

目录

1 KEIL 和 VS Code 开发环境 1

2 在 VS Code 中安装 Keil 助手和 Cline..... 4

3 接入 DeepSeek 7

4 本地部署人工智能大模型..... 9

1 KEIL 和 VS Code 开发环境

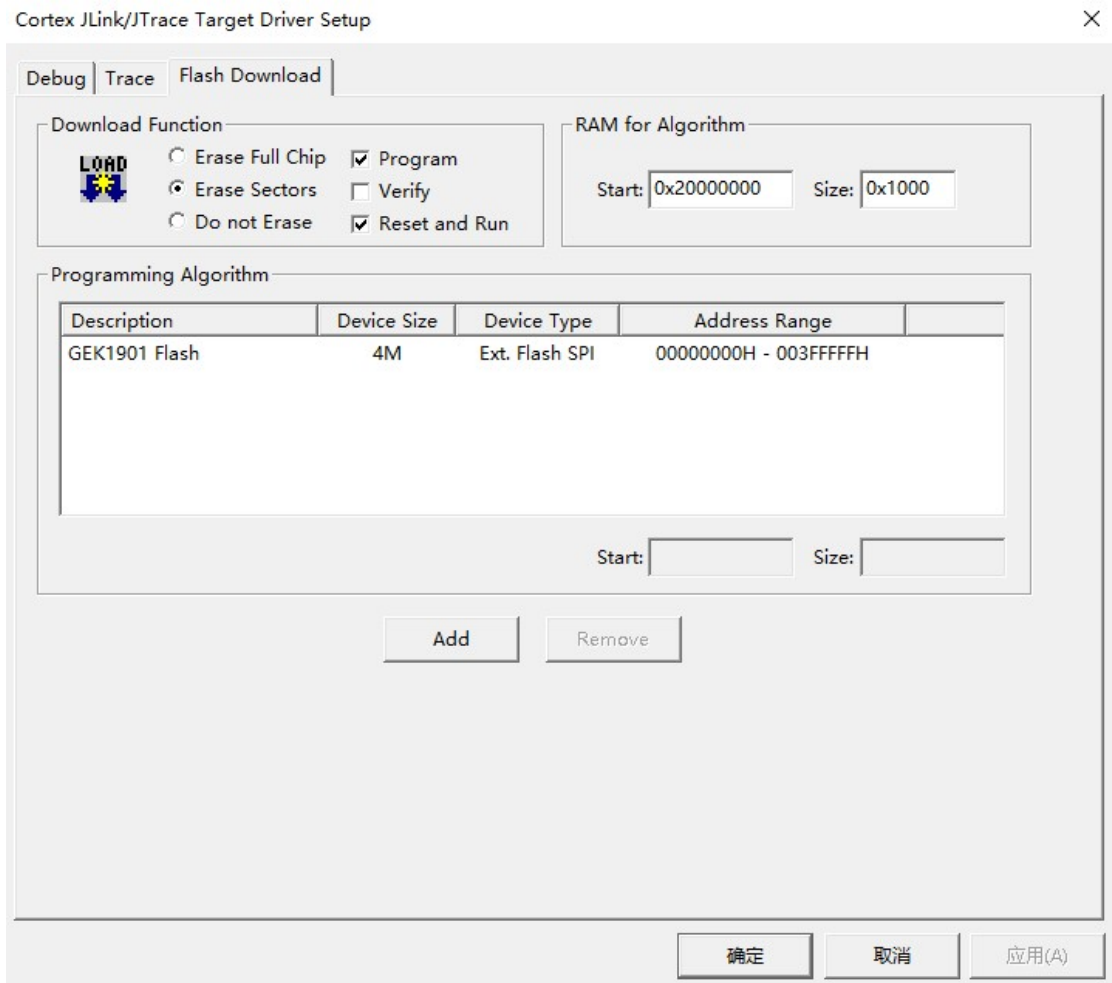
BSR 提供的 SDK 代码是基于 ARM Keil 或者 VS Code 开发的，代码放置在电脑的文件夹时，层级不要太深（不要超过三级），同时**文件夹命名时，不要出现中文字样**，否则可能导致 Keil 下载代码时异常。完整的 SDK 软件代码：

盘 (D:) > GitHub > BSR1901 > software > sdk				
名称	修改日期	类型	大小	
gui	2026/1/9 13:12	文件夹		
include	2026/1/9 13:13	文件夹		
Listings	2026/1/9 15:17	文件夹		
norflash	2026/1/9 13:12	文件夹		
Objects	2026/1/13 17:25	文件夹		
Output	2026/1/9 15:19	文件夹		
peripheral	2026/1/9 13:12	文件夹		
SCGUI	2026/1/9 13:12	文件夹		
startup	2026/1/9 13:12	文件夹		
tft_lcd	2026/1/9 13:34	文件夹		
user	2026/1/9 17:40	文件夹		
BSR1901_4MB.FLM	2025/12/23 19:16	FLM 文件	25 KB	
bsr1901_demo.uvguix.ALIENWARE	2025/12/23 19:24	ALIENWARE 文件	92 KB	
bsr1901_demo.uvguix.Thinkpad	2026/1/13 17:25	THINKPAD 文件	180 KB	
bsr1901_demo.uvguix.ZhouJunfeng	2023/8/13 19:27	ZHOIJUNFENG ...	100 KB	
bsr1901_demo.uvoptx	2026/1/13 17:25	UVOPTX 文件	27 KB	
bsr1901_demo.uvprojx	2026/1/10 8:36	ision5 Project	24 KB	
build.bat	2026/1/9 15:18	Windows 批处理...	1 KB	
byte2uint.exe	2025/1/7 22:22	应用程序	4,102 KB	
img_flower_60x160.jpg	2024/11/23 12:45	JPEG Image	63 KB	
img_star_128x128.jpg	2024/11/28 20:22	JPEG Image	14 KB	
JLinkLog.txt	2026/1/1 10:00	TXT 文件	1,599 KB	
JLinkSettings.ini	2024/2/17 14:19	配置设置	1 KB	
Keil5_disp_size_bar_v0.3.exe	2025/12/26 14:20	应用程序	51 KB	
LCD_mcuTool.exe	2026/1/4 17:06	应用程序	1,616 KB	
LCD_mcuTool.png	2026/1/4 17:22	PNG Image	28 KB	
scatter_bsr1901_demo.sct	2025/10/25 10:47	Windows Script ...	1 KB	

可以为客户提供的 Demo 板：



添加 Flash 下载算法



把 SDK 提供的 BSR1901.FLM 文件放到: C:\Keil_v5\ARM\Flash 目录下，如下图所示。

磁盘 (D:) > Keil_v5 > ARM > Flash

名称	修改日期	类型	大小
ADuC7124_6.FLX	2015/7/8 16:24	FLX 文件	13 KB
ADuC7229.FLX	2015/7/8 16:25	FLX 文件	15 KB
ADUCMxxx.FLM	2015/7/8 17:20	FLM 文件	12 KB
ADUCMxxx_128.FLM	2015/7/8 17:20	FLM 文件	12 KB
AM29F160DB.FLX	2022/12/1 14:29	FLX 文件	14 KB
AM29F160DT.FLX	2022/12/1 14:29	FLX 文件	14 KB
AM29F320DB.FLX	2022/12/1 14:29	FLX 文件	14 KB
AM29F320DBx2.FLX	2022/12/1 14:29	FLX 文件	14 KB
AM29F320DT.FLX	2022/12/1 14:29	FLX 文件	14 KB
AM29F320DTx2.FLX	2022/12/1 14:29	FLX 文件	14 KB
AM29x033.FLX	2022/12/1 14:29	FLX 文件	13 KB
AM29x128.FLM	2022/12/1 14:29	FLM 文件	13 KB
AM29x128.FLX	2022/12/1 14:29	FLX 文件	13 KB
AM29x800BB.FLX	2022/12/1 14:29	FLX 文件	14 KB
AM29x800BBx2.FLX	2022/12/1 14:29	FLX 文件	14 KB
AM29x800BT.FLX	2022/12/1 14:29	FLX 文件	14 KB
AM29x800BTx2.FLX	2022/12/1 14:29	FLX 文件	14 KB
AM29x800DB.FLX	2022/12/1 14:29	FLX 文件	14 KB
AM29x800DBx2.FLX	2022/12/1 14:29	FLX 文件	14 KB
FlashOS.h	2022/12/1 14:29	C/C++ Header	4 KB
GEK1901_4MB.FLM	2025/12/9 11:29	FLM 文件	25 KB
K8P5615UQA_x2.FLM	2022/12/1 14:29	FLM 文件	11 KB
LPC18xx43xx_MX25V8035F.FLM	2022/12/1 14:29	FLM 文件	112 KB
LPC18xx43xx_S25FL032.FLM	2022/12/1 14:29	FLM 文件	77 KB
LPC18xx43xx_S25FL064.FLM	2022/12/1 14:29	FLM 文件	77 KB
LPC407x_8x_S25FL032.FLM	2022/12/1 14:29	FLM 文件	34 KB

如果在硬件连接了 reset pin 的条件下，碰到异常的提示，可以设置一下红色框框：

HW : V9.40 dll : V6.94a

FW : J-Link V9 compiled May 7 2025

Port: SW

Max: 1 MHz

Auto Clk

Automatic Detection

ID CODE:

Manual Configuration

Device Name:

Add

Delete

Update

IR len:

Connect & Reset Options

Connect: with Pre-res

Reset: Reset Pin

Reset after Connect

Cache Options

Cache Code

Cache Memory

Download Options

Verify Code Download

Download to Flash

Interface

USB

TCP/IP

Scan

State: ready

TCP/IP

Network Settings

IP-Address: 127 . 0 . 0 . 1

Port (Auto: 0)

Autodetect

Ping

Misc

JLink Info

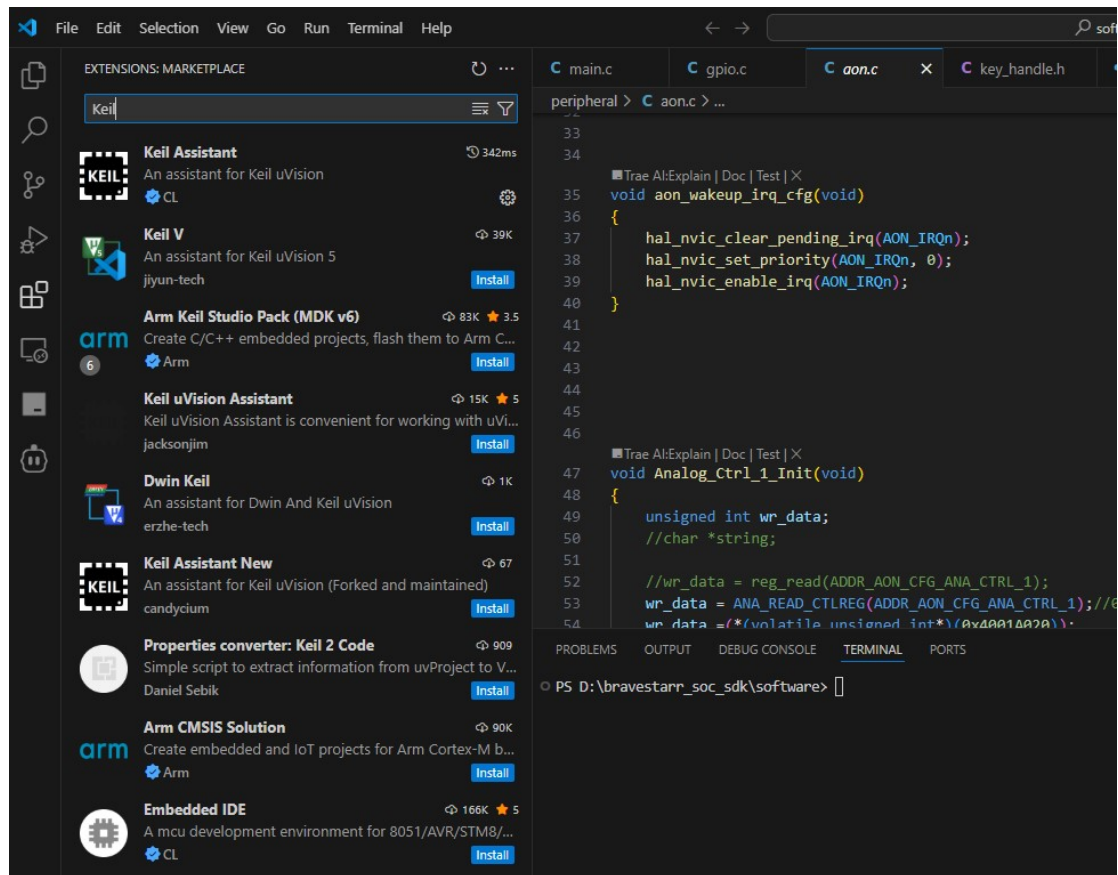
JLink Cmd

确定

取消

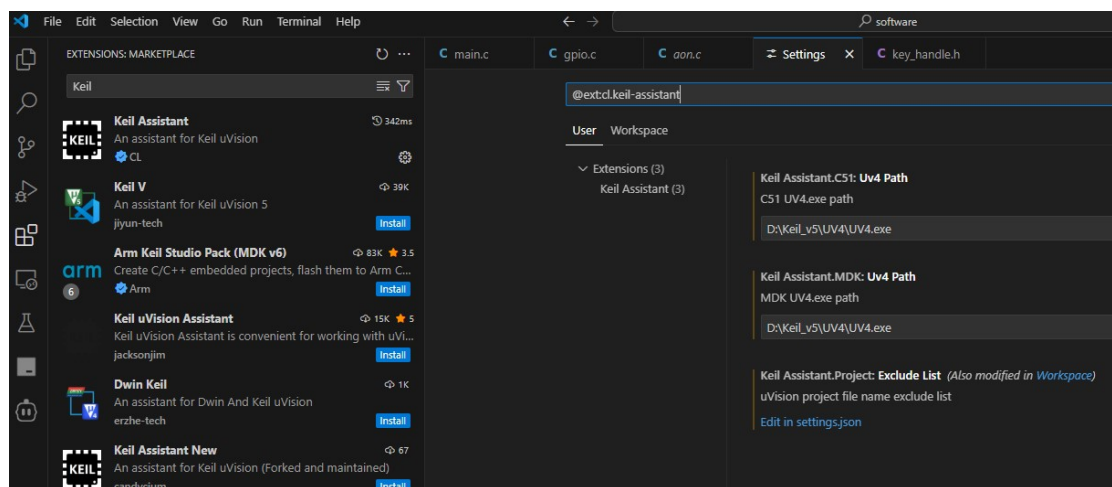
应用(A)

2 在 VS Code 中安装 Keil 助手和 Cline



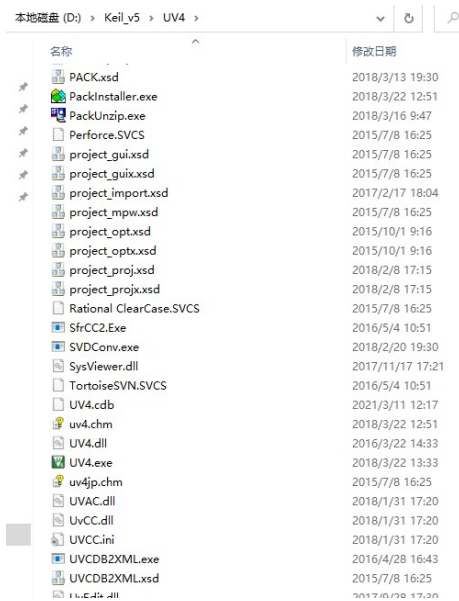
添加 Keil 环境变量 (让 VS Code 能够顺利找到我们的 Keil)

在 Keil 助手的 setting 菜单中, 指定一下本地电脑的 Keil 工具的路径。

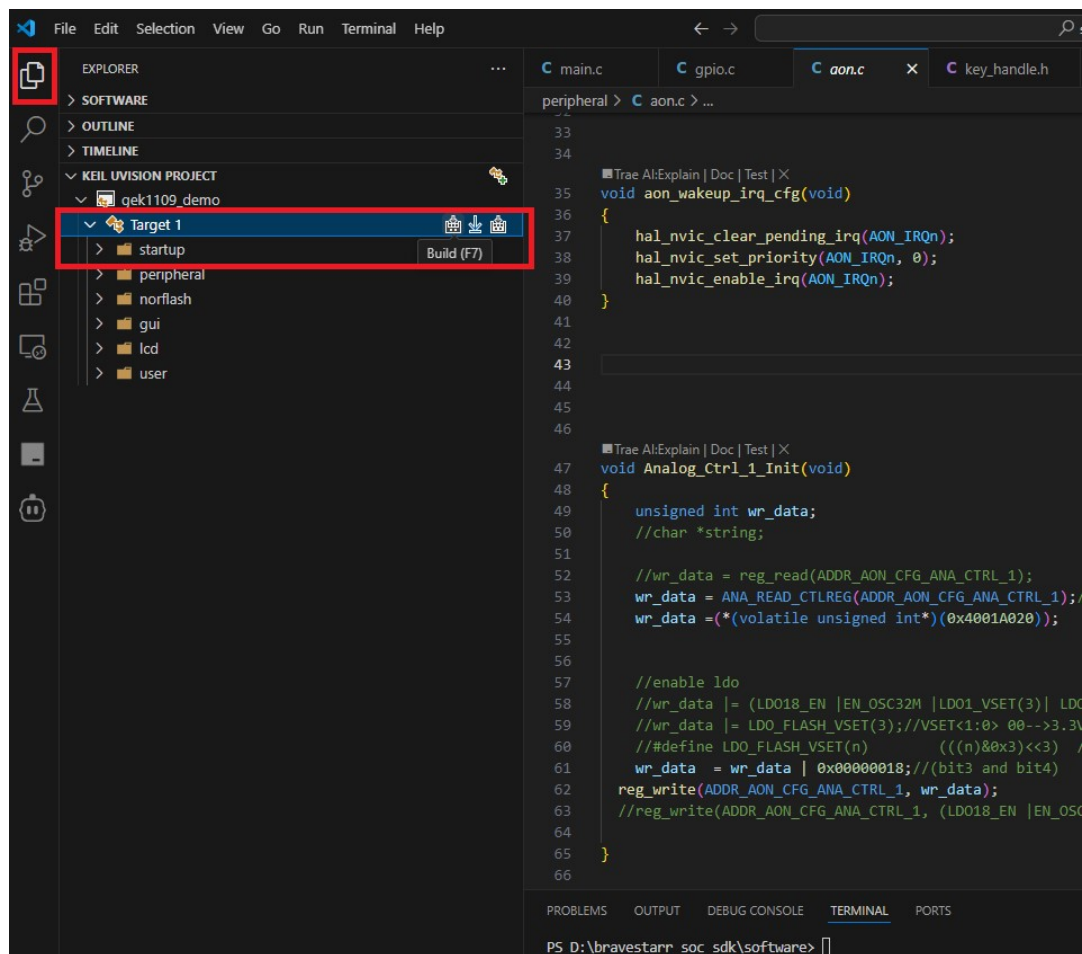


我的工程中没有 C51 的部分,可以不填。

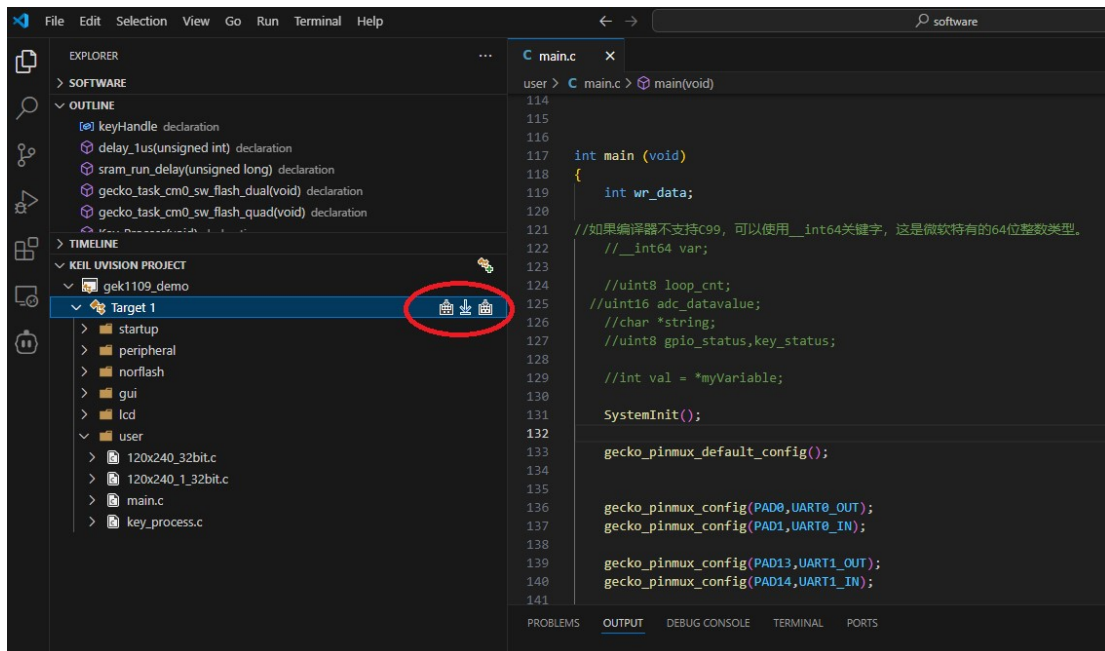
找到你本地电脑上安装的 Keil，把这个路径复制到 Keil Assistant.



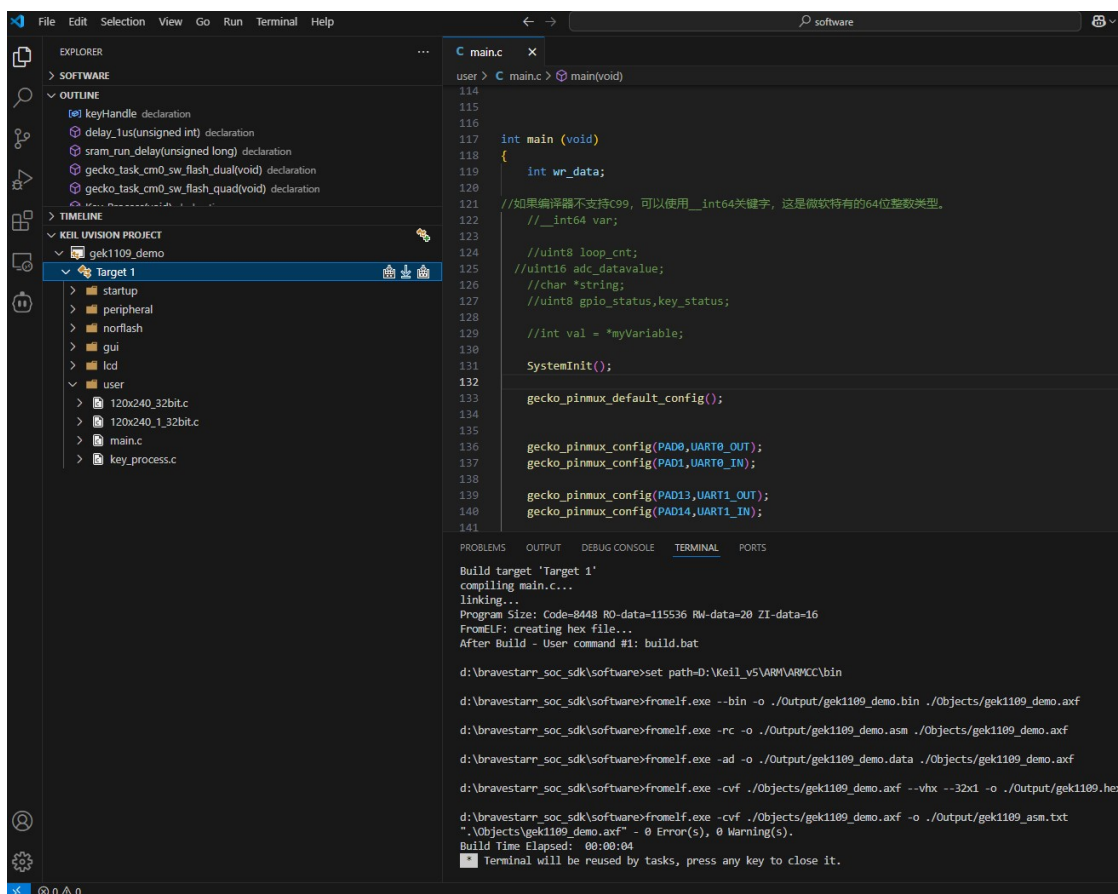
加载 Keil 工程文件



在 VS Code 中进行编译工程以及烧录程序

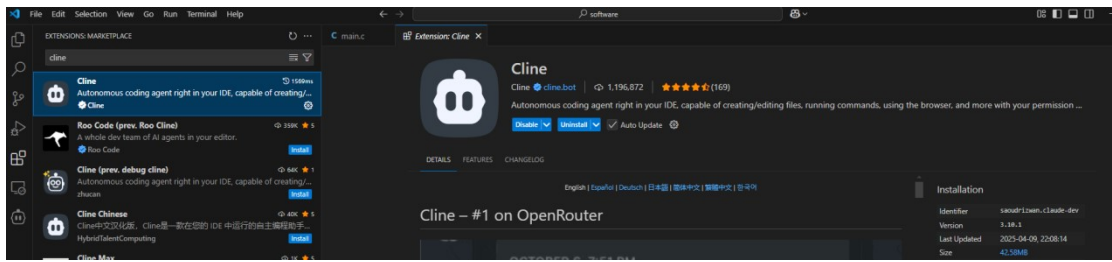


编译出来的结果跟 Keil 是一样的。



可以看到编译出来的信息是正确的。

在 VS Code 中安装 Cline

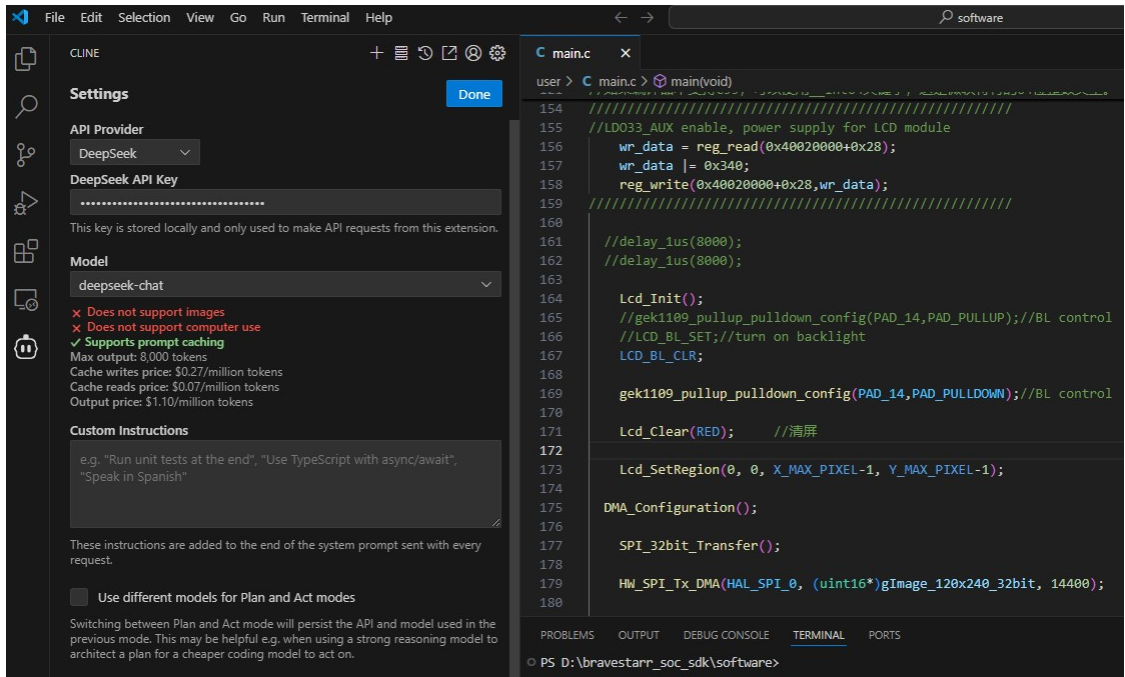


Cine 是 Visual Studio Code 中非常重要的一个插件，可以认为是一个 MCP 的 Client。

同时我们需要在 Cline 的 Setting 菜单中，添加你期望接入的大模型，在这里，我们选择 DeepSeek API 接入。

3 接入 DeepSeek

推荐直接使用 DeepSeek 的 API 接入



在申请 DeepSeek API，获取 API Key 的时候，这个过程是需要付费的，我已经付费试用过了。



申请 API Keys 的地址: https://platform.deepseek.com/api_keys

deepseek 开放平台

用量信息

API keys

充值

账单

API keys

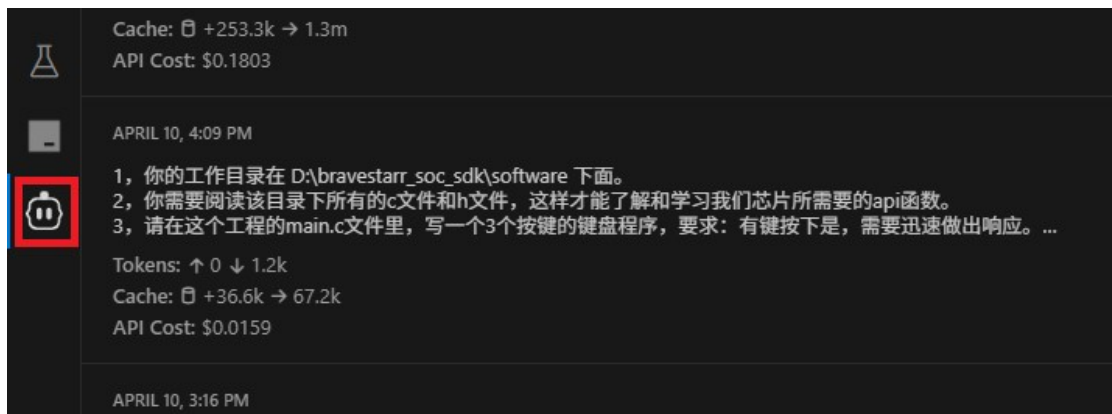
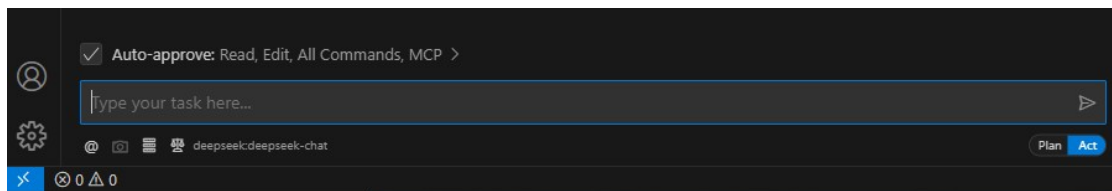
列表内是你的全部 API key，API key 仅在创建时可见可复制，请妥善保管。不要与他人共享你的 API key，或将其暴露在你的浏览器或其他客户端代码中。为了保护你的帐户安全，我们可能会自动禁用我们发现已公开泄露的 API key。我们未对 2024 年 4 月 25 日前创建的 API key 的使用情况进行追踪。

名称	Key	创建日期	最后使用日期
13817016973'Key	sk-ffe1e*****1405	2025-04-08	2025-04-22

创建 API key

Cline 结合 DeepSeek 写代码还是非常快，对代码的理解比人快。

下面我们让 Cline 来调用 DeepSeek 在我们的 1109 sdk 的基础，按照我们的要求，自动完成我们代码需求：

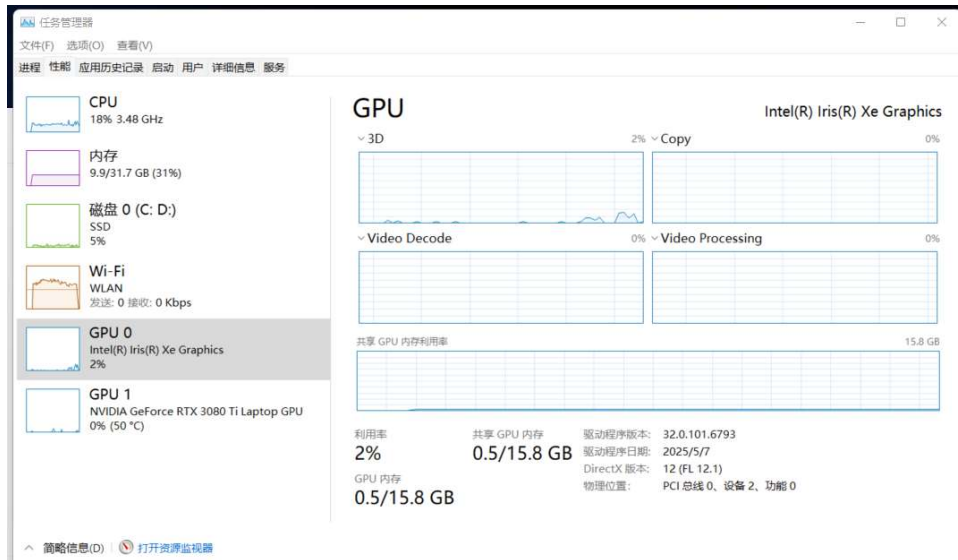


上面的这个提示词，是我给 Cline 下达新任务写的，工程师可以参考我这个提示词，构建新的任务提示词。

Cline 完成代码之后，可以在 Keil 中下载代码到 1901 芯片中，验证是否已经符合工程师的要求。

4 本地部署人工智能大模型

我们也可以本地部署大模型，本地部署的好处是省去了 token 费用，不需要支付调用大模型的年费，同时所有的数据都在本地，安全性和保密性更好。我的本地电脑硬件是这样的：



```
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe

Welcome to Ollama!

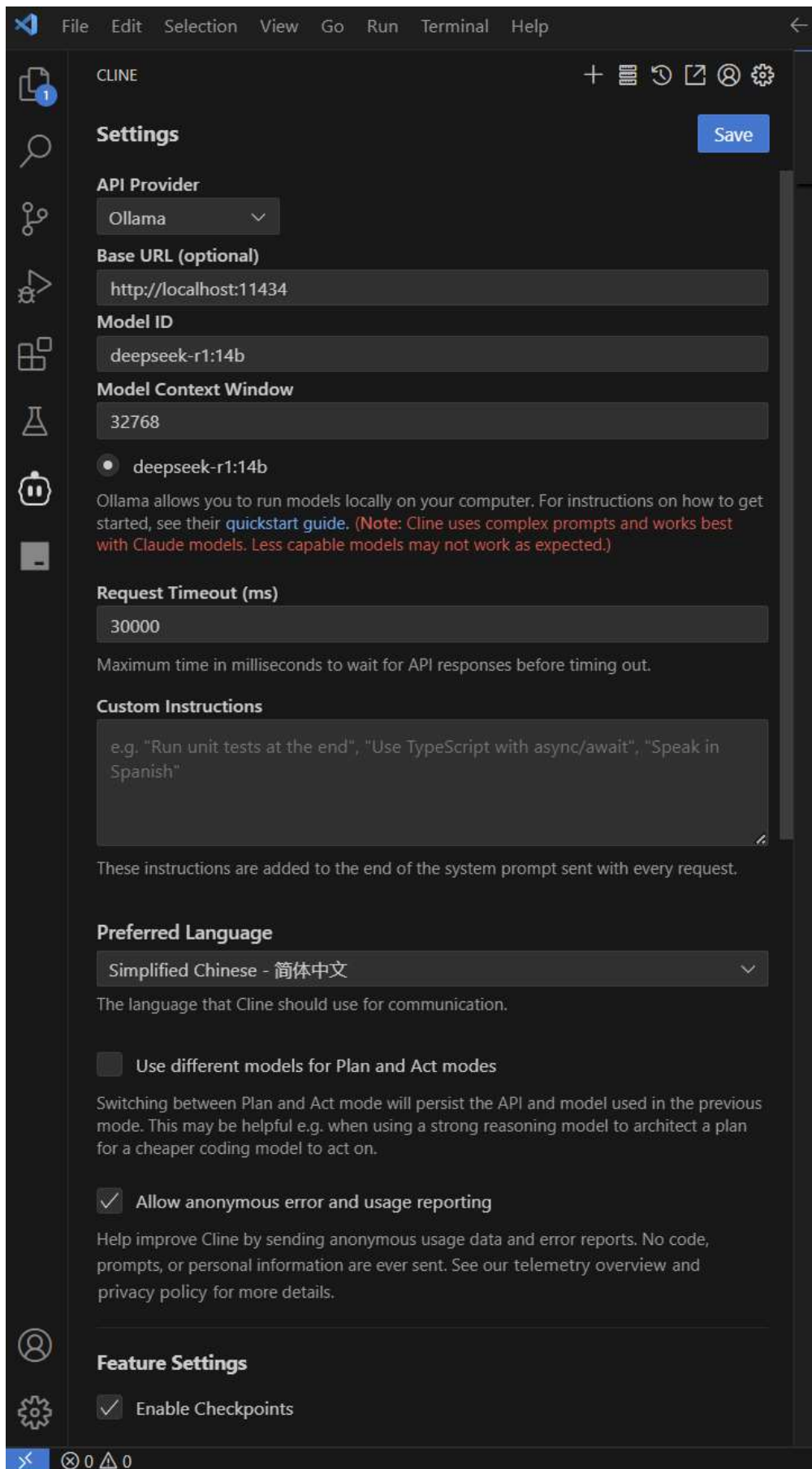
Run your first model:

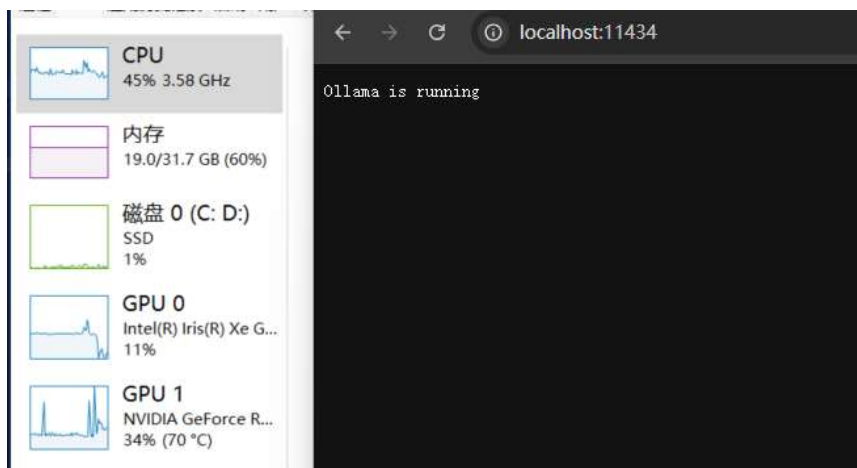
ollama run llama3.2

PS C:\WINDOWS\system32> ollama -v
ollama version is 0.7.0
PS C:\WINDOWS\system32> ollama run deepseek-r1:14b
pulling manifest
pulling 6e9f90f02bb3: 49% | 4.4 GB/9.0 GB 77 MB/s 59s_
```

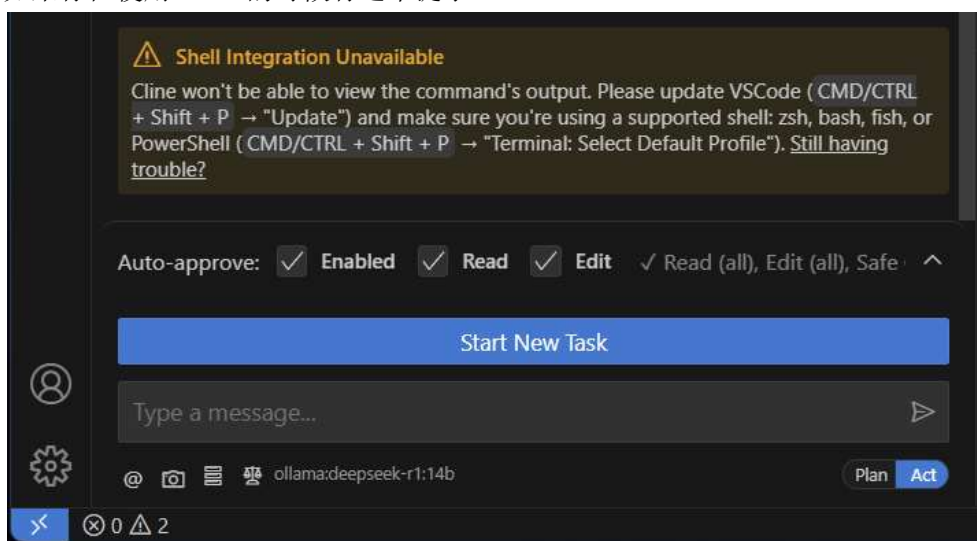
```
PS C:\Users\ALIENWARE> ollama -v
ollama version is 0.7.0
PS C:\Users\ALIENWARE>
PS C:\Users\ALIENWARE> ollama run qwen3:14b
pulling manifest
pulling a8cc1361f314: 17% | 1.6 GB/9.3 GB 64 MB/s 1m59s_
```

```
PS C:\Users\ALIENWARE> ollama -v
ollama version is 0.7.0
PS C:\Users\ALIENWARE>
PS C:\Users\ALIENWARE>
PS C:\Users\ALIENWARE> ollama run qwen3:14b
pulling manifest
pulling a8cc1361f314: 100% | 9.3 GB
pulling eb4402837c78: 100% | 1.5 KB
pulling d18a5cc71b84: 100% | 11 KB
pulling cff3f395ef37: 100% | 120 B
pulling 78b3b822087d: 100% | 488 B
verifying sha256 digest
writing manifest
success
```

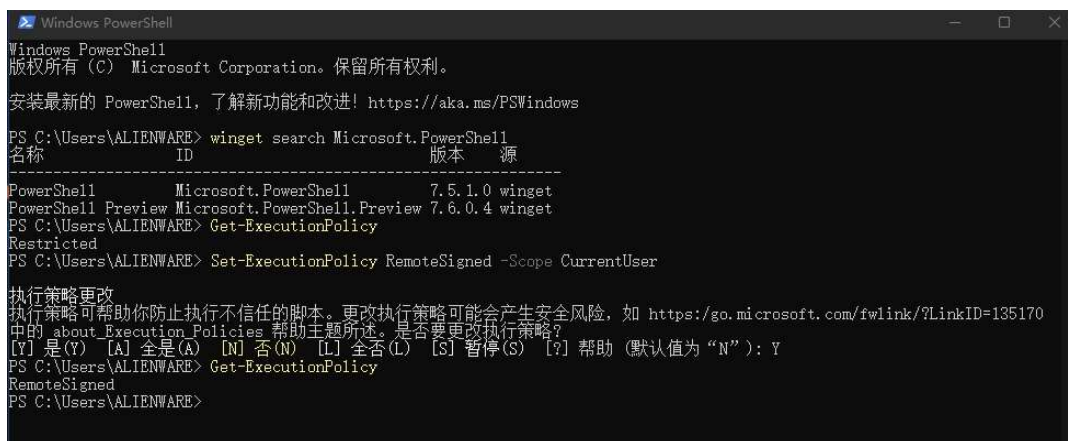




如果你在使用 Cline 的时候有这个提示:



这是因为 Windows PowerShell 的安全策略设置导致的，按照下面的设置改一下就可以了。



Get-ExecutionPolicy

对于 VS Code 中的开发工作，一般建议使用 RemoteSigned 策略。它允许本地创建的脚本不受限制地运行，同时维护下载脚本的安全性。

Set-ExecutionPolicy RemoteSigned -Scope CurrentUser

此时策略将会调整为 RemoteSigned