

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8月3日·第17次

```
.short 0xc0000000 # e_ident.EI_MAG0 = 0xc0
```

```
.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
```

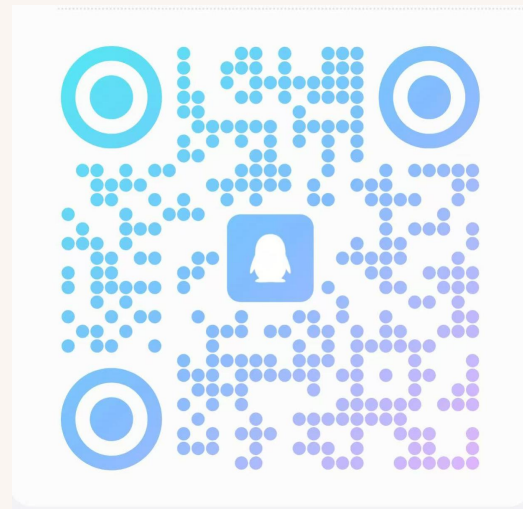
```
.short 0 # e_shentsize
```

```
.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:
```

本页预定讲者

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

binutils

- xry111 [发现](#)并修复了 BFD 链接器会算错符号 relax 后尺寸的问题

- xen0n 理解错了 splay_tree_predecessor 函数用法

- xen0n 向 xry111 支付了一杯咖啡的等价红包

- eu-elflint 是个有用的工具，推荐了解

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

gcc

```
# ident
```

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

```
.byte 0x03 # EI_CLASS = 64
```

```
.byte 0x01 # EI_DATA = LITTLE_ENDIAN
```

```
.byte 0x01 # EI_CURRENT
```

```
.byte 0x00 # EI_OSABI = 0
```

```
.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:
```

- chenglulu: Remove the definition of CASE_VECTOR_SHORTEN_MODE

LLVM

- mintsuki 避免了一条与 triple ABI 有关的多余的警告
- nerfur 新增了龙架构 OpenBSD 构建目标支持!
- 代码生成后端改进
 - zhaoqi5 优化了、又 优化了向量常量的构造方式, 防止了含有 undef 的向量的构造操作被扩展, 改善了往向量中插入浮点元素的做法, 用“比较+选择” 优化了可变下标的插入元素操作
 - hev 修复了拓宽 [x]vmsk{lt,ge,ne}z 操作数时的崩溃
 - Ami-zhang 在 RuntimeDyld JIT 后端 支持了 large 代码模型
- MC、LLD 相关改进
 - #123730, #123743, #151264 (Zhaoxin Yang)

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

.NET

```
file  
# e_ident
```

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

```
.byte 0x04 # EI_CLASS = ELF_CLASS
```

```
.byte 0x01 # EI_DATA = ELF_DATA
```

```
.byte 0x01 # EI_CURRENT
```

```
.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:
```

- LuckyXu [提交了](#)动态处理 LASX/LSX/FPU 上下文的补丁

- 应能解决与 2K0300 等 no-LSX 平台的兼容问题

- ~~截至 07-25, xen0n 指出了实现方式有问题, 待原作者改正~~

- 已于 08-02 合并

Linux 内核 (loongarch 列表)

- Chenghao Duan
 - 为龙架构 BPF 子系统实现函数蹦床 (trampoline) 支持 ([第 5 版](#))
- Wentao Guan
 - 移除 vDSO Makefile 中的 -nostdlib 参数 ([第 1 版](#))
- Takashi Iwai
 - 更新 loongson3_defconfig 中 HD Audio 相关配置 ([第 2 版](#))
- Haoran Jiang
 - 修复 BPF 系统中的两处尾呼叫 (tail call) 问题 ([第 5 版](#))
- Bibo Mao
 - 使用 ELF 程序头实现内核物理内存地址探测 ([第 3 版](#))
- Steven Rostedt
 - 将龙架构 kvm_iocsr 追踪宏定义挪入龙架构专属代码 ([第 1 版](#))
- Binbin Zhou
 - 2K0500/2K1000 NAND 控制器支持 ([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

Arch Linux for loong64

- 上游接纳与补丁精简:

- WebKit & WPEBackend:

webkit2gtk, webkit2gtk-4.1, webkitgtk-6.0 和 wpewebkit 依赖的 Skia 引擎中, 关于 LoongArch LASX 扩展支持的补丁已被上游合并。因此, 我们移除了相关的本地补丁, 使得打包过程更加简洁。这对于所有使用 WebKit 渲染引擎的应用来说都是一个好消息。(by wszqkzqk & Pluto Yang)

- Firefox:

firefox-developer-edition 的一个关于 Loong64 汇编器的修复已被上游 Mozilla 接纳, 我们相应地移除了旧的修复补丁。(by wszqkzqk)

- ONNX Runtime:

onnxruntime 的一个修复补丁 (fixgitlab.patch) 已被上游吸收, 不再需要本地维护。(by wszqkzqk)

- Pixi (conda-like anaconda tool):

此前为 pixi 及其依赖 rattler 添加 LoongArch 支持的补丁, 现已由上游合并并发布新版本。我们的本地补丁随之移除。感谢上游 prefix-dev 和 conda 社区的快速响应! (by wszqkzqk)

- Python-PyArrow:

python-pyarrow 在更新到 21.0.0-1 版本后, 所有测试均已通过, 不再需要任何 LoongArch 相关的本地补丁。这表明 Apache Arrow 项目对 LoongArch 的支持日趋成熟。(by Pluto Yang)

Arch Linux for loong64

- 核心组件与运行时环境的重要更新:

- Node.js 版本回退与测试修复:

我们发现 Node.js 24 在 LoongArch 上存在稳定性问题，会导致 chromium 等多个关键软件包构建或运行时崩溃。为解决此问题，我们采取了以下措施：

- 将主仓库的 nodejs 包降级至 23.x 版本作为临时解决方案，并调整了部分会导致构建失败的测试。(by wszqkzqk)
- 为需要新版 Node.js 特性的包，我们更新了 nodejs-lts-jod (22.18.0) 并修复了其测试套件，使其可以作为一个备选项。(by wszqkzqk)
- chromium 的构建已切换回使用仓库中的 nodejs 23，不再需要依赖 nodejs-lts-jod。(by wszqkzqk)

- Chromium 浏览器持续跟进:

- 修复了 syscall coverage 和 XNNPACK 相关的问题，提升了性能和稳定性。(by wszqkzqk)
- 跟进上游 Arch Linux 的构建方式变更，从手动 git clone 改为从 tarball 构建，并适配了版本 hash 的获取逻辑。(by wszqkzqk)

- devtools-loong64 现已同步上游 devtools 1.4.0 版本，并为 LoongArch64 添加了完善的 Fortran 支持，包括设置了合适的 FFLAGS 编译选项。这对于科学计算和高性能计算社区非常重要。(by wszqkzqk)

Arch Linux for loong64

- 新增软件包支持与疑难问题解决:

- Python-Pandas 依赖问题绕行:

python-pandas 的检查依赖 python-numba 依赖 python-llvmlite, 而后者又需要 llvm15。由于 llvm15 不支持 LoongArch64, 这条依赖链在我们的平台上是断裂的。我们通过构建时移除 python-numba 检查依赖的方式, 成功打包了 python-pandas。(by wszqkzqk)

- Buildbot 与 Node.js 生态:

为了打包新版的 buildbot, 需要其 Web UI 使用较新版本的 rollup 进行原生编译。我们通过 backport 大量的 yarn.lock 文件来解决此问题, 并切换到 nodejs-lts-jod 以绕过 Node.js 24 的崩溃问题。(by Li Bohai)

- libunwind 修复:

我们为 libunwind 添加了补丁, 用于修正 LoongArch64 上的 soname, 并处理了在 nspawn 容器环境中 coredump 测试 xpass 的问题, 增强了系统库的稳定性和 CI 的可靠性。(by wszqkzqk)

- GST-Plugins-rs 构建修复:

为在 LoongArch64 上成功构建 gst-plugins-rs, 我们添加了 cmake 和 clang 作为构建依赖(用于编译 aws-lc-sys)。同时, 由于上游不提供 LoongArch64 的 skia 或 gn 预编译二进制文件, 我们选择跳过了 gst-plugin-skia 插件的构建。(by wszqkzqk)

```
.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

本页预定讲者

开放申请龙架构授权!

- 有消息称：所谓“龙架构授权”，实为中国自动化学会·自主指令系统技术与应用专业委员会（筹）会籍
- 点击就送：去中国自动化学会网站[注册会员](#)，完善个人信息，申请加入这个专委会，缴纳会费（每年 200 CNY，一次性缴纳 10 年会费 = 终身会员）
- 不需要拥有自动化相关专业学位，或从事自动化相关工作
- 单位一栏可以填“个人身份”，职称一栏可以填“无”
- 已知对港澳台、外籍人士无特殊限制（缺数据点）

开放申请龙架构授权？

- 200 块一年的 CAA 会费，就是龙架构授权费吗？
 - 龙芯自己在大陆打官司确立的“指令集不受著作权保护”原则，怎么自己就开始收授权费了？
- 请求第一方证实以下内容：龙架构授权免费，缴费入会获取的是龙架构™相关商标、专利的（可能附条件的）授权
 - 您尽可任意将龙架构实现，但“龙架构™”等称呼是商标
 - 为实现某些语义所需的硬件实现细节被专利覆盖
 - 据 xen0n 2024 年进行的个人研究：LA32 不受专利限制；LA64 基础指令集受 1 条专利限制（边界检查访存相关）；LSX、LBT、LVZ 等第一方专利墙高耸入云
 - 如果您未缴纳授权费，您仍可：以硬件实现未被专利覆盖的部分，并以软件模拟专利部分（虽然对于 SIMD/LBT 这会抹杀性能）
 - 与 RISC-V 情况十分类似：本体只是几本手册，RVI 管理商标

英雄帖：2K3000 视频硬解适配

- 已知此为第三方 IP，猜测为芯原 Hantro VC9000 系列
 - 与 2K1000(LA) 所搭载 VPU 相似性未知（均无细节公开于文档）
- Linux 主线有支持，限于个别 ARM SoC，接口形式为 V4L2
 - 大概率以 PCIe 设备形式存在于龙芯 SoC
 - Loongnix 支持情况未知：手头暂无设备，淘宝客服不知情
- 目前进展：前期准备阶段
 - 已下单一块 2K3000 AI EVB
 - 暂未联系到任何相关人士（龙芯、芯原）
 - 欢迎有志于捍卫消费者权益的同学在[英雄帖](#)下回复！

```
.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_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
```

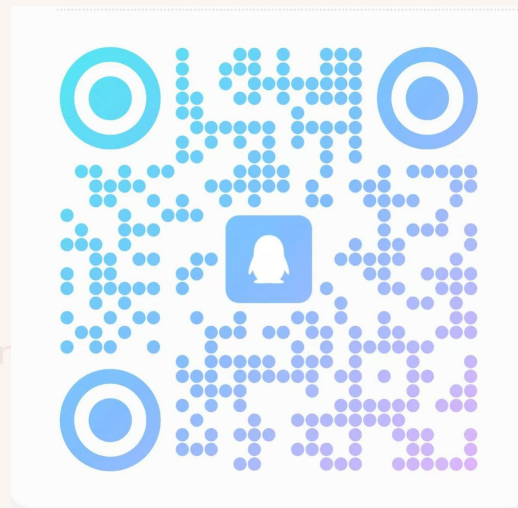
```
.short 0 # e_shstrndx
```

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

```
phdr:
```



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