

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

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
```

```
.byte
```

```
.endr
```

龙架构双周会

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

```
.set base_addr, 0x2000000
```

2025年3月30日 · 第8次

```
.short #
```

```
.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 ehsiz # e_ehsiz
```

```
.short phentsiz # e_phentsiz
```

```
.short 1 # e_phnum
```

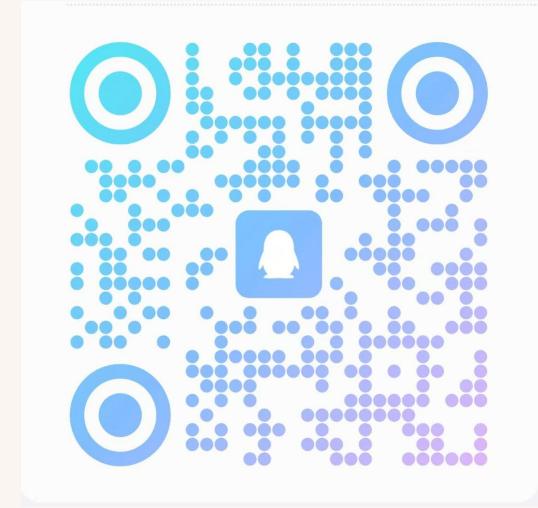
```
.short 0 # e_shentsiz
```

```
.short 0 # e_shnum
```

```
.short 0 # e_shstrndx
```

```
.set ehsiz, . - 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
# a random base address that's big enough for even 64KiB-page kernels
.set base_addr, 0x0100000000000000
龙架构项目进展
.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 ehsizze   # e_ehsizze
.short phentsize # e_phentsize
.short 1         # e_phnum
.short 0         # e_shentsize
.short 0         # e_shnum
.short 0         # e_shstrndx
.set ehsizze, . - filestart
phdr:
```

本页预定讲者

龙架构
双周会
LoongArch
Biweekly

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

手册更新

- 龙架构参考手册（卷一）v1.11 和龙芯架构 32 位精简版参考手册 v1.03 已发布
- 可以在两手册内的“版本历史”一节查阅本次更新作出的各项修改
- 没有增加新的指令或功能

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

```
.set e_shoff, . - filestart
```

```
.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 ehsiz
.short phentsiz
.short 1       # e_phnum
.short 0       # e_shentsize
.short 0       # e_shnum
.short 0       # e_shstrndx
.set ehsiz, . - filestart
```

本页预定讲者

phdr:

龙架构
双周会
LoongArch
Biweekly

```
.section ".blob", "aw", @progbits  
filestart:  
# e_ident  
.ascii "\177ELF"  
.byte 0x01 # ELFCLASS64  
.byte 0x01 # ELFDATA2LSB  
.byte 0x01 # EV_CURRENT  
.byte 0x02 # ET_DYNVERSTON = 0  
.rept 0x00  
.byte 0  
.endr
```

LATX 正式开源

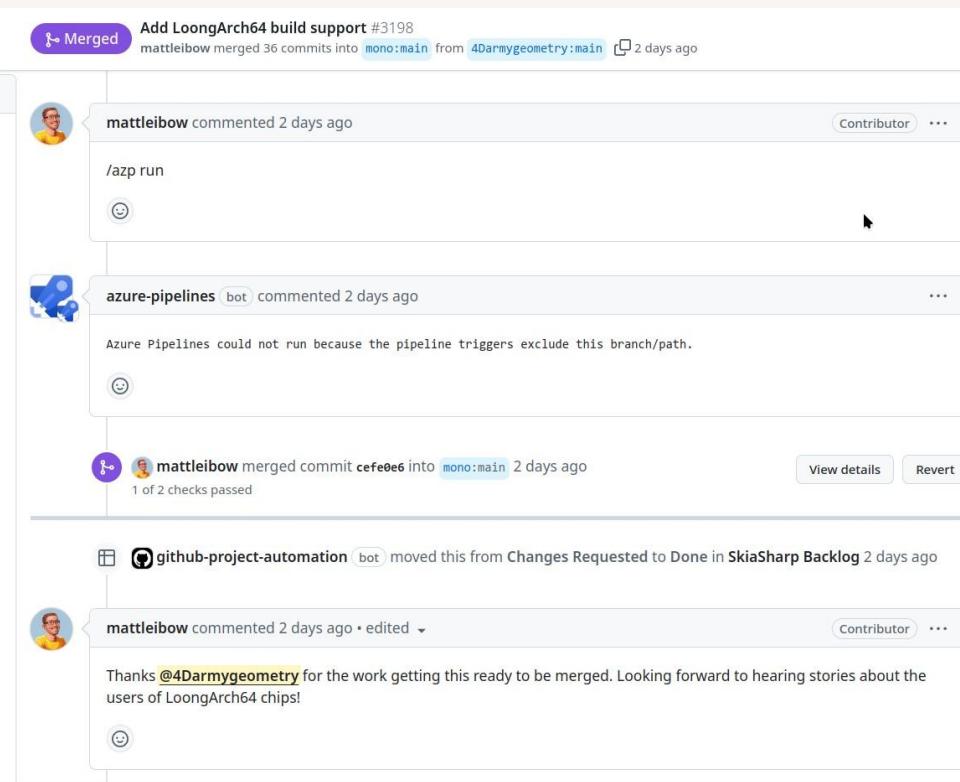
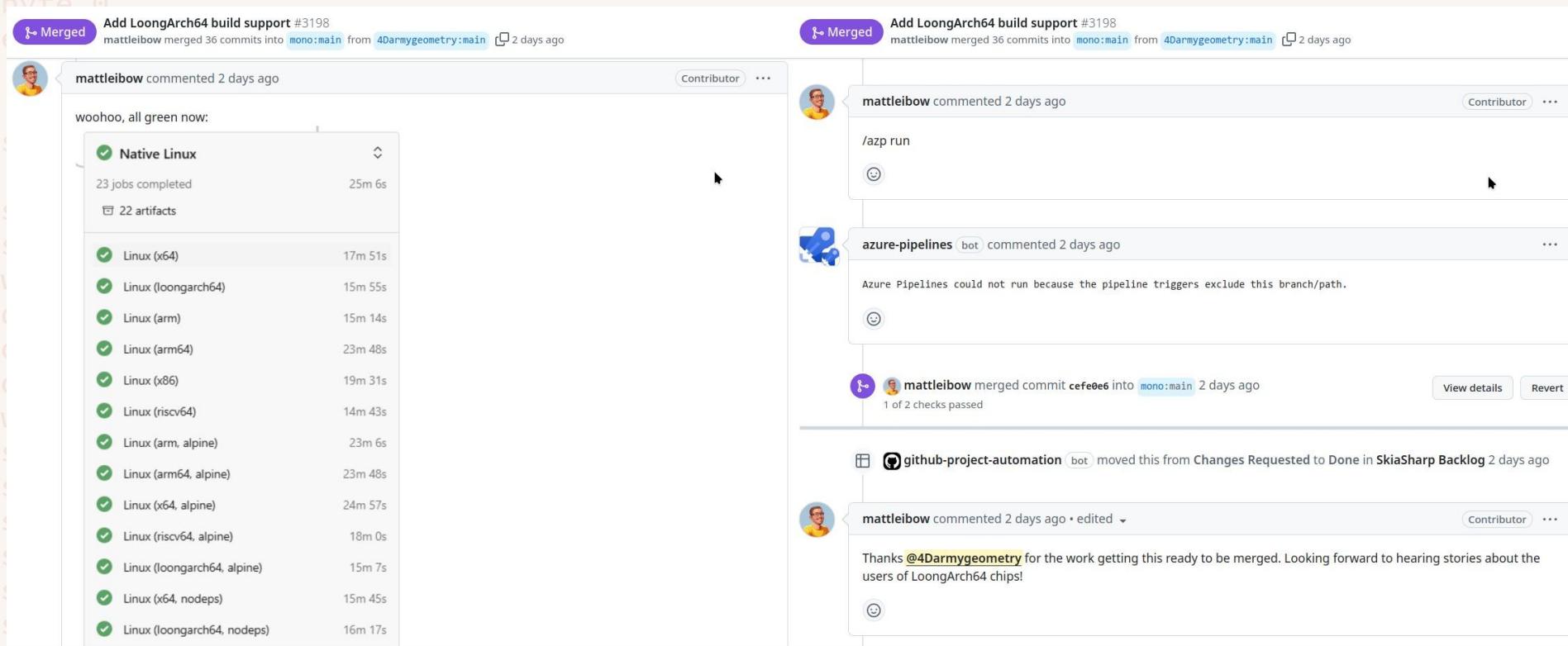
- 3月 26 日，LATX (Loongson Architecture Translator for x86) ，即龙芯 x86 架构转译器，正式发布 1.6.0 版本，并已开源。该工具是一个面向龙架构 (LoongArch) 的高性能用户级二进制翻译器，能够支持在基于龙架构的系统上高效运行 32/64 位 x86 应用程序。
- 该项目遵循 GPLv2 开源协议。

• <https://github.com/lat-opensource/lat>

```
.section ".blob", "aw", @progbits
filler
# e_ident
.ascii "\177ELF"
.byte 0x02 # ELFCLASS64
.byte 0x01 # EV_CURRENT
.byte 0x02 # ET_REL
.set ehsize, . - filestart
phdr:
```

SkiaSharp 和 HarfBuzzSharp 支持龙架构

- 4Darmygeometry 仿照 kasperk81 添加 RISC-V 支持代码的格式给 mono/SkiaSharp 项目添加龙架构新世界编译支持代码，并于 2025 年 3 月 25 日编译成功，并入主线



龙架构
LoongArch
Biweekly
双周会

```
.section ".blob", "aw", @progbits  
fill 0x0  
# e_ident  
.ascii "\177ELF"  
.byte 0x01 # EV_CURRENT  
.byte 0x00 # SHT_NOBITS  
.byte 0x00 # FT_ABIVERSION = 0  
.rept 7  
.byte 0  
.endr  
# a random base address that's big enough for even 64KiB-page kernels  
.set base_
```

SkiaSharp和HarfBuzzSharp支持龙架构

- 目前已经在 Azure 自动构建机构建了测试版 SkiaSharp 和 HarfBuzzSharp 的 NuGet 包，可以通过如下命令下载使用：

- SkiaSharp:nuget install SkiaSharp.NativeAssets.Linux -Version 3.119.0-preview.0.38 -Source https://pkgs.dev.azure.com/xamarin/public/_packaging/SkiaSharp/nuget/v3/index.json

- HarfBuzzSharp:nuget install HarfBuzzSharp.NativeAssets.Linux -Version 8.3.1-preview.0.38 https://pkgs.dev.azure.com/xamarin/public/_packaging/SkiaSharp/nuget/v3/index.json

- Azure 构建机构建情况：

- https://dev.azure.com/xamarin/public/_build/results?buildId=138744&view=results
- <https://github.com/mono/SkiaSharp/pull/3198>

```
.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
# a random base address that's big enough for even 64KiB-page kernels
.set _adr, 0x20000000
```

快速报告

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 ehsiz     # e_ehsiz
.short phentsiz  # e_phentsiz
.short 1          # e_phnum
.short 0          # e_shentsiz
.short 0          # e_shnum
.short 0          # e_shstrndx
.set ehsiz, . - filestart
```

本页预定讲者

phdr:

龙架构
双周会
LoongArch
Biweekly

GNU 工具链

GNU 工具链

- 为 binutils 添加更多指令
 - LA32R rdcnt{vl,vh,i}
 - 是 LA32S/LA64 rdtimer 的官方名称
 - 由于单独名称的可读性，是计数器值，哪个是节
 - 另发现 LA64 v1.10 和 v1.11 实际的助记符为 llacq.
 - 相关维护者回复表示这
- 修复 objdump 多个反汇
- 遗漏了一些 return 0 导

本页预定讲者 xen0n

- 为 binutils 添加更多指令别名支持
 - LA32R rdcnt{vl,vh,id}.w
 - 是 LA32S/LA64 rdtimer[lh].w 的单输出寄存器特例，但是龙芯给了它们单独的官方名称
 - 由于单独名称的可读性更佳（谁能 1 秒说出 rdtimer 的 rd/rj 两个操作数，哪个是计数器值，哪个是节点 ID?），在反汇编时也支持输出该组别名
 - 另发现 LA64 v1.10 和 v1.11 中 ll.acq.[wd]、sc.rel.[wd] 在汇编器中实际的助记符为 llacq.[wd]、screl.[wd]（少了一个“.”）
 - 相关维护者回复表示这是手册中的书写错误，将在下一版手册修复
 - 修复 objdump 多个反汇编选项只有第一个生效的问题
 - 遗漏了一些 return 0 导致第一个选项的处理被误认为失败

龙架构 双周会

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

GCC

- 根据上次双周会提出的意见，增加了对浮点寄存器 ABI 别名 (fa0-fa7, ft0-ft15, fs0-fs7) 的支持 ([r15-8284](#))

- 为了兼容存量代码并与 GCC 文档相一致，增加了浮点常量值非标准后缀 q、Q 的支持 ([PR119408](#), [r15-8962](#), [r14-11456](#))

- 建议新代码使用 C23 风格的 _Float128 类型和对应的 f128 后缀，而非标准化前 GCC 为一些架构提供的 __float128 和 q/Q 后缀

- 根据实测数据调整了默认的代码对齐 ([r15-9022](#))

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

LLVM

- 支持将 b{lt,gt,le,ge}z 反汇编为相同形式，便于阅读
- 支持反汇编选项 no-aliases
- 支持重定位浮点/向量访存指令 {f,v,xv}{ld,st}
- 将 LA64 平台默认的代码模型扩大到 medium
- SIMD 优化
- 向量截断
- 向量符号扩展/零扩展

本页预定讲者 xenOn

```
.section ".blob", "aw", @progbits
file_start:
# e_ident
.ascii "\177ELF"
.byte ELFCLASS64
.byte ELFDATA2LSB
.byte 0x01 # EV_CURRENT
.byte 0x00 # EI_ABIVERSION = 0
.rept 7
.byte 0
.endr
# a random base address is enough for even 64KiB-page kernels
.set base_addr, 0x2000000

.short 2      # ET_EXEC
.short 0x102 # EM_LOONGARCH
.word 0        # version = 1
.dword base_addr + entry - filestart # e_entry
.dword hdr_size # 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:
```

QEMU

- 新特性

1. [cpu hotplug](#)

2. [paravirt ipi](#)

3. [kvm steal time](#)

4. [virtual extioi feature](#)

- 后续支持

1. [3A6000新特性 新指令 新的中断模型 ptw仿真](#)

2. [bug修复](#)

```
.section ".blob", "aw", @progbits  
filestart  
# e_ident  
.ascii "\177ELF"  
.byte 0x01 # ELFCLASS64  
.byte 0x02 # ELFDATA2LSB  
.byte 0x00 # ELFOSABI_NONE  
.byte 0x00 # ET_APTEXT  
.byte 0x00 # PT_LOAD  
.rept 7  
.byte 0  
.endr
```

Linux 内核 (loongarch 列表)

- Hengqi Chen

- 修复 BPF 支持代码中 build_prologue() 的差一 (off-by-one) 错误 ([第 1 版](#))
- 使用 move_addr() 实现 BPF_PSEUDO_FUNC ([第 1 版](#))
- 修复 BPF 子程序错误覆盖返回值的问题 ([第 1 版](#))

- Huacai Chen

- 将 ARCH_DMA_MINALIGN (DMA 最小对齐单位, KiB) 设置为 16 ([第 1 版](#))
- 将 I/O 可编程控制器数量提高到 8 ([第 1 版](#))
- 通过为 dml21_copy() 函数新增 FPU 上下文保护, 修复龙架构上无法驱动 AMD RDNA4 显卡的问题 ([第 1 版](#))
- 修复部分龙芯主板 (如华硕 XC-LS3A6M) 上随机丢失 USB 设备的问题 ([第 3 版](#))

- Maoqian Lin

- 修复 fdt_cpu_clk_init() 中一处 refcount 泄漏 ([第 1 版](#))

Linux 内核 (loongarch 列表)

- Bibo Mao

- 提前 KVM 中断处理，优化 I/O 性能（征求意见，[第 2 版](#)）

- Rui Wang

- 修复使用 GCC 构建 Rust 组件的支持（[第 1 版](#)）

- Yuli Wang

- 重构 arch_kgdb_breakpoint()（[第 1 版](#)）

- Zhibang Xie

- 修正 CMDLINE_EXTEND 的帮助说明（[第 1 版](#)）

- Tianyang Zhang

- 新增 irq-redirect (IRQ 重定向) 控制器支持（[第 1 版](#)）

- Qunqin Zhao

- 龙芯安全引擎、TPM 设备支持（[第 6 版](#)）

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
# a random base address that's big enough for even 64KiB-page kernels
.set base_addr, 0x1000000000000000
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 ehsizze   # e_ehsizze
.short phentsize # e_phentsize
.short 1         # e_phnum
.short 0         # e_shentsize
.short 0         # e_shnum
.short 0         # e_shstrndx
.set ehsizze, . - filestart
phdr:
```

本页预定讲者

龙架构
双周会
LoongArch
Biweekly

```
.section ".blob", "aw", @progbits
file_start:                                # e_ident
# E_I_DENT
.ascii "\177ELF"
.byte 0x01 # ELFCLASS64
.byte 0x01 # ELFPLATNESS
.byte 0x01 # EM_CURRENT
.byte 0x00 # ET_ABTOVERSION = 0
.rept 7
.byte 0
.endr
# a random base address for kernel
.set base_addr, 0x2000000
.short 2 # ET_EXEC
.short 0x102 # EM_LITTLE
.word 1 # EI_VERSION
.dword base_addr + entry - filestart # EI_START
.dword phdr - filestart # EI_PHOFF
.dword 0 # e_shoff
.word 0x41 # objabi v1, soft-float
.short ehsize # e_ehsize
.short 0 # e_phnum
.short 0 # e_shentsize
.short 0 # e_shstrndx
.set ehsize, .. - filestart
phdr:
```

Arch Linux for Loong64 (补上次)

- LDC's 16 byte atomic (by kinke, wszqkzqk)
 - test所需的druntime使用了16字节原子操作
 - 未默认链接到libatomic
 - x86_64等架构可以inline为指令，不受影响，但loong会undefined reference
- 修复：现在构建**LDC自身**时，在loong64/riscv64下会默认链接libatomic
 - LDC维护者希望的方式
 - 与gdc存在行为差异：gdc似乎默认尝试将程序链接libatomic
 - Arch Linux中收录的其他D语言程序**均未**使用16字节原子操作，可能影响不大
- 已向后移植并推送升级
 - <https://github.com/ldc-developers/ldc/pull/4864/files>

```
.section ".blob", "aw", @progbits  
file_start:  
# e_ident  
.ascii "\177ELF"  
.byte 0  
.byte 0  
.byte 0x01 # EV_CURRENT  
.byte 0  
.byte 0x00 # ET_APVERSTON = 0  
.rept 7  
.byte 0  
.endr
```

Arch Linux for Loong64

- 切换至全新的软件仓库结构 (by Pluto Yang, wszqkzqk)
 - 跟进Arch Linux上游仓库维护方式
 - 软件包主体文件统一存放在pool/packages下
 - extra/core/extra-testing/core-testing等目录下的软件包为**软链接**

```
# a random base address for even 64KiB-page kernels  
.set base_addr, 0x200000  
.short 2  
.short 0x0  
.word 1  
.dword base_addr - filestart # e_entry  
.dword phdr # e_phoff  
.dword 0 # 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
file_start:
# e_ident
.ascii "\177ELF"
.byte 0x02 # ELFCLASS64
.byte 0x01 # EV_CURRENT
.byte 0x00 # ET_ABSTRACT = 0
.rept 7
.byte 0
.endr
# a random base address that's big enough for even 64KiB-page kernels
.set base=0x1000000000000000
short 2 # ET_EXEC
short 2 # PT_LOAD_GAP
.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 ehsiz
.short phentsiz
.short 1          # e_phnum
.short 0          # e_shentsiz
.short 0          # e_shnum
.short 0          # e_shstrndx
.set ehsiz, . - filestart
phdr:
```

Arch Linux for Loong64

- Debuginfod服务上线 (by Pluto Yang, wszqkzqk)
 - elfutils 项目提供的调试符号分发服务
 - 允许调试工具 (如 gdb) 自动从远程服务器下载所需调试符号
- 开箱即用，只需将 libelf 升级至 0.192-4.1 或更高版本
 - 支持 gdb、 delve、 valgrind、 KDE Crash Report 等主流工具的自动获取符号
- Arch Linux for Loong64 项目 Debuginfod 调试符号服务正式上线!

```
.section ".blob", "aw", @progbits
file_start:
# e_ident
.ascii "\177ELF"
.byte 0x0 # EI_CLASS
.byte 0x0 # EI_DATA
.byte 0x01 # EV_CURRENT
.byte 0x00 # EI_ABIVERSION = 0
.rept 7
# a random blob address that's big enough to cover 3-paged memory
.set base_addr, 0x00000000
.short 0x102 # EM_LOONGARCH
.word 1 # e_version = 1
.dword phdr - filestart # e_phoff
.dword 0 # e_shoff
.word 0x41 # objabi v1, soft-float
.short ehsiz
# e_ehsiz
.short phentsize
# e_phentsize
.short 1
# e_phnum
.short 0
# e_shentsize
.short 0
# e_shnum
.short 0
# e_shstrndx
.set ehsiz, . - filestart
phdr:
```

Arch Linux for Loong64

- 新增Pytorch支持 (CPU/GPU) (by wszqkzqk)
 - python-pytorch: CPU, 无额外优化 (仍有默认的LSX)
 - python-pytorch-opt: CPU, LASX优化
 - python-pytorch-rocm: ROCm GPU, 无额外优化
 - python-pytorch-rocm-opt: ROCm GPU, LASX优化
- 可以按照需要选择对应的版本
 - <https://github.com/lcpu-club/loongarch-packages/pull/537>

```
.section ".blob", "aw", @progbits
file_start:
# e_ident
.ascii "\177ELF"
.byte 0x02 # ELFCLASS64
.byte 0x01 # ELFDATA2LSB
.byte 0x01 # EV_CURRENT
.byte 0x00 # ELF_ABIVERSION = 0
.rept 7
.byte 0
.endr
# a random base address that is big enough for even 64KiB page kernels
.set base_addr=0x1000000000000000
.short 2
.short 0x102 # EM_LOONGARCH
.word 1
.dword base_addr + entry - filestart # e_entry
.dword 0x1000000000000000 # objabi_v1, soft-float
.set phentsize=1
.short phentsize # e_phentsize
.short 1 # e_phnum
.short 0 # e_shentsize
.short 0 # e_shnum
.short 0 # e_shstrndx
.set ehsize, . - filestart
.phdr:
```

Arch Linux for Loong64

- Sleef (Pytorch所用的第三方向量库)
- 尝试PR添加龙架构支持 (by wszqkzqk)
 - fp128切换至标准浮点拓展(_Float128): 解决字面量Q后缀的兼容性问题
 - 维护者响应积极，但考虑clang兼容性，计划兼容_Float128与__float128两套，未合
 - Cheng Lulu已经在gcc中支持fp128字面量Q后缀并backport到gcc 14分支
 - LoongArch的纯C实现支持
 - 合并受阻
- <https://github.com/shibatch/sleef/issues/671>
- <https://github.com/shibatch/sleef/pull/672>

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

Arch Linux for Loong64

- Sleef (Pytorch所用的第三方向量库)
- LoongArch的纯C实现支持
 - PR时明确提示"Our project **collects a fee from you** to approve your PR"
 - 维护者要求赞助其CI用于测试
 - 本人提出将架构支持分级，被维护者回避
 - 应该是确实需要赞助才会予新架构以支持

 **wszqkzqk** commented 5 days ago • edited

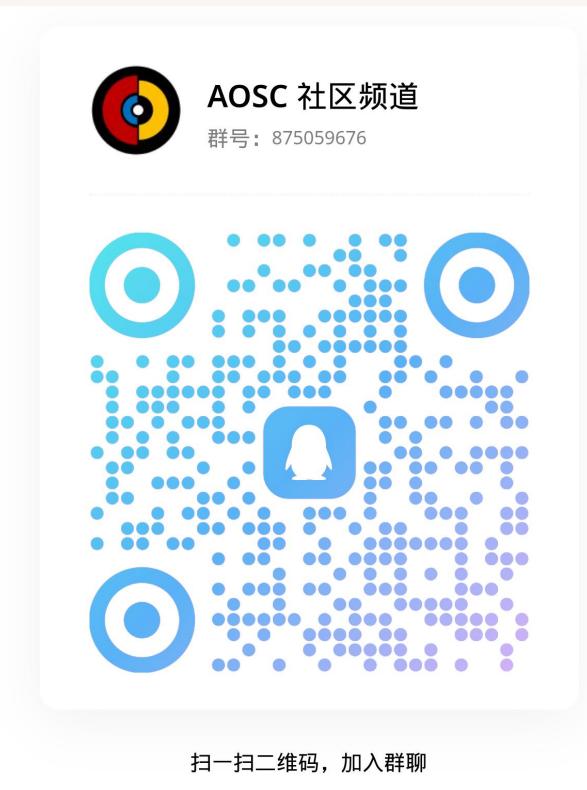
Author ...

I cannot provide ongoing support for free. If no one is willing to cover the cost, please just fork the project and support LoongArch at your repo.

Sigh. As an open source software developer, I completely understand your thoughts. But as a downstream Linux distribution maintainer without financial support, I am also unwilling to bear too much ongoing maintenance costs, so I am unwilling to fork this project. I have already told Loongson about the importance of your project, so let's wait for Loongson's response. 😊😊😊

安同 OS (AOSEC OS)

- 过去两周的工作主要集中在常规维护上
 - AMD RDNA4 显卡 (Radeon RX 9070 及 9070 XT 支持)
 - 未来两周
 - LATEX 1.6 及 Box64 测试
 - ollama 及 ROCm 支持修缮
 - libLoL 软件支持情况排查、更新网站安装教程等
 - Chromium 134，带有完整硬解支持
 - 安全更新与用户公告
 - **社区软件源公钥将在 4 月 10 日迭代，请确保在此日期前至少完整更新一次系统**
 - 自动镜像跳转服务 (redir.aosc.io) 计划于 4 月 5 日正式上线，处理 repo.aosc.io 系统和软件包下载的请求
 - 相信可以提高国内绝大多数地区的软件安装和更新体验
 - 建议关注公众号“安同开源”或社区主页 (aosc.io) 新闻



龙架构 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
# a random base address that's big enough for even 64KiB-page kernels
.set base_addr 0x2000000
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 ehsizze   # e_ehsizze
.short phentsize # e_phentsize
.short 1         # e_phnum
.short 0         # e_shentsize
.short 0         # e_shnum
.short 0         # e_shstrndx
.set ehsizze, . - filestart
phdr:
```

本页预定讲者

龙架构
双周会
LoongArch
Biweekly

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

社区悬赏

```
filestart
```

```
# e_ehdr
```

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

```
.byte 0x02 # EI_ECLASS64
```

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

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

```
.rept 7
```

```
.byte 0x00 # EI_NIDENT
```

```
.endr
```

国家中小学智慧教育平台客户端移植

悬赏任务已基本完成

任务奖品联想开天 3A6000 笔记本已发

```
.endr
```

放到完成者 Katyusha 手中

```
# a random base address that's big enough for the whole file
```

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

```
.short 2      # ET_EXEC
```

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

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

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

```
.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
```

本页预定讲者

phdr:



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

社区悬赏

- 人证对比系统（客户/服务端架构）

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

- 系统采用 Python 3.8 编写系统搭配身份证阅读器（华旭金卡 HX-FDX3S）

```
.rept 7
```

```
.byte 0
```

```
.endr
```

- HX-FDX3S 在 Windows 以外的终端上被识别为密码设备

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

- Windows 上调用的驱动为系统驱动

```
.short 2 # _ET_EXEC
```

- 需要签署 NDA 协议

```
.short 0x00000001 # e_version = 1
```

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

```
.dword phdr_size + entry - filestart # e_shoff
```

```
.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  
filler:  
# e_header  
.ascii "\177ELF"  
.bss :ASSUME64  
htab_start: .INTA2I SR
```

社区徽标
由白铭骢设计



龙芯爱好者社区
2024.10.24

```
.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  
.rep  
.byt  
.end
```

问答环节

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

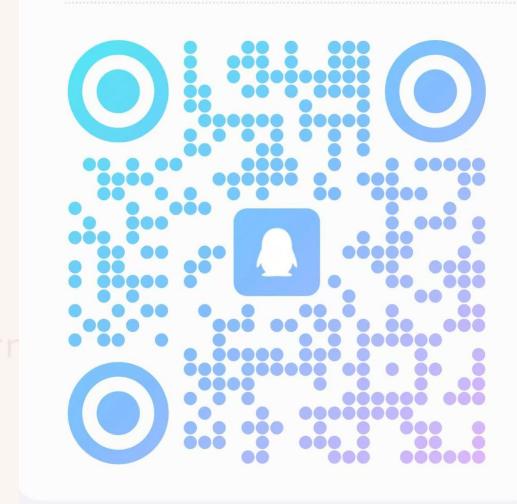
社区问答及意见反馈

```
.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 ehsizze # e_ehsizze  
.short phentsize # e_phentsize  
.short 1        # e_phnum  
.short 0        # e_shentsize  
.short 0        # e_shnum  
.short 0        # e_shstrndx  
set_ehsizze, . - 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_CURRENT  
.byte 0x00 # ELFOSABI_0  
.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 -  
双周会议论 (请先添加管理员)  
.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:
```



双周会议论 (请先添加管理员)

爱好者交流群

龙架构
LoongArch
Biweekly
双周会