

欢迎参加龙架构双周会

• 编辑权限申请

- 计划好主讲的议题和大致用时
- 在本文档申请编辑权限且附上简短的申请理由
- 在龙架构双周会交流群中 **@群主** 或 **管理员** 获取权限
- 向 loongarch@whlug.cn 发送主题为龙架构双周会报告的邮件
- 邮件内请简要说明您将要报告的内容，我们将在收到邮件后同您取得联系，为您提供文档的编辑权限

• 内容编辑

- 请在对应的议题版块下添加您想要分享的内容
- 若无对应议题，请直接在幻灯片其他议题最前方添加
- 快速报告一页控制在 3 分钟以内，报告期间请勿讨论发言
- 专题报告 15~30 分钟，分享结束后可讨论交流

```
.section ".blob", "aw", @progbits
```

```
filestart:
```

```
# e_ident
```

```
.ascii "\177ELF"
```

```
.byte 0x02 # ELFCLASS64
```

```
.byte 0x01 # ELFDATA2LSB
```

```
.byte 0x01 # EV_CURRENT
```

```
.byte 0x00 # ELFOSABI_NONE
```

```
.byte 0x00 # EI_ABIVERSION = 0
```

```
.rept
```

```
.byte
```

```
.endr
```

龙架构双周会

```
# a random base address that's big enough for even 64KiB-page kernels
```

```
.set base_addr, 0x200000
```

2025年5月25日·第12次

```
.short 0x00 # EM_LOONGARCH
```

```
.short 0x102 # EM_LOONGARCH
```

```
.word 1 # e_version = 1
```

```
.dword base_addr + entry - filestart # e_entry
```

```
.dword phdr - filestart # e_phoff
```

```
.dword 0 # e_shoff
```

```
.word 0x41 # objabi v1, soft-float
```

```
.short ehsize # e_ehsize
```

```
.short phentsize # e_phentsize
```

```
.short 1 # e_phnum
```

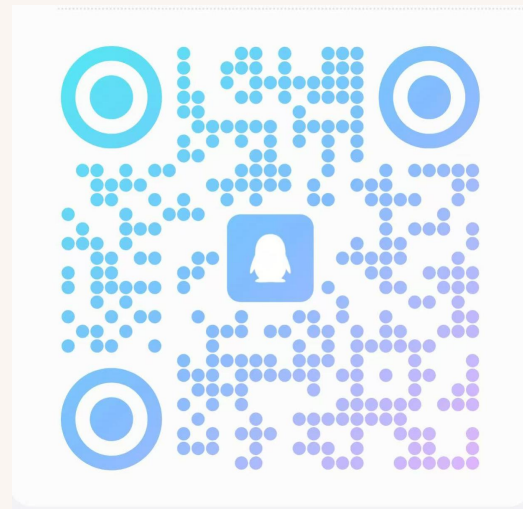
```
.short 0 # e_shentsize
```

```
.short 0 # e_shnum
```

```
.short 0 # e_shstrndx
```

```
.set ehsize, . - filestart
```

```
phdr:
```



龙架构 LoongArch
Biweekly
双周会

```
.section ".blob", "aw", @progbits
```

```
filestart:
```

```
# e_ident
```

```
.ascii "\177ELF"
```

```
.byte 0x02 # ELFCLASS64
```

```
.byte 0x01 # ELFDATA2LSB
```

```
.byte 0x01 # EV_CURRENT
```

```
.byte 0x00 # ELFOSABI_NONE
```

```
.byte 0x00 # EI_ABIVERSION = 0
```

```
.rept 7
```

```
.byte 0
```

```
.endr
```

会前注意事项

```
# a random base address that's big enough for even 64KiB-page kernels
```

```
.set base_addr, 0x200000
```

```
.short 2 # ET_EXEC
```

```
.short 0x102 # EM_LOONGARCH
```

```
.word 1 # e_version = 1
```

```
.dword base_addr + entry - filestart # e_entry
```

```
.dword phdr - filestart # e_phoff
```

```
.dword 0 # e_shoff
```

```
.word 0x41 # objabi v1, soft-float
```

```
.short ehsize # e_ehsize
```

```
.short phentsize # e_phentsize
```

```
.short 1 # e_phnum
```

```
.short 0 # e_shentsize
```

```
.short 0 # e_shnum
```

```
.short 0 # e_shstrndx
```

```
.set ehsize, . - filestart
```

```
phdr:
```

龙架构 LoongArch
Biweekly
双周会

会前注意事项

- 本次会议仅涉及软件技术课题
 - 关于龙芯相关的硬件产品，除官方层面已解禁的消息及本档内可公开的消息外，其他均不作任何回应
- 本次会议与股市无关，不构成任何投资建议

```
.section ".blob", "aw", @progbits
```

```
filestart:
```

```
# e_ident
```

```
.ascii "\177ELF"
```

```
.byte 0x02 # ELFCLASS64
```

```
.byte 0x01 # ELFDATA2LSB
```

```
.byte 0x01 # EV_CURRENT
```

```
.byte 0x00 # ELFOSABI_NONE
```

```
.byte 0x00 # EI_ABIVERSION = 0
```

```
.rept 7
```

```
.byte 0
```

```
.endr
```

快速报告

```
# a random base address that's big enough for even 64KiB-page kernels
```

```
.set base_addr, 0x00000000
```

龙架构上游动向

```
.short 2 # ET_EXEC
```

```
.short 0x102 # EM_LOONGARCH
```

```
.word 1 # e_version = 1
```

```
.dword base_addr + entry - filestart # e_entry
```

```
.dword phdr - filestart # e_phoff
```

```
.dword 0 # e_shoff
```

```
.word 0x41 # objabi v1, soft-float
```

```
.short ehsize # e_ehsize
```

```
.short phentsize # e_phentsize
```

```
.short 1 # e_phnum
```

```
.short 0 # e_shentsize
```

```
.short 0 # e_shnum
```

```
.short 0 # e_shstrndx
```

```
.set ehsize, . - filestart
```

```
phdr:
```

龙架构 LoongArch
Biweekly
双周会

本页预定讲者

GCC: BitInt 支持

- Yang Yujie 提交了 [RFC 补丁](#)，由于需要修改架构无关代码，预期需要较长的评审周期
- 上游开发者 Jakub Jelinek 提出了若干意见，并自己动手修复了架构无关代码中处理 BitInt 左移时未正确对其最高位所在 limb 进行符号/零扩展的问题
- 现有支持 BitInt 的架构 (x86 和 AArch64) ABI 无此要求，故之前未发现此问题
- 而预期 LoongArch 和 S390x 将会有此要求

内核：内嵌汇编中的 `csrxchg`

- 为了节省指令编码空间，`csrxchg` 要求 `rj` 不能为 `r0` 或 `r1`
- 但是内核中有类似 `asm ("csrxchg %0, %1, %2" : "+r"(oldv) : "r"(mask), "i"(csr));` 这样的代码，而 `"r"` 这个限定符却可能将 `mask` 分配到 `r1` 中，在一些内核配置下会产生错误的汇编代码
- 对于 GCC，对 `mask` 改用 `"q"` 限定符 (GCC 12 起即支持，GCC 16 起 [正式记入文档](#)) 即可
- 对于 LLVM 则并不能 (见下一页) ,因此[最后的修复方案](#)不得不硬编码 `t0` 为存放 `mask` 的寄存器

LLVM

- heihier [增加了](#) LA32R 的代码生成支持
- ziyao233 使 Clang 对 LoongArch FPR ABI 名称的支持与 GCC [保持一致](#)
- heihier 解决了将 r0/r1 分配给了 CSRXCHG rj 的[问题](#)
 - [代码生成后端](#)
 - [内联汇编的 q 约束](#)

Haskell

• 历经 4 个月，GHC LoongArch64 NCG 后端于 5 月 14 日 [进入了](#)主线

• 移植作者: Ticat Fp

• CI 协力: lrzlin

```
.section ".blob", "aw", @progbits
file_start
# e_ident
.ascii "\177ELF"
.byte 0x00 # EI_CLASS = ELFCLASS32
.byte 0x00 # EI_DATA = ELF_DATA_LITTLE_ENDIAN
.byte 0x01 # EV_CURRENT
.byte 0x00 # ET_ABIVERSION = 0
.rept 7
.byte 0
.endr

# a random base address that's big enough for even 64KiB-page kernels
.set base_addr, 0x200000

.short 2 # ET_EXEC
.short 0x102 # EM_LOONGARCH
.word 1 # e_version = 1
.dword base_addr + entry - filestart # e_entry
.dword phdr - filestart # e_phoff
.dword 0 # e_shoff
.word 0x41 # objabi v1, soft-float
.short ehsize # e_ehsize
.short phentsize # e_phentsize
.short 1 # e_phnum
.short 0 # e_shentsize
.short 0 # e_shnum
.short 0 # e_shstrndx
.set ehsize, - filestart
phdr:
```

```
.section ".blob", "aw", @progbits
```

Rust

```
file
```

```
# e_ident
```

```
.ascii "\177ELF"
```

- heiherr [制作了](#) PoC, 欢迎试用

```
.byte 0x00 # EV_CURRENT
```

```
.byte 0x00 # ELFOSABI_NONE
```

```
.byte 0x00 # EI_ABIVERSION = 0
```

```
.rept 7
```

```
.byte 0
```

```
.endr
```

```
# a random base address that's big enough for even 64KiB-page kernels
```

```
.set base_addr, 0x200000
```

```
.short 2 # ET_EXEC
```

```
.short 0x102 # EM_LOONGARCH
```

```
.word 1 # e_version = 1
```

```
.dword base_addr + entry - filestart # e_entry
```

```
.dword phdr - filestart # e_phoff
```

```
.dword 0 # e_shoff
```

```
.word 0x41 # objabi v1, soft-float
```

```
.short ehsize # e_ehsize
```

```
.short phentsize # e_phentsize
```

```
.short 1 # e_phnum
```

```
.short 0 # e_shentsize
```

```
.short 0 # e_shnum
```

```
.short 0 # e_shstrndx
```

```
.set ehsize, . - filestart
```

```
phdr:
```

本页预定讲者

龙架构 LoongArch
Biweekly
双周会

Util-linux

- xry111 [修复了](#)由于 Linux ≥ 6.12 的 /proc/cpuinfo 的 ISA 域中 "loongarch32" [改为](#) "loongarch32s" 和 "loongarch32r", 而导致 lscpu 的 “CPU 运行模式” 一项显示错误的问题

```
.section ".blob", "aw", @progbits
file
# e_ident
.ascii "\177ELF"
.byte 0x02 # EV_CURRENT
.byte 0x02 # LOONGARCH
.byte 0x00 # ET_ABIVERSION = 0
.rept 7
.byte 0
.endr

# a random base address that's big enough for even 64KiB-page kernels
.set base_addr, 0x200000

.short 2 # ET_EXEC
.short 0x102 # EM_LOONGARCH
.word 1 # e_version = 1
.dword base_addr + entry - filestart # e_entry
.dword phdr - filestart # e_phoff
.dword 0 # e_shoff
.word 0x41 # objabi v1, soft-float
.short ehsize # e_ehsize
.short phentsize # e_phentsize
.short 1 # e_phnum
.short 0 # e_shentsize
.short 0 # e_shnum
.short 0 # e_shstrndx
.set ehsize, - filestart
phdr:
```

Linux 内核 ([loongarch](#) 列表)

• Huacai Chen

- 在进入和退出 S4 休眠时保存及恢复 CSR.CNTC 寄存器 ([第 1 版](#))
- 为了 KVM 客人内核的 rdttime 读数单调连续性
- 避免将 \$r0/\$r1 作为 csrxcchg 指令的 mask 使用 ([第 3 版](#))
- 修复使用 LLVM 构建内核时的报错
- 回合 (backport) 至 6.1/6.6: 为龙架构内核明确设置 small 代码模型 ([第 2 版](#))
- LoongArch 工具链的默认代码模型不日将扩大至 medium 以便构建大型软件
- 提前 backport 以防新工具链不能构建 LTS 内核

• Tiezhu Yang

- 修缮 uprobe 逻辑 ([第 1 版](#))
- 移除 asm/loongarch.h 中对 larchintrin.h 编译器头文件的引用 ([征求意见](#))

龙架构 LoongArch
Biweekly
双周会

Linux 内核 (loongarch 列表)

• Tiezhu Yang

- 定义 DNAME_INLINE_LEN 的固定值 ([第 1 版](#))
- 为了迁就一个已被替代的过时工具，换新即可 (详见讨论串)
- 6.14 分支：修复 perf 构建错误 ([第 1 版](#))

• Tianyang Zhang

- 龙架构 irq-redirect 支持 ([第 3 版](#))

• Binbin Zhou

- 龙芯 2K0500 BMC 支持 ([第 2 版](#))

• Yao Zi

- 中科云久久派初始支持 ([第 3 版](#))
- 2K0300 SoC 时钟支持 ([第 1 版](#))

```
.section ".blob", "aw", @progbits
```

```
filestart:
```

```
# e_ident
```

```
.ascii "\177ELF"
```

```
.byte 0x02 # ELFCLASS64
```

```
.byte 0x01 # ELFDATA2LSB
```

```
.byte 0x01 # EV_CURRENT
```

```
.byte 0x00 # ELFOSABI_NONE
```

```
.byte 0x00 # EI_ABIVERSION = 0
```

```
.rept 7
```

```
.byte 0
```

```
.endr
```

快速报告

```
# a random base address that's big enough for even 64KiB-page kernels
```

```
.set base_addr, 0x00000000
```

龙架构发行版变动

```
.short 2 # ET_EXEC
```

```
.short 0x102 # EM_LOONGARCH
```

```
.word 1 # e_version = 1
```

```
.dword base_addr + entry - filestart # e_entry
```

```
.dword phdr - filestart # e_phoff
```

```
.dword 0 # e_shoff
```

```
.word 0x41 # objabi v1, soft-float
```

```
.short ehsize # e_ehsize
```

```
.short phentsize # e_phentsize
```

```
.short 1 # e_phnum
```

```
.short 0 # e_shentsize
```

```
.short 0 # e_shnum
```

```
.short 0 # e_shstrndx
```

```
.set ehsize, . - filestart
```

```
phdr:
```

龙架构 LoongArch
Biweekly
双周会

本页预定讲者

安同 OS (AOSC OS)

• 系统减重进展

- 目前已完成静态库的重构，预计减重 ~1.5GiB

oma topics --opt-in static-library-stripping-survey-2025

- 在五月发行之前仍有部分依赖优化

- 清理无用的 Python 及取消 WebKitGTK 预装

- x86-64 镜像已达到 7GiB 水准，龙架构可能依然接近 8GiB

- LATX + x86 运行时较大

• 内核修复：6.14.7 及 6.12.29

- 修复龙芯 3A6000/3B6000/3C6000 系列处理器下处理器拓扑设置错误的问题

• Linux 6.15 内核进展

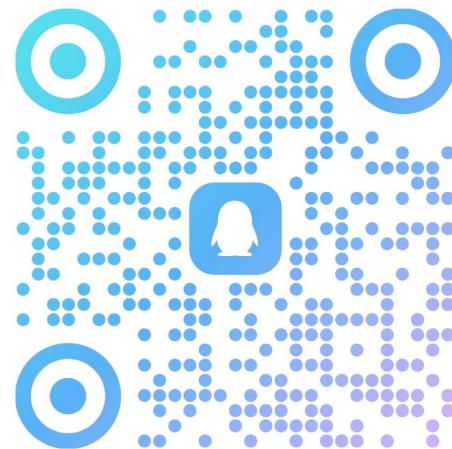
- 配置与补丁集已基本就绪，LoongGPU 有待修复，预计六月推送

• 2K0300 系统构建：修复大量构建问题并修缮构建脚本，稳步推进中



AOSC 社区频道

群号：875059676



扫一扫二维码，入群聊

龙架构 LoongArch
Biweekly
双周会

安同 OS (AOSC OS)

- 本期安全更新

- **GNU Screen 5.0.1**

- 修复 5 个安全漏洞 (低危)

- 默认壁纸投票

- 安同 OS 今年上半年的默认壁纸及锁屏界面投票中

- 将于下周三零点截止

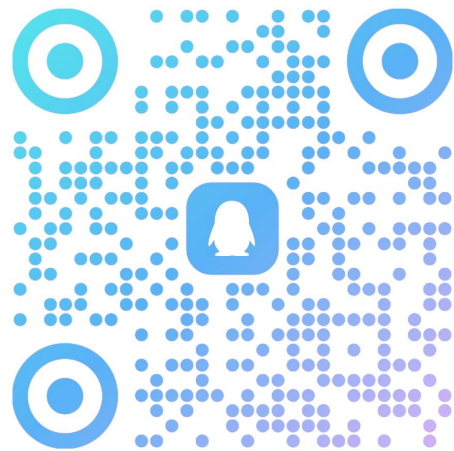
- [点此链接](#)或扫描右下角二维码即可参加投票

- 建议关注公众号“安同开源”或社区主页 (aosc.io) 新闻



AOSC 社区频道

群号: 875059676



NixOS(nix4loong)

• 初步组建核心代码和基础设施维护小组



aleksana



cryolitia



arkyzhou



amforever



wangank

• GNOME/KDE 安装镜像构建已完成，在实机上测试通过

• 构建服务器已重新部署为NixOS

• 默认桌面平台和 no LSX 支持

• 默认 loongarch64-linux 为 LA64v1.0，在构建缓存覆盖范围内

• no LSX 平台可由 pkgsCross.loongarch64-linux-embedded 构建软件

• NixOS 25.11 更新日志将报道 LoongArch64 支持变更

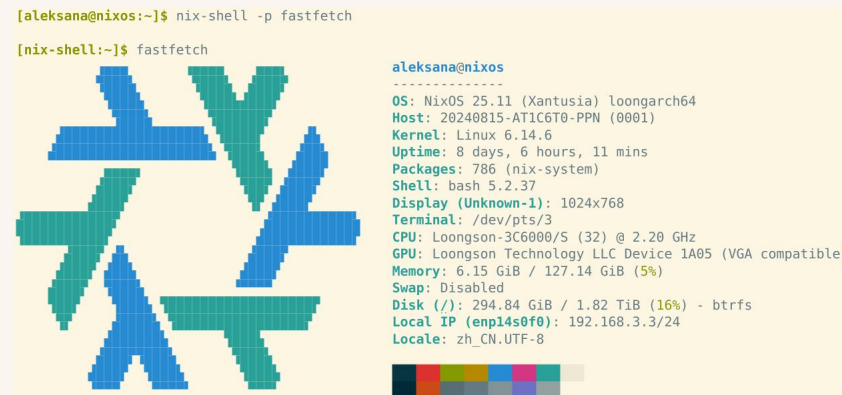
• 正式发布前的准备工作正在进行

• 构建缓存托管服务等待部署，感谢[中山大学开源镜像站](#)

• 域名 nix4loong.cn 已购买，正在备案

• 构建和上传缓存的相关服务正在调试

• 预计于两周后正式发布 NixOS 安装镜像、Nix 安装器、构建缓存和项目文档



龙架构 LoongArch
Biweekly
双周会

Arch Linux for Loong64

- 升级Chromium 136 (by wszqkzqk)

- <https://github.com/lcpu-club/loongarch-packages/pull/579>

- 适配Electron 36 (by wszqkzqk)

- 跟随Chromium上游PlatformArch::kLoong64 -> kLoongarch64

- <https://github.com/lcpu-club/loongarch-packages/pull/580>

- Rust与LLVM 19兼容性修复 (by wszqkzqk)

- 消除LLVM 19不支持特性的warning

- <https://github.com/lcpu-club/loongarch-packages/pull/582>

Arch Linux for Loong64

- gcc14 FP128补丁重新引入 (by Pluto Yang)
 - gcc包滚动升级至gcc15后跟随上游新增维护了单独的gcc14包
 - 重新为gcc14引入了FP128修复
 - github.com/lcpu-club/loongarch-packages/commit/7ba8a16b1561271006e2953d17ece87b7b5f5b62
- 支持python-awkward/syslog-ng (by Pluto Yang)
 - 禁用CUDA相关部分
- zig支持即将上线 (by wszqkzqk)
 - 等待Arch Linux上游从testing移动到stable

```
.section ".blob", "aw", @progbits
```

```
filestart:
```

```
# e_ident
```

```
.ascii "\177ELF"
```

```
.byte 0x02 # ELFCLASS64
```

```
.byte 0x01 # ELFDATA2LSB
```

```
.byte 0x01 # EV_CURRENT
```

```
.byte 0x00 # ELFOSABI_NONE
```

```
.byte 0x00 # EI_ABIVERSION = 0
```

```
.rept 7
```

```
.byte 0
```

```
.endr
```

快速报告

```
# a random base address that's big enough for even 64KiB-page kernels
```

```
.set base_addr, 0x200000
```

社区事务

```
.short 2 # ET_EXEC
```

```
.short 0x102 # EM_LOONGARCH
```

```
.word 1 # e_version = 1
```

```
.dword base_addr + entry - filestart # e_entry
```

```
.dword phdr - filestart # e_phoff
```

```
.dword 0 # e_shoff
```

```
.word 0x41 # objabi v1, soft-float
```

```
.short ehsize # e_ehsize
```

```
.short phentsize # e_phentsize
```

```
.short 1 # e_phnum
```

```
.short 0 # e_shentsize
```

```
.short 0 # e_shnum
```

```
.short 0 # e_shstrndx
```

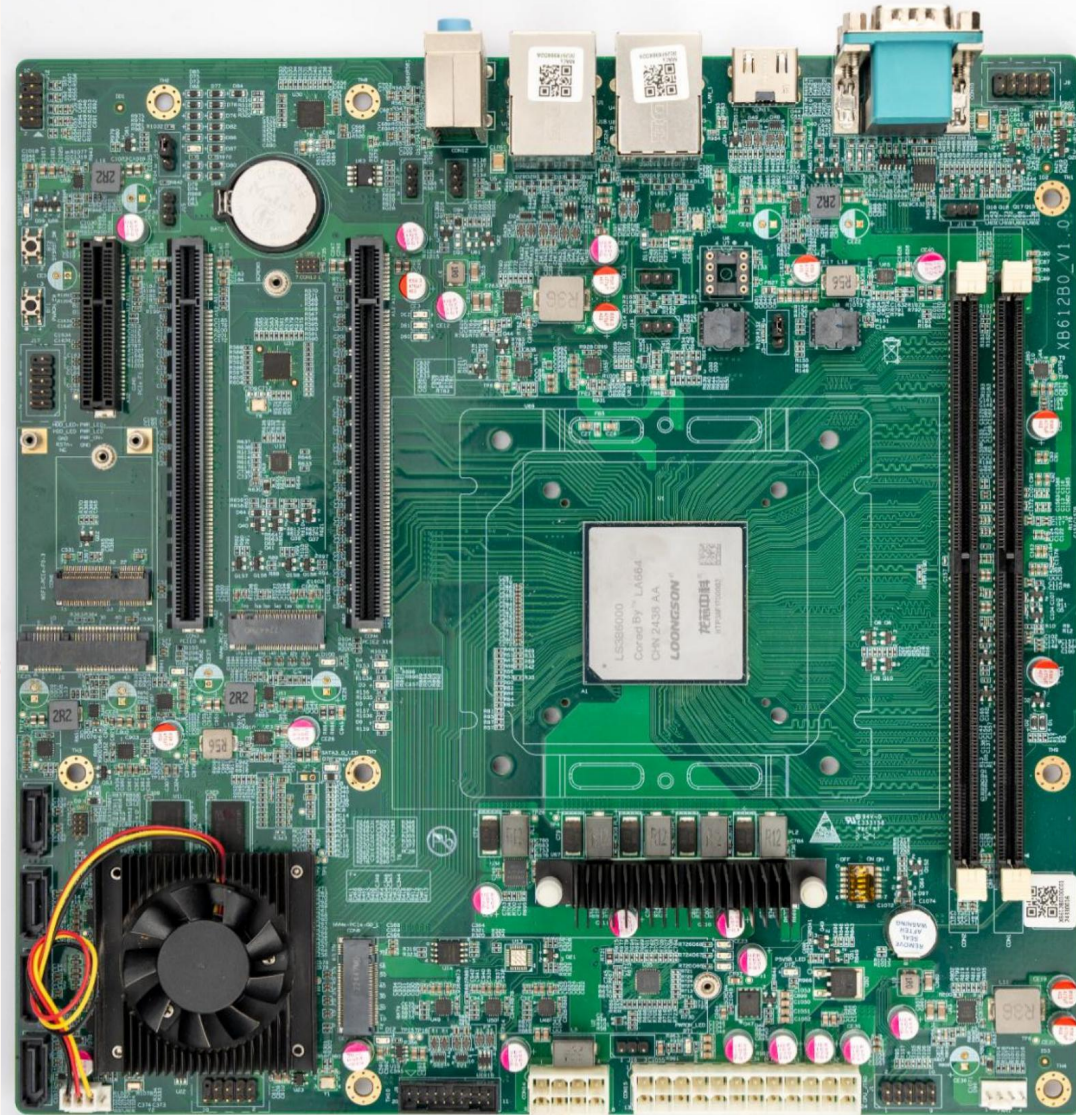
```
.set ehsize, . - filestart
```

```
phdr:
```

龙架构 LoongArch
Biweekly
双周会

3B6000 主板特价团购

- 社区特价 3B6000 主板，数量有限
 - 售价：2799 元
 - 主板型号：XB612B0_V1.1
 - 规格特性：
 - 龙芯 3B6000（八核心），2.3GHz
 - microATX, 244 x 244mm
 - 2 个内存插槽，规格 DDR4 3200MT/s，支持 RECC
 - 三个 PCIe 插槽
 - 两个 x16 槽 (1 * x16, 1 * x8) 均为 CPU 直出
 - 一个 x4 槽，由 7A 桥片引出
 - 两个 m.2 NVMe 插槽，均由 7A2000 桥片引出
 - 2 * GbE 以太网，4 * USB 3.0, 2 * USB 2.0, ...
 - 配件：后挡板及散热器 (120mm)



龙架构 LoongArch
Biweekly
双周会

```
.section ".blob", "aw", @progbits
```

```
filestart:
```

```
# e_ident
```

```
.ascii "\177ELF"
```

```
.byte 0x02 # ELFCLASS64
```

```
.byte 0x01 # ELFDATA2LSB
```

```
.byte 0x01 # EV_CURRENT
```

```
.byte 0x00 # ELFOSABI_NONE
```

```
.byte 0x00 # EI_ABIVERSION = 0
```

```
.repl
```

```
.byt
```

```
.end
```

问答环节

```
# a random base address that's big enough for even 64KiB-page kernels
```

```
.set base_addr, 0x00000000
```

社区问答及意见反馈

```
.short 2 # ET_EXEC
```

```
.short 0x102 # EM_LOONGARCH
```

```
.word 1 # e_version = 1
```

```
.dword base_addr + entry - filestart # e_entry
```

```
.dword phdr - filestart # e_phoff
```

```
.dword 0 # e_shoff
```

```
.word 0x41 # objabi v1, soft-float
```

```
.short ehsize # e_ehsize
```

```
.short phentsize # e_phentsize
```

```
.short 1 # e_phnum
```

```
.short 0 # e_shentsize
```

```
.short 0 # e_shnum
```

```
.short 0 # e_shstrndx
```

```
.set ehsize, . - filestart
```

```
phdr:
```

本页预定讲者

龙架构 LoongArch
Biweekly
双周会

```
.section ".blob", "aw", @progbits
```

```
filestart:
```

```
# e_ident
```

```
.ascii "\177ELF"
```

```
.byte 0x02 # ELFCLASS64
```

```
.byte 0x01 # ELFDATA2L
```

```
.byte 0x01 # EV_CURREN
```

```
.byte 0x00 # ELFOSABI_
```

```
.byte 0x00 # EI_ABIVER
```

```
.rept 7
```

```
.byte 0
```

```
.endr
```

```
# a random base address
```

```
.set base_addr, 0x20000
```

```
.short 2 # ET_EXEC
```

```
.short 0x102 # EM_LOONGARCH
```

```
.word 1 # e_version = 1
```

```
.dword base_addr + entry - filestart # e_entry
```

```
.dword phdr - # e_phdr
```

```
.dword 0 # e_shoff
```

```
.word 0x41 # objabi v1, soft-float
```

```
.short ehsize # e_ehsize
```

```
.short phentsize # e_phentsize
```

```
.short 1 # e_phnum
```

```
.short 0 # e_shentsize
```

```
.short 0 # e_shnum
```

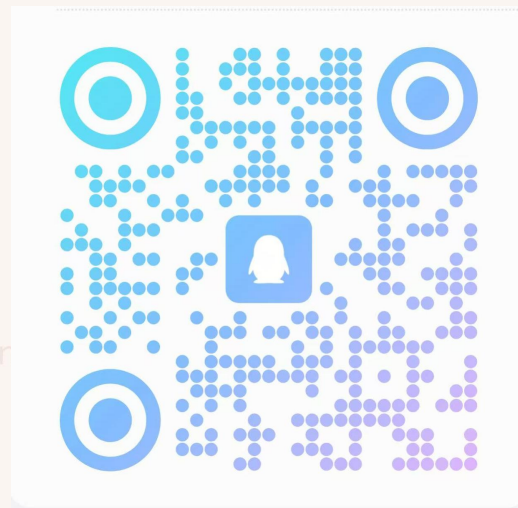
```
.short 0 # e_shstrndx
```

```
.set ehsize, . - filestart
```

```
phdr:
```



双周会讨论 (请先添加管理员)



爱好者交流群

龙架构 LoongArch
Biweekly
双周会