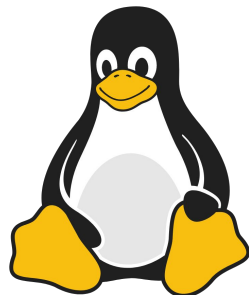


Linux Kernel

Primeiros Passos



Reunião de Férias do
FLUSP: FLOSS at USP



Estrutura da comunidade

Modularização

Exemplos de subsistemas

alsa-devel, autofs, backports, ceph-devel, cgroups, cpufreq, dash, dccp, devicetree-compiler, devicetree-spec, devicetree, dmaengine, dwarves, ecryptfs, fio, fstests, initramfs, irda-users, kernel-janitors, kernel-packagers, kernel-testers, keyrings, kvm-commits, kvm-ia64, kvm-ppc, kvm, lartc, libzbc, linux-8086, linux-acpi, linux-admin, linux-alpha, linux-api, linux-apps, linux-arch, linux-arm-msm, linux-assembly, linux-bbs, linux-bcache, linux-block, linux-bluetooth, linux-btrace, linux-btrfs, linux-c-programming, linux-can, linux-cifs, linux-clk, linux-config, linux-console, linux-coverity, linux-crypto, linux-diald, linux-doc, linux-edac, linux-efi, linux-embedded, linux-ext4, linux-fbdev, linux-fido, linux-fpga, linux-fscrypt, linux-fsdevel, linux-fsf, linux-ftp, linux-gcc, linux-gpio, linux-hams, linux-hexagon, linux-hotplug, linux-hwmon, linux-i2c, linux-ia64, linux-ibcs2, linux-ide, linux-IIO, Linux-input, linux-integrity, linux-ipx, linux-isdn, linux-japanese, linux-kbuild, linux-kernel-announce, linux-kernel-announce.posters, linux-kernel, linux-kselftest, linux-laptop, linux-leds, linux-linuxss, linux-lugnuts, linux-m68k-cvscommit, linux-m68k, linux-man, linux-mca, linux-media, linux-metag, linux-mmc, linux-modules, linux-msdos-devel...

Mantenedores

- Zelam pelo código de um subsistema: revisam/aprovam contribuições, gerenciam as branches, etc.
- O arquivo MAINTAINERS: lista de mantenedores
- Recuperando um mantenedor: `kw m` ou [scripts/get_maintainers.pl](#)

Repositórios

- <https://git.kernel.org/pub/scm/linux/kernel/git/>
- Algumas organizações têm seus próprios repositórios oficiais.
- Dica: Observe as branches do repositório e descubra em qual trabalhar. Às vezes a mais atual não é a master!

Comunicação

- Lista de emails, IRC, etc. (Cada subsistema tem suas particularidades)
- Inscrever-se em uma ou mais listas do seu interesse
 - Conjunto de todas as listas: <http://vger.kernel.org/vger-lists.html>
- Contribuições são enviadas e revisadas nas listas
- Archives: LKML, MARC, etc.
- **Importante: O padrão nas listas é plain text e resposta inline**

On Mon, 2018-11-05 at 16:59 -0200, Lucas Santos wrote:

> Hello,
>

Hey,

Sorry for the late reply.
I deferred answering your email, and nearly forgot.

> I'm looking at the code of impedaance-analyzer/ad5933, trying to
> understand
> what is needed to move it to the main tree. I am part of a study group
> and we
> are willing to work on it.

Cool :)
Thanks

Revisões

- Feitas pelos mantenedores e outros contribuidores
- Corrija e reenvie utilizando os prefixos v2, v3, ...
- **Dica: Espere alguns dias antes de enviar novas versões, para receber mais revisões**

Processo de Desenvolvimento

Timeline de desenvolvimento

1. Encontrar onde contribuir
2. Escrever o código
3. Testar o código (compilação e possíveis testes)
4. Verificar o codestyle
5. Escrever uma boa mensagem de commit
6. Encontrar os destinatários do patch
7. Enviar o patch
8. Ao receber feedbacks, aplicar as correções e reenviar o patch

Overview do Repositório

- COPYING
- Documentation
- Kbuild
- LICENSES
- MAINTAINERS
- Makefile
- README
- arch
- crypto
- drivers
- fs
- include
- init
- ipc
- kernel
- lib
- mm
- net
- samples
- scripts
- security
- sound
- tools

Mão na Massa: Ambiente e IIO

Configurando ambiente

1. Git
2. Baixar código fonte
3. KW
4. Inscrever-se na mailing list
5. Filtros do Gmail
6. Git send-email
7. Compilação
8. Qemu e VM

Git

- Checar instalação: `git --version`
- Configurar:
 - `git config --global user.name "<your name>"`
`git config --global user.email "<your email>"`

Baixar código fonte

- `git clone https://git.kernel.org/pub/scm/linux/kernel/git/jic23/iio.git`
- Dica: Visite o site acima. É um bom viewer online do repositório

Instalar KW

- <https://github.com/rodrigossiqueira/kworkflow>
- Mudar para a branch “unstable” e rodar `./setup.sh -i`

Inscriver-se na Mailing List + Filtro do Gmail

- <http://vger.kernel.org/vger-lists.html#linux-iio>
 - Email para majordomo@vger.kernel.org
 - Sem assunto
 - Corpo: "subscribe linux-iio"
- Filtros do Gmail

Has the words

Doesn't have



Skip the Inbox (Archive it)

...

Apply the label:

Git send-email

- Configurar:
<https://matheusbernardino.github.io/howto/2018/10/24/using-git-send-email.html>

Compilando o Kernel

- O super arquivo .config
- Como conseguir um .config para a sua máquina:
 - `zcat /proc/config.gz > .config`
 - `cp /boot/config-`uname -r` .config`
- Habilitar/desabilitar módulos: `make nconfig`
- Apenas drivers enabled: `make localmodconfig`
- Compilar: `make ARCH=x86_64 -jX` //X=número de threads
- Referência para compilar/instalar:
<https://oslongjourney.github.io/linux-kernel/Kernel-compilation-and-installation/>

Qemu e VM

- Instalar o Qemu
- Criar VM com arch linux

Considerações no IIO

- Docs: <https://www.kernel.org/doc/html/v4.14/driver-api/iio/index.html>
- Desenvolver na branch 'testing'
- Primeiros passos
 - drivers/staging/iio
 - correções de style
- Posteriormente
 - Escolher um driver com "problemas"
 - Ler datasheet
 - Mãos à obra

Leituras

Leituras importantes

- Sobre o processo de desenvolvimento:
<https://www.kernel.org/doc/html/v4.15/process/2.Process.html> e
<https://www.kernel.org/doc/html/latest/process/submitting-patches.html>
- Primeiro patch: <https://kernelnewbies.org/FirstKernelPatch>
- Boas mensagens de commit: <https://chris.beams.io/posts/git-commit/>
- Linux codestyle:
<https://www.kernel.org/doc/html/v4.10/process/coding-style.html>
- Filosofia do patch: <https://kernelnewbies.org/PatchPhilosophy>
- Docs nos repos: Documentation e Documentation/process