StarFive 2 编译 Home Assistant 2026.8.3 - Author： 阿峰不坏

环境依赖与 Requirement 清单 · Debian trixie/sid · RISC-V riscv64

1 硬件与基础系统

项目	值 / 版本	备注
开发板	StarFive 2 (JH7110)	4 核 RISC-V U74 @ 1.5GHz
内存	3.8 GiB LPDDR4	编译时需 +4 GiB Swap
磁盘	106 GB NVMe	可用 87 GB
Swap	4.0 GiB	`/swapfile`，编译刚需
操作系统	Debian GNU/Linux trixie/sid	riscv64 架构
内核	6.12.5-starfive	`#7OSF SMP`

2 编译工具链

工具	版本	来源
gcc	12.2.0	系统自带
g++	12.2.0	系统自带
GNU Make	4.3	系统自带
CMake	3.25.1	系统自带
pkg-config	1.8.1	系统自带
git	2.40.1	系统自带
rustc	1.98.1	rustup 安装
cargo	1.98.1	rustup 安装
pyenv	最新	用于管理 Python 3.14.2
Cython	3.1.1	手动降级，用于编译 PyAV
setuptools	>=77.0.0, <82	配合 Cython 3.1.1
wheel	最新	—

3 系统级依赖（apt + dpkg 强装）

📦 通过 apt 正常安装 OK

包名	说明
`build-essential`	编译基础套件
`libssl-dev`	OpenSSL 开发头文件
`zlib1g-dev`	zlib 开发头文件
`libbz2-dev`	bzip2 开发头文件
`libreadline-dev`	readline 开发头文件
`libsqlite3-dev`	SQLite 开发头文件
`liblzma-dev`	xz/lzma 开发头文件
`libncursesw5-dev`	ncurses 宽字符开发头文件
`libgdbm-dev`	GDBM 开发头文件
`uuid-dev`	UUID 开发头文件
`libtiff-dev`	TIFF 图像库开发头文件
`libwebp-dev`	WebP 图像库开发头文件
`libxml2-dev`	libxml2 开发头文件
`libxslt1-dev`	libxslt 开发头文件

`libturbojpeg0-dev`	TurboJPEG 开发头文件
`libffi-dev`	libffi 开发头文件
`tk-dev`	Tk 开发头文件（pyenv 依赖）
`curl`	下载工具
`xz-utils`	xz 压缩工具

🔧 通过 dpkg --force-depends 强装 绕过冲突

包名	版本	说明
`libnss3`	2:3.129-1	覆盖 StarFive 的 `-SF2` 版本
`libnss3-dev`	2:3.129-1	匹配运行时库
`libnspr4-dev`	2:4.35-1.1	NSS 依赖

4 Python 环境

项目	版本	说明
系统 Python	3.11.4	仅用于 pyenv 引导
编译用 Python	3.14.2	pyenv 源码编译

虚拟环境	`~/core/venv`	基于 Python 3.14.2
pip	最新	venv 内升级
isal	1.8.0	额外安装，加速压缩

5 多媒体栈（FFmpeg + PyAV）

组件	版本	安装方式 / 备注
FFmpeg	7.1	源码编译到 `/usr/local`
libavcodec	61.19.100	PyAV 链接成功
libavformat	61.7.100	—
libavutil	59.39.100	—
libswscale	8.3.100	—
PyAV	17.0.1	手动修补源码后编译

> 🔧 **PYAV 源码修补要点**
>
> - 删除 `av/option.py` 第 7 行多余 `cimport`
> - `av/video/reformatter.py`：`sws_free_context` → `sws_freeContext`
> - `av/video/reformatter.pxd`：函数签名同步修改

- 注释掉对 `SwsContext` 内部成员 `threads` / `flags` 的访问
- 编译命令：`pip install . --no-build-isolation --ffmpeg-dir=/usr/local`

6 Home Assistant 本体与核心依赖

组件	版本 / 来源	说明
Home Assistant Core	2026.8.3	`git checkout 2026.8.3`
安装方式	`pip install -e . --no-deps`	editable 安装
核心依赖	`requirements.txt`	全部成功编译
启动命令	`hass -c ~/ha-config`	配置目录 `/home/user/ha-config`

📋 requirements.txt 中成功编译的关键包（节选）

包名	版本	备注
`grpcio`	1.78.0	RISC-V 源码编译
`numpy`	2.3.2	RISC-V 源码编译
`cryptography`	48.0.1	RISC-V 源码编译
`orjson`	3.11.9	Rust 扩展，编译成功

`cffi`	2.0.0	—
`PyNaCl`	1.6.2	—
`bcrypt`	5.0.0	—
`ciso8601`	2.3.3	—
`fnv-hash-fast`	2.0.3	—
`PyYAML`	6.0.3	—
`lru-dict`	1.4.1	—
`Pillow`	12.3.0	—
`pymicro-vad`	1.0.1	—
`pyspeex-noise`	1.0.2	—
`PyTurboJPEG`	1.8.3	—
`ulid-transform`	2.2.9	—
`pycares`	5.0.1	—
`habluetooth`	6.26.5	—
`bluetooth-data-tools`	1.29.18	—
`PyRIC`	0.1.6.3	—

`psutil`	7.2.2	—

7 ESPHome 集成依赖

包名	来源	说明
`aioesphomeapi`	HA esphome 集成 `manifest.json`	按指定版本安装
`esphome-dashboard-api`	同上	—
ESPHome 本体	独立 venv：`~/esphome-venv-new`	最新稳定版，用于编译固件

8 开机自启服务（systemd）

/ETC/SYSTEMD/SYSTEM/HOME-ASSISTANT@USER.SERVICE

```
[Unit]
Description=Home Assistant
After=network-online.target

[Service]
Type=simple
User=%i
ExecStart=/home/user/core/venv/bin/hass -c "/home/user/ha-config"
Restart=on-failure
RestartSec=10
KillMode=process
```

```
TimeoutStopSec=60

[Install]
WantedBy=multi-user.target
```

启用命令：`sudo systemctl enable --now home-assistant@user.service`

已成功运行

9 核心版本一览

✅ 最终验证通过的版本组合

- **Python**：3.14.2（pyenv）
- **FFmpeg**：7.1（`/usr/local`）
- **libavcodec**：61.19.100
- **PyAV**：17.0.1（手动修补）
- **Cython**：3.1.1
- **Home Assistant**：2026.8.3
- **ESPHome 集成**：依赖已安装，兼容 ESPHome 2025.5+

StarFive 2 (JH7110) · Debian trixie/sid · RISC-V riscv64 · 2026 年 9 月26

本清单基于真实编译日志整理，所有命令均可复现 Author：阿峰不坏 整理