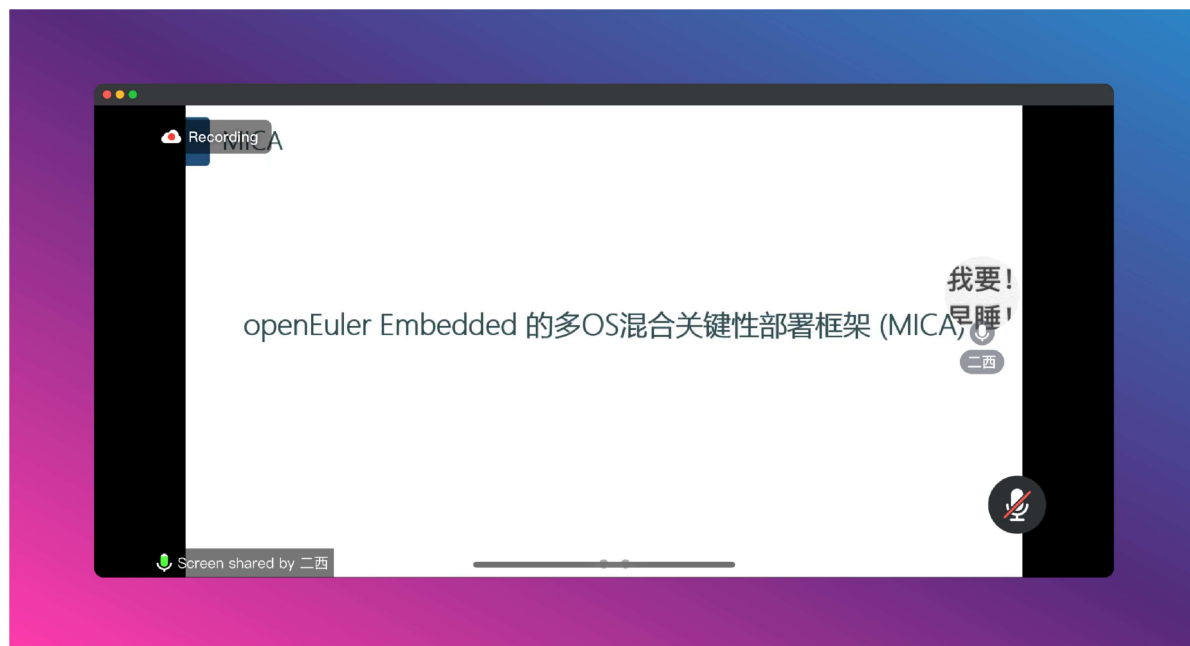


2.29SIG例会记录——>年后第一场

一、MICA

实时操作系统，多底座。统一接口的共享内存。



Recording

MICA实现原理

我要! 早睡! 二西

Screen shared by 二西

多 OS 之间的通信:
通知和响应、共享内存

Linux Core ↔ Share Memory ↔ AMP Core
Linux Core to AMP Core Interrupt
AMP Core to Linux Core Interrupt

问题:

- 不同的部署场景, 通知机制、共享内存的实现发生改变。
通知机制: IPI / hvc / mailbox
- 共享内存: 物理内存 / ivshmem / ...
- 如何高效实现共享内存的管理, 为Linux/RTOS的多服务提供支持
需要有兼容 Linux/ RTOS 的传输层、链路层实现

OpenAMP

传输层: RPMsg

链路层: VirtIO

逻辑层: 共享内存, 通知通道

RemoteProc

物理层: ivshmem 虚拟化, 物理内存 AMP, PCIE 内存空间, IPI, HVC, mailbox, PCIE

Recording

MICA 使用方法

我要! 早睡! 二西

Screen shared by 二西

构建指导:

- 基于 [oebuild 构建](#): oebuild generate → 选择 opapeuler-mos
- 基于 [源码构建](#): cmake -S . -B build; cmake -S . -B build -DCMAKE_BUILD_TYPE=Debug

使用方法:

- 在 [ARM64 GEMU 上运行](#)
- [mica 命令与配置文件](#)

```
Back
Terminal Sessions View X server Tools Games Settings Macros Help
Open command palette (Ctrl+Shift+P)
$ ls
20240229032328
[Build@localhost 15:11:32 ~/work_master/build/mcs_arm64_debug_2-27/output]
$ cd 20240229032328/
[Build@localhost 15:11:34 ~/work_master/build/mcs_arm64_debug_2-27/output/20240229032328]
$ ls
Image-5.10.0-openeuler          vmlinux-5.10.0-openeuler      zImage-5.10.0-openeuler
openeuler-image-mcs-qemu-aarch64-20240229032328.rootfs.cpio.gz  zImage
[Build@localhost 15:11:34 ~/work_master/build/mcs_arm64_debug_2-27/output/20240229032328]
$ pwd
/home/openeuler/hanzongcheng/work_master/build/mcs_arm64_debug_2-27/output/20240229032328
[Build@localhost 15:11:35 ~/work_master/build/mcs_arm64_debug_2-27/output/20240229032328]
$ ls
Image-5.10.0-openeuler          vmlinux-5.10.0-openeuler      zImage-5.10.0-openeuler
openeuler-image-mcs-qemu-aarch64-20240229032328.rootfs.cpio.gz  zImage
[Build@localhost 15:11:39 ~/work_master/build/mcs_arm64_debug_2-27/output/20240229032328]
$ ls
Image-5.10.0-openeuler          vmlinux-5.10.0-openeuler      zImage-5.10.0-openeuler
openeuler-image-mcs-qemu-aarch64-20240229032328.rootfs.cpio.gz  zImage
[Build@localhost 15:24:30 ~/work_master/build/mcs_arm64_debug_2-27/output/20240229032328]
$ cd ^C
[Build@localhost 15:24:31 ~/work_master/build/mcs_arm64_debug_2-27/output/20240229032328]
$ ..
[Build@localhost 15:24:32 ~/work_master/build/mcs_arm64_debug_2-27/output]
$ ..
[Build@localhost 15:24:32 ~/work_master/build/mcs_arm64_debug_2-27]
$ ..
[Build@localhost 15:24:32 ~/work_master/build]
$ ls
3403_qt      mcs_arm64_debug      mcs-arm64-debug_2-29  ok3588      rpi4_mcs_rproc      x86_mcs_219      x86_rt_systemd
jailhouse_mcs  mcs_arm64_debug_2-27  oebuild.log          qt_systemd_x86  x86_mcs             x86_rt_busybox_add_rtconfig  x86_std
[Build@localhost 15:24:33 ~/work_master/build]
$ cd ^C
[Build@localhost 15:24:34 ~/work_master/build]
$ oebuild gen
```

使用的方式：

MICA的相关的

source SDK

cmake -S

cd

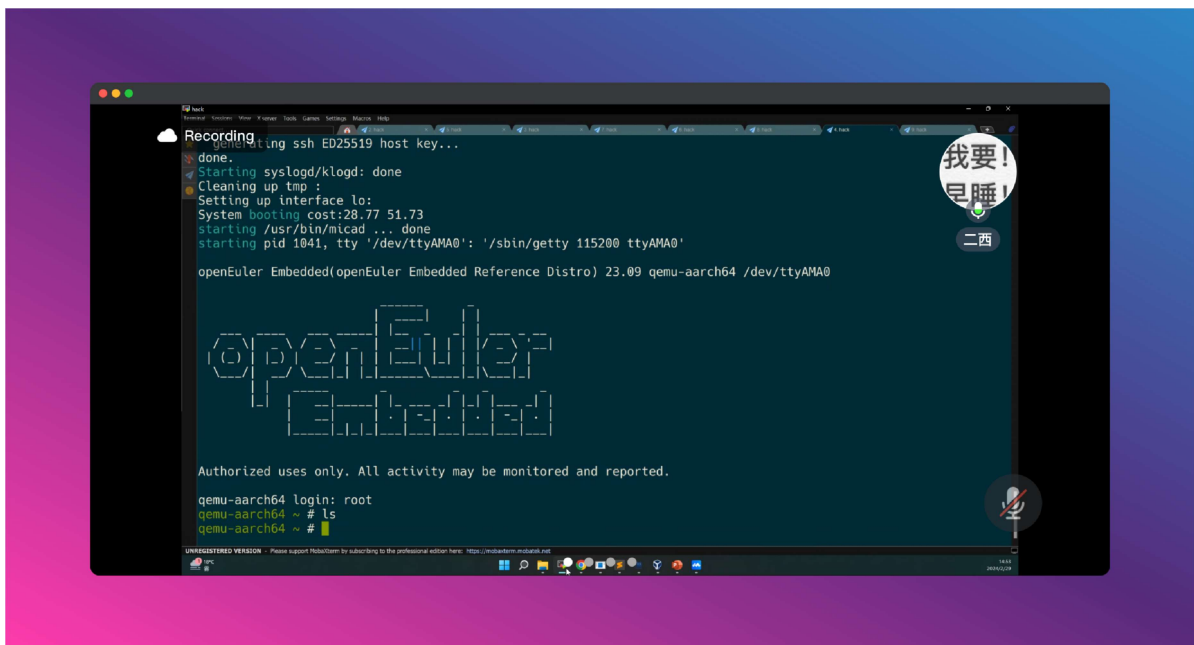
从QEMU角度选

SSH QEMU

ifconfig eth0

ls

gs



文档MICA，使用QEMU部署、RTOS部署等

定义了一个函数

二、近期环节

isula完善和问题修复：

(1) 支持极简容器镜像，引入openeuler-container-os及资料

(2) 1xc版本回退（修复启动问题），嵌入式保留isula极简运行时1xc，暂不升级runc，否则将会引入go语言，较为厚重2、MICA框架重构：

混合部署统一命令管理，为后续虚拟化底座做管理准备

3、重构和BSP完善：(1) 重构meta-hisilicon，兼容openeuler-image镜像 (2) 支持hieulerpi

(ss928/sd3403)：正式合入master，伙伴正在完善各项配套，3月底将在南京meetup正式发布 (3) 支持hi3093：支持直接在openeuler-image集成海思BSP生成emc部署镜像，同时支持不集成BSP驱动的镜像（可后续通过海思解决方案工程进一步打包）

1. 新增登录打印：openEuler Embedded字符终端LOGO打印
2. 其他优化重构：openeuler_source重构，优化了构建使用openeuler_source列表时的构建空间占用openeuler-image配方重构，更加方便的配置镜像不用手动配置
OPENEULER_SRC_URI_REMOVE，当配方对应的仓库在manifest中存在时，会自动移除外部http、https、git源
3. 当前待修复关键问题：master编译器升级后，已知存在strip失败问题，导致镜像增大，近期即将修复其他问题见记录ISSUE

发言：雪球计划签到(只有我们到了)

NXP的看过之后，这些文件是在19年开发myd-jxmx_4.19.35时用到的文件。

sig/sig-Yocto

里程碑
未关联

分支
未关联

参与者 (4)

+ 添加评论

SOC型号	soc厂商	bsp型号	赞助商	gitee id	进度	完成时间	备注
T113-S3	全志	MYD-YT113S3-4E128D-110-I-G	米尔科技				
T113-I	全志	MYD-YT113i-4E256D-110-I	米尔科技				
T527	全志	MYD-LT527M-16E2D-180-E	米尔科技				
RZ/G2L	瑞萨	MYD-YG2L23-8E1D-120-C-REMI	米尔科技	alichinese			
i.MX 6ULL	NXP	MYD-Y6ULY2-V2-4E512D-50-I	米尔科技	darrenpig			
i.MX 8M Plus	NXP	MYD-JX8MPQ-8E2D-160-I	米尔科技	puai			
STM32MP135	ST	MYD-YF135-4E512D-100-I	米尔科技	bigclouds99			
AM62x	TI	MYD-YM6254-8E2D-140-I	米尔科技	emancipator			

Screen shared by lisir

gitee 开源软件 企业版 商城 私有云 博客 我的

基于ARM架构的SoC和开发板可以基于
· 深圳米尔科技的产品进行选择

登记表

SOC型号	soc厂商	bsp型号	赞助商	gitee id
T113-S3	全志	MYD-YT113S3-4E128D-110-I-G	米尔科技	
T113-I	全志	MYD-YT113i-4E256D-110-I	米尔科技	
T527	全志	MYD-LT527M-16E2D-180-E	米尔科技	
RZ/G2L	瑞萨	MYD-YG2L23-8E1D-120-C-REMI	米尔科技	alichinese
IMX 6ULL	NXP	MYD-Y6ULY2-V2-4E512D-50-I	米尔科技	darrenpig
IMX 8M Plus	NXP	MYD-JX8MPQ-8E2D-160-I	米尔科技	puai
STM32MP135	ST	MYD-YF135-4E512D-100-I	米尔科技	bigclouds99
AM62x	TI	MYD-YM6254-8E2D-140-I	米尔科技	emancipator

附件: 上传图片

+ 添加评论

Gitee Reward

查看其他的项目

第一步: 研读资料, 把握系统启动, 烧录等, 熟悉, 了解
第二步: 驱动移植, 固件的SoC驱动开发比较成熟

三、要求项目跟进

SDK资料\熟悉、了解、板子的内核迁移

NXP、大部分都在Linux_openEuler里有驱动

按设备树驱动移植

切换内核、驱动移植、验证、Debug

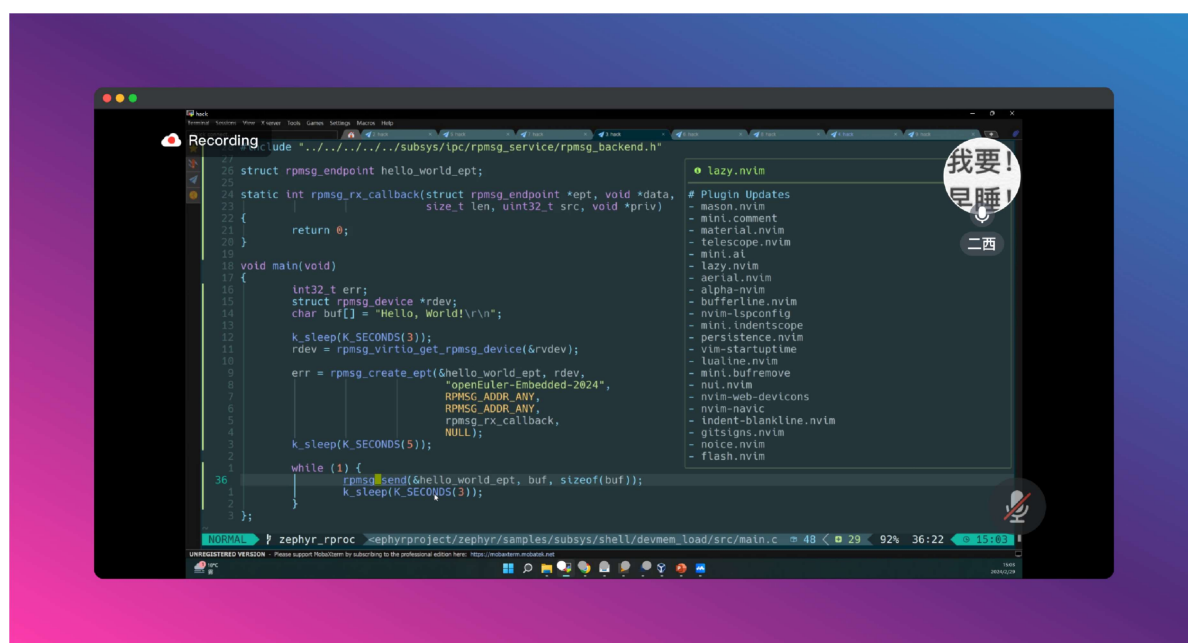
Yocto引入BSP层，按树莓派、瑞星微，引入官方的层，代码欧拉化驱动、引进 注意NXP里的软件层

官方的资料的文档.....欧拉的文档要相应的跟进

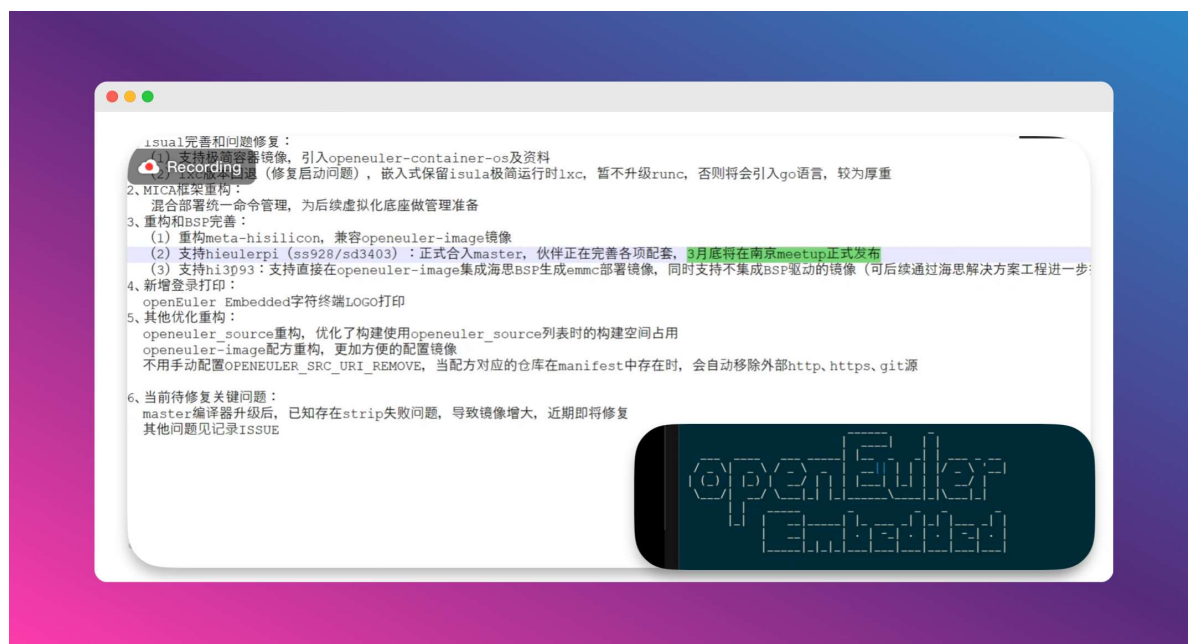
提交欧拉的板块、最小系统的拉起、官方地方特定工程

内核参考、如果内核成熟可以简略

5月15日截至



新做了一个UI



外部工具链 改为Poky localcast 文件中

TCMODE = (外部构建)

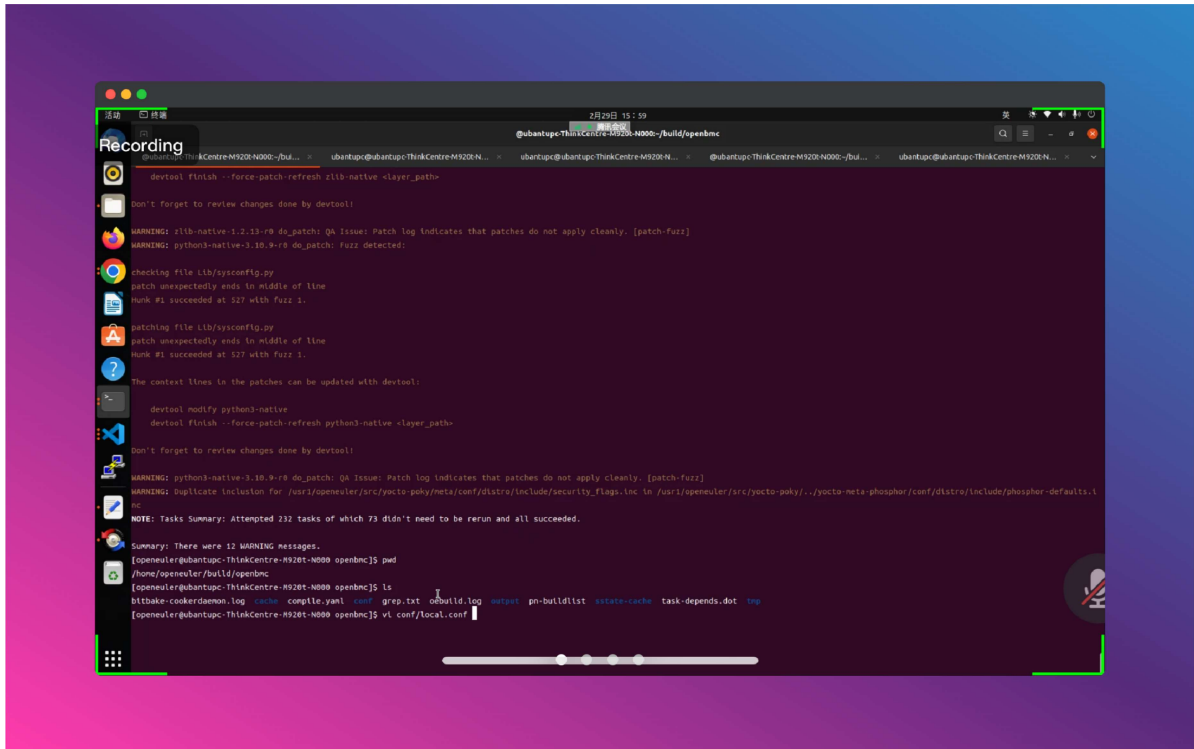
Yocto 的 `OPENEULER_PREBUILT_TOOLS_ENABLE` = " "

原生的gcc 的cross

关掉上述两个之后，就用Poky原件GCC 的方法

Qmake的进度，已经可以在构建SDK

社区的相关的问题



[fork 主仓参考 提交PR上去]，NXP我们的进度，参考树莓派文档风格，