提示会删除已有条目）。

第 4 步：设置默认模型

```
openclaw config set agents.defaults.model.primary "openai/glm-5.2"
```

第 5 步：写入 API Key

```
openclaw config set models.providers.openai.apiKey "你的GLM完整Key"
```

⚠ 最关键的坑：命令行写入 Key 不可靠（PowerShell 可能转义/截断特殊字符）。**强烈建议用记事本直接改配置文件：**

```
notepad %USERPROFILE%\openclaw\openclaw.json
```

找到 `"openai"` 段落的 `"apiKey": "..."`，粘贴你的完整 Key（注意保留逗号）。

第 6 步：重启网关 + 测试

```
openclaw gateway restart
openclaw tui --message "你好，请用一句话介绍你自己"
```

五、踩坑总结（务必看完，省几小时）

坑 1：环境变量覆盖配置（最隐蔽！）

现象：Key、模型、地址全对，直连 curl 也通，但 OpenClaw 一调就 401「令牌已过期或验证不正确」。

根因：之前给别的模型（如 Kimi）设置过环境变量 `OPENAI_API_KEY`（旧 Key）。**OpenClaw 认证优先级：**环境变量 > 配置文件 `apiKey`，所以它拿旧 Key 去请求智谱 → 401。

解决：把 `OPENAI_API_KEY` 环境变量改成当前模型的 Key（或清空）：

```
[Environment]::SetEnvironmentVariable("OPENAI_API_KEY", "你的GLM完整Key", "User")  
# 或清空: [Environment]::SetEnvironmentVariable("OPENAI_API_KEY", $null, "User")
```

然后重启网关。

坑 2：模型名要与账号实际可用一致

- 先 `curl .../v1/models` 查自己账号能用的 id
- Kimi 的 `moonshot-v1-8k` 在新账号可能不存在；GLM-5.2 官方 id 就是 `glm-5.2`

坑 3：模型条目需同时有 id 和 name

`models` 数组里每个对象必须同时有 `id` 和 `name`，缺 `name` 会报 `models.0.name: Invalid input`。

坑 4：区分"开发文档的 cURL"与"OpenClaw 配置"

官方文档给的完整地址 `/v4/chat/completions` 是手动调用用的；OpenClaw 配置只填 baseUrl `/v4`。

六、验证成功后的效果

- `openclaw tui` 能正常和 GLM-5.2 对话
- 本地 agent 具备操作电脑的能力（创建文档、生成内容、保存到指定位置、运行代码测试）

七、接入微信（手机远程指挥）

OpenClaw 通过腾讯官方插件 `openclaw-weixin` 接入微信，**扫码登录**（个人微信，建议用小号）。

第 1 步：安装微信插件

```
openclaw plugins install "@tencent-weixin/openclaw-weixin"
```

安装成功后显示： `Installed plugin: openclaw-weixin`

第 2 步：启用插件

```
openclaw config set plugins.entries.openclaw-weixin.enabled true
```

第 3 步：重启网关（加载插件）

```
openclaw gateway restart
```

第 4 步：扫码登录微信

```
openclaw channels login --channel openclaw-weixin
```

- 终端会显示二维码，用手机微信「扫一扫」登录并确认
- 登录成功后显示绑定成功

第 5 步：验证通道状态

```
openclaw channels status --probe
```

看到 `openclaw-weixin ... enabled, configured` 即正常。

💡 手机发「你好」测试：微信收到 AI 回复 = 整条链路（手机→微信→本地AI→回手机）已打通。

八、授权 AI 操控电脑（关键）

执行模式说明（exec mode）

模式	行为	安全
<code>ask</code>	每次操作前询问（默认推荐）	高
<code>allowlist</code>	只允许名单内命令	高
<code>auto</code>	有安全机制时自动执行	中
<code>full</code>	完全放开，任意命令直接执行	⚠️ 低

方案一：保持询问（推荐起步）

AI 执行敏感操作时，在电脑的 PowerShell/控制台弹确认，输入 `y` 批准。

方案二：加入允许名单（半自动）

把常用命令加入免审批名单：

```
openclaw approvals allowlist add --node local "python"
openclaw approvals allowlist add --node local "node"
openclaw approvals allowlist add --node local "powershell"
```

方案三：完全放开（本教程采用，最快）

```
openclaw config set tools.exec.mode full
```

⚠ 注意：选项是 **full**，不是 **allow**（**allow** 会报 **Invalid option**）。

⚠ 风险提示：完全放开后 AI 有电脑最高权限，任何能给这个微信号发消息的人都能指挥电脑，务必保密微信号。

九、桌面图标：双击打开控制台（不用每次敲命令）

方式 A：浏览器图形界面（推荐）

桌面创建「OpenClaw控制台」图标，双击打开网页管理界面：

```
$WshShell = New-Object -ComObject WScript.Shell
$Shortcut = $WshShell.CreateShortcut("$env:USERPROFILE\Desktop\OpenClaw控制台.lnk")
$Shortcut.TargetPath = "cmd.exe"
$Shortcut.Arguments = "/c start http://127.0.0.1:18789"
$Shortcut.IconLocation = "shell32.dll,21"
$Shortcut.Save()
```

方式 B：终端聊天界面（TUI）

```
$oc = (Get-Command openclaw).Source
$WshShell = New-Object -ComObject WScript.Shell
$Shortcut = $WshShell.CreateShortcut("$env:USERPROFILE\Desktop\OpenClaw聊天.lnk")
$Shortcut.TargetPath = "powershell.exe"
$Shortcut.Arguments = "-NoExit -Command `"$oc tui`""
$Shortcut.Save()
```

十、登录控制台需要「网关令牌/token」

首次打开浏览器界面 <http://127.0.0.1:18789> 会要求输入网关令牌。

怎么查网关令牌（重点！）

`openclaw config get gateway.auth` 会显示 `__OPENCLAW_REDACTED__`（出于安全打码，看不到真实值），所以要用下面的方法：

方法 1：直接看配置文件

```
notepad "C:\Users\Administrator\.openclaw\openclaw.json"
```

⚠ 路径是 `C:\Users\Administrator\...`（完整路径），用 `%USERPROFILE%` 有时会找不到文件。

打开后在记事本里用 `Ctrl+F` 搜索 `"token"`，找到：

```
"auth": {  
  "mode": "token",  
  "token": "8b8b...3b9d"  
}
```

把 `"token": "..."` 里那串字符复制出来，粘贴到界面的「网关令牌/Token」输入框。

密码框：配置里一般没有 password，留空即可；留空不行就把 token 也填进去试。

方法 2（找不到配置文件时的定位）：

```
openclaw config file
```

会显示配置文件真实路径（如 `~\.openclaw\openclaw.json`）。

⚠ 安全提醒

- 网关令牌 = 你电脑本机登录凭证，**只在自己电脑上用，不要发到任何聊天里**
- 忘掉可以随时打开配置文件重新查看

十一、完整效果与日常使用

- **开机自动运行**：网关是后台服务（Windows 计划任务），开机自启，**无需手动启动**
- **日常入口**：双击桌面「OpenClaw控制台」图标 → 打开浏览器界面
- **手机指挥**：微信发指令 → 本地 AI（GLM-5.2）直接操控电脑 → 结果回手机
- **示例指令**：
 - 「在桌面创建文档 test.txt，内容写你好」
 - 「运行 python 脚本并告诉我结果」
 - 「整理桌面文件」

完整配置清单（回顾）

```
// ~/.openclaw/openclaw.json 核心部分
{
  models: {
    providers: {
      openai: {
        baseUrl: "https://open.bigmodel.cn/api/paas/v4",
        apiKey: "你的GLM完整Key", // 建议记事本手动改
        models: [{ id: "glm-5.2", name: "glm-5.2" }],
      },
    },
  },
  agents: {
    defaults: {
      model: { primary: "openai/glm-5.2" },
    },
  },
  plugins: {
    entries: {
      "openclaw-weixin": { enabled: true },
    },
  },
  tools: {
    exec: { mode: "full" }, // 完全放开
  },
}
```