JAILHOUSE HYPERVISOR ON I.MX

Peng Fan
peng.fan@nxp.com

About me, about Jailhouse

- Member i.MX Kernel & Virtualization team
- Focus on Kernel, Virtualization, U-Boot, ATF, SoC bringup
- i.MX OP-TEE maintainer(2015~2018)
- OSS developer, upstream contributor

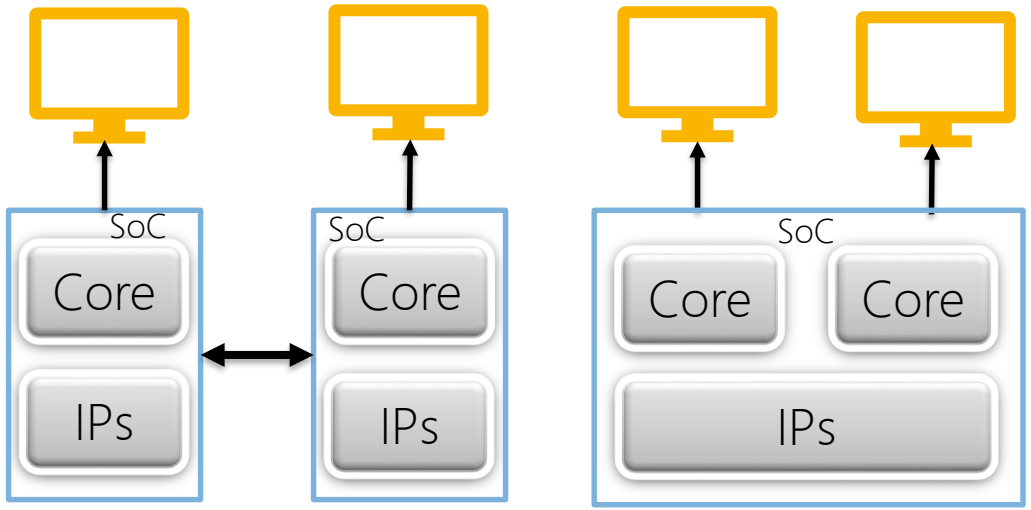
- Jailhouse
  - https://github.com/siemens/jailhouse
  - GPLv2
  - Started as open source project by Siemens
  - Small community

Agenda

- Basic Introduction
- Current status
- Inter-VM Communication, virtio on the road
- Eliminate VM-EXIT
- Jailhouse on i.MX
- Jailhouse on the wheels
- Q&A

Why Hypervisor?

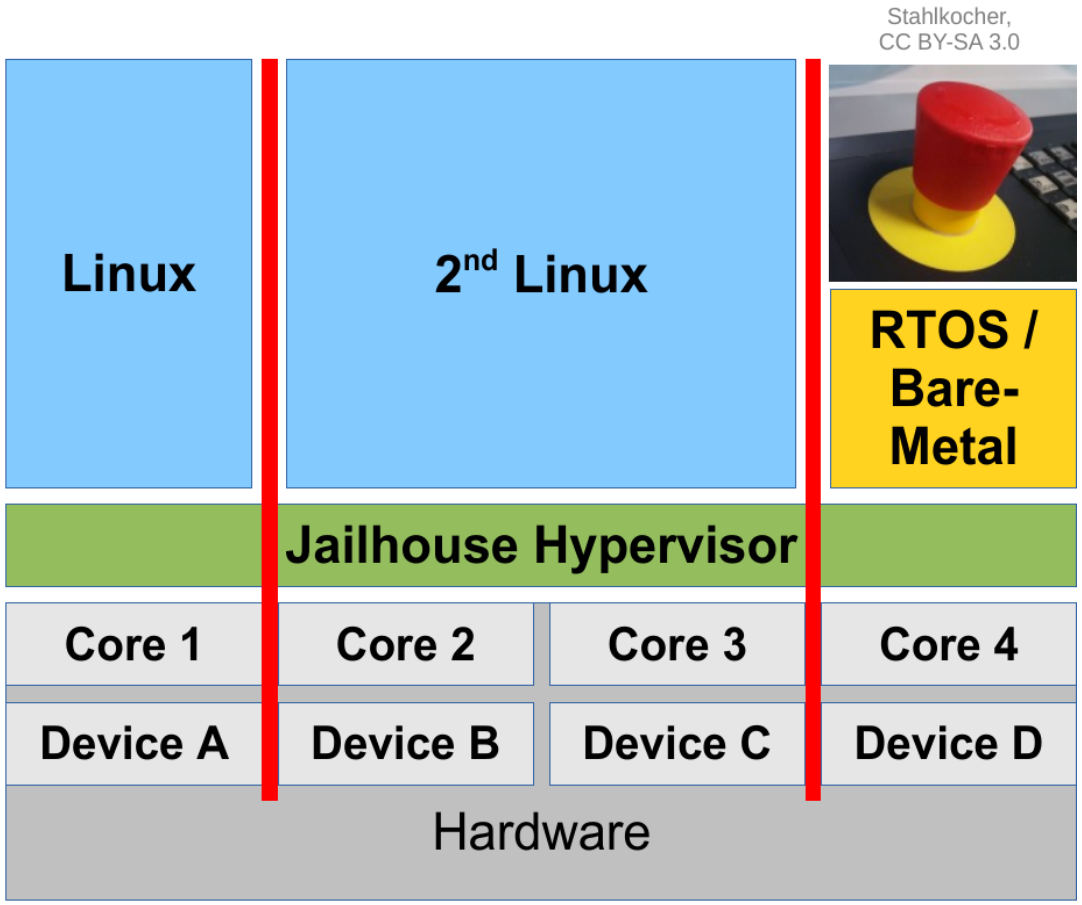
The reduced size, weight and power requirements make using virtualization/hypervisor a very attractive alternative for next generation designs



Jailhouse: Introduction

- Focus on maintaining static partitions
- No scheduling
- 1:1 resource assignment
- (Almost) no device emulation
- Keep runtime code base minimal
- Enable / simplify safety certification

- Full CPU isolation
- Minimal I/O latencies

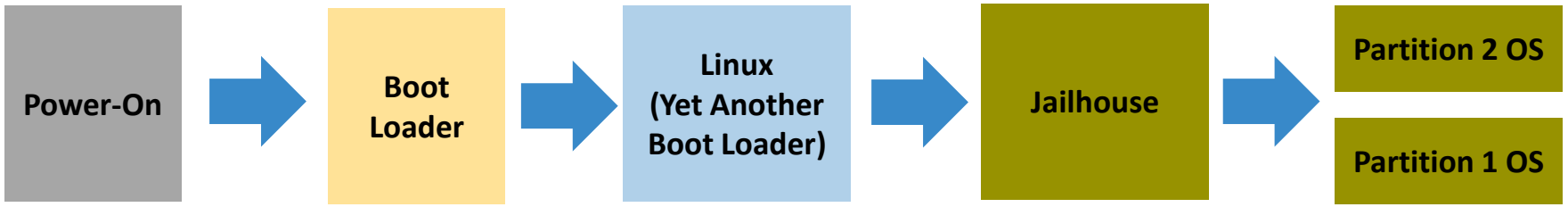


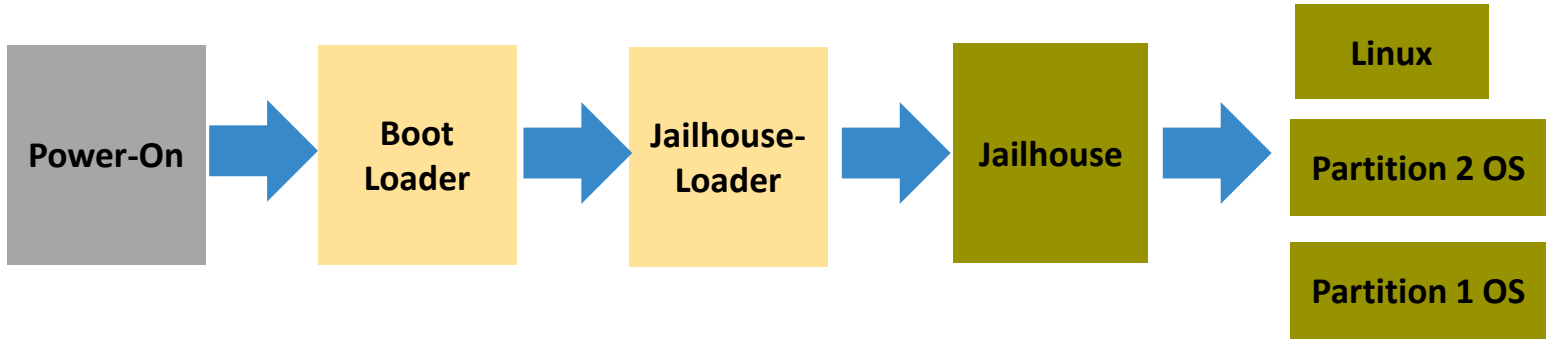
Current stauts

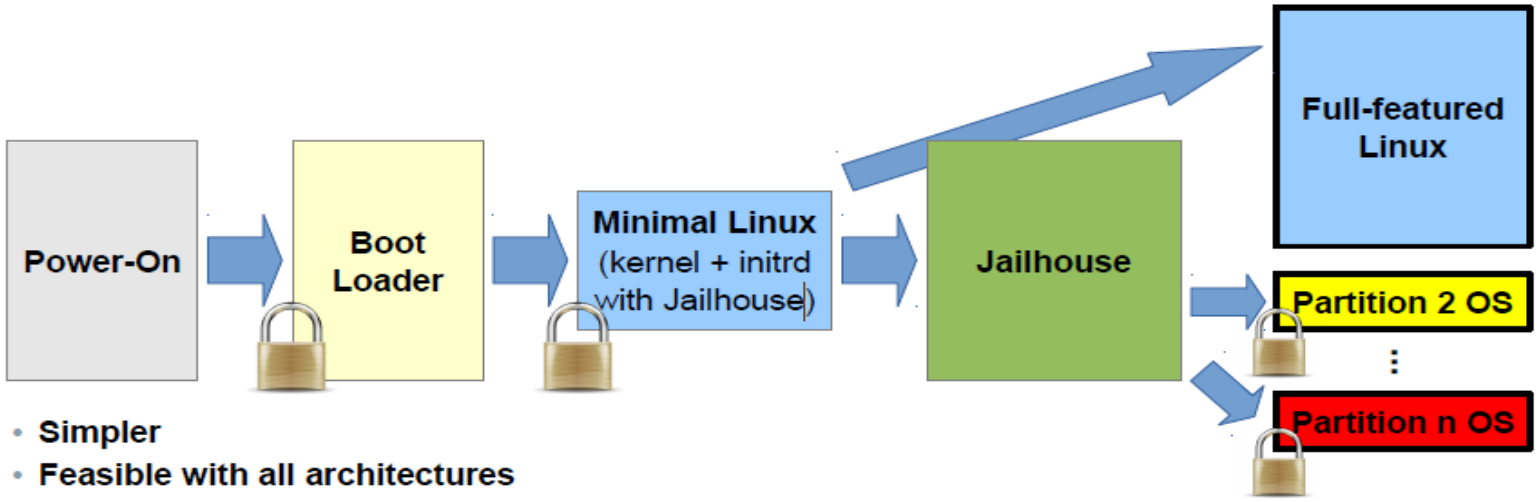
- X86: (From Jan)
  - **Supports both Intel and AMD systems**
    - • Requirement: VT-x / VT-d, AMD-V / IOMMU
    - • Works inside QEMU/KVM
    - • AMD interrupt remapping on to-do list
  - **• It's small!**
  - • Currently ~9.2K lines of code (for Intel)
  - **• Direct interrupt delivery**
  - • Zero VM exits, minimal latencies feasible
  - • Max. timer IRQ latency (Xeon D-1540): **<1 µs**
  - **• Cache Allocation Technology**
  - • Intel feature for partitioning caches
  - • L3 supported, L2 on to-do list
    - Support iommu including smmu-v2/v3

- ARM
  - Supports both v7/v8, including NXP i.MX8/8M, TI, Nvidia, Xilinx ZynqMP, QEMU
  - Support GIC-v2/v3
  - Support SMMU-v2/v3
  - Code size small

Jailhouse boot process







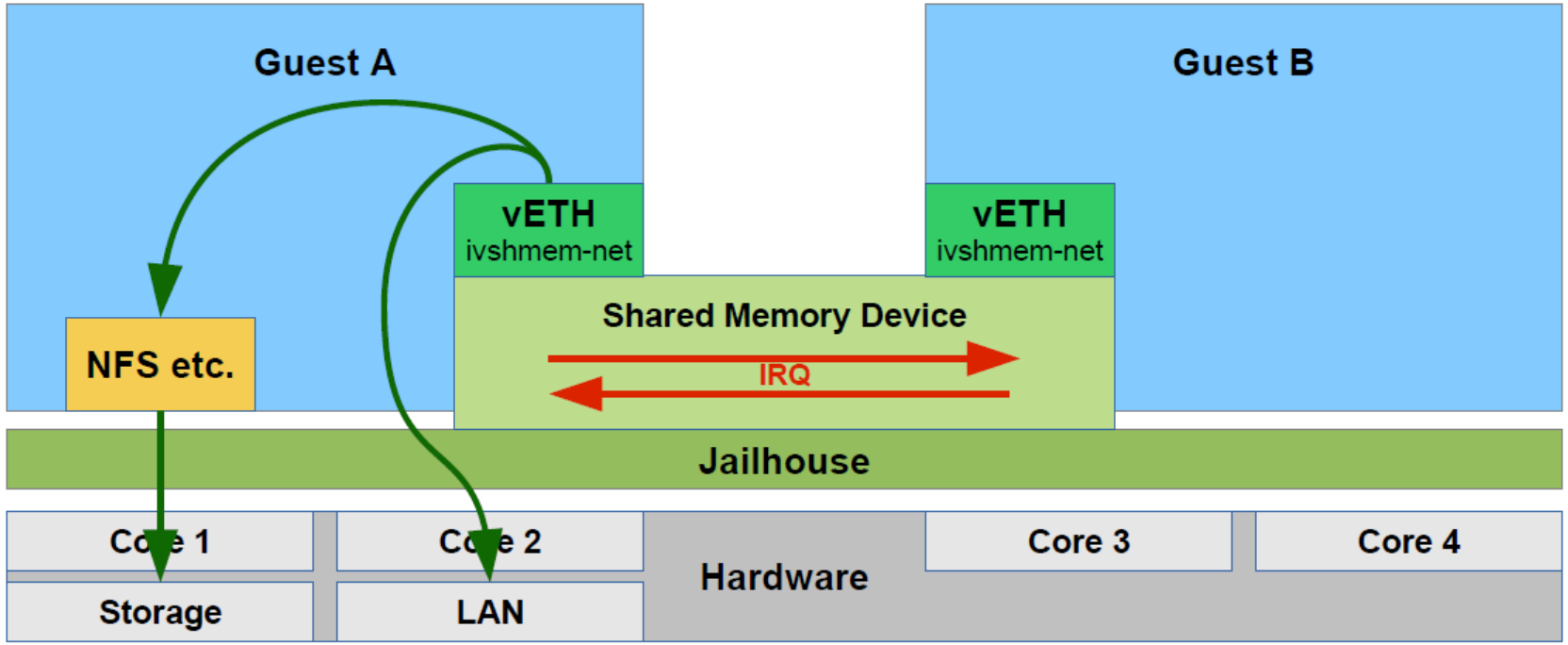
- Simpler
- Feasible with all architectures
- Prevent undesired hardware access of full-featured Linux

Jailhouse management interface

- linux # jailhouse enable system.cell
- linux # jailhouse cell create realtime.cell
- linux # jailhouse cell load my-cell rtos.bin
- linux # jailhouse cell start my-cell
- linux # jailhouse cell destroy my-cell
- linux # jailhouse cell linux linux.cell kernel -i initrd -d dtb
  - jailhouse cell linux /usr/share/jailhouse/cells/imx8mm-linux-demo.cell /run/media/mmcblk1p1/Image -d /run/media/mmcblk1p1/imx8mm-evk-inmate.dtb -c "clk_ignore_unused console=ttymxc3,115200 earlycon=ec_imx6q,0x30890000,115200 root=/dev/mmcblk2p2 rootwait rw"
- linux # jailhouse disable

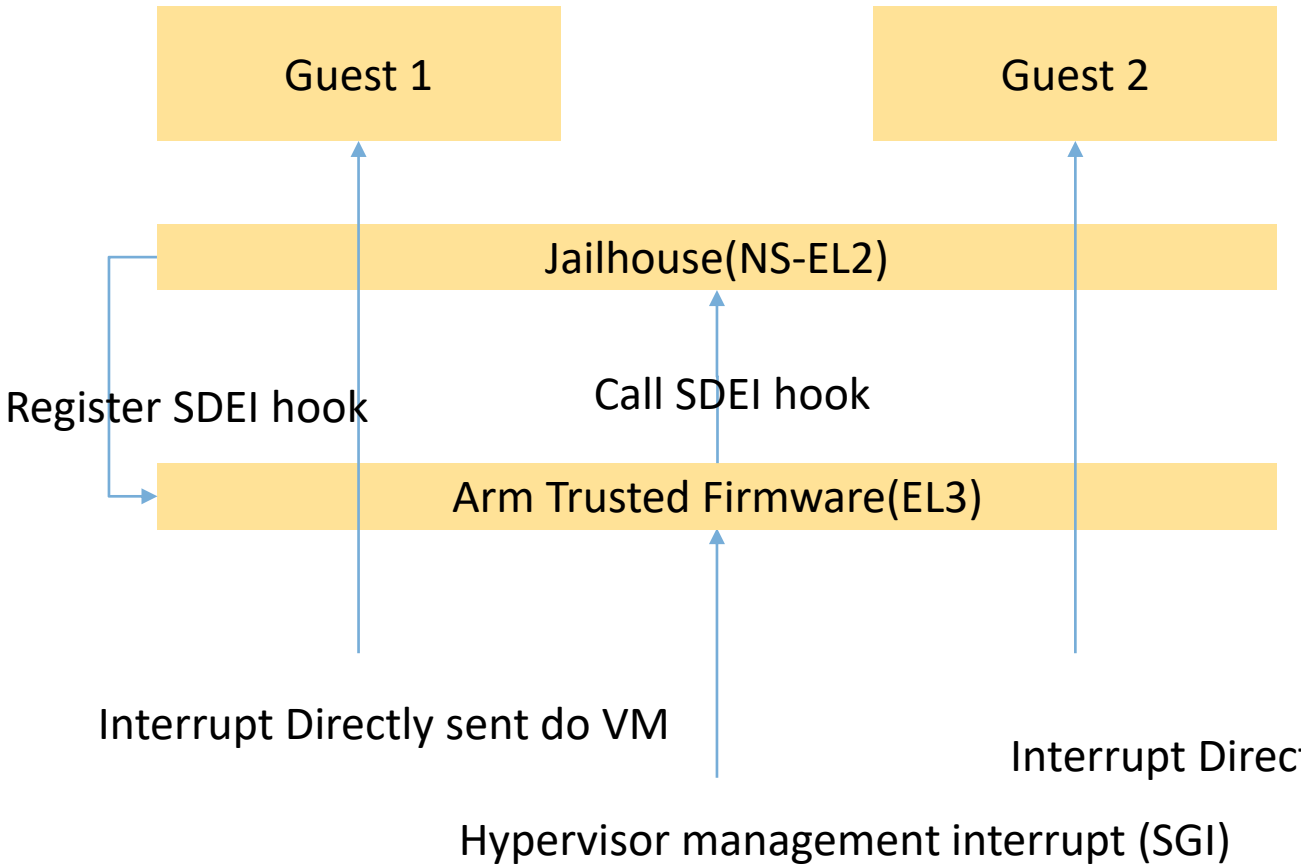
Inter-VM communication

- ivshmem



ivshmemv2 + virtio on the way: https://www.mail-archive.com/jailhouse-dev@googlegroups.com/msg08548.html

Eliminate vm-exit



1. JH core0 issue SDEI event to core1
2. SGI interrupt triggered to core1 EL3
3. EL3 ATF call SDEI hook of JH to modify EL2 configuration to force IABT core1 VM
4. EL3 eret to core1 VM
5. Core1 VM will IABT immediately and trap to JH NS-EL2
6. JH NS-EL2 restore EL2 configuration of core1 VM and do other work

7. JH: https://github.com/siemens/jailhouse/commits/wip/arm64-zero-exits
8. SDEI: https://git.trustedfirmware.org/TF-A/trusted-firmware-a.git/commit/?id=8567103ef94b1abb52f9fb053bd6118913878d74

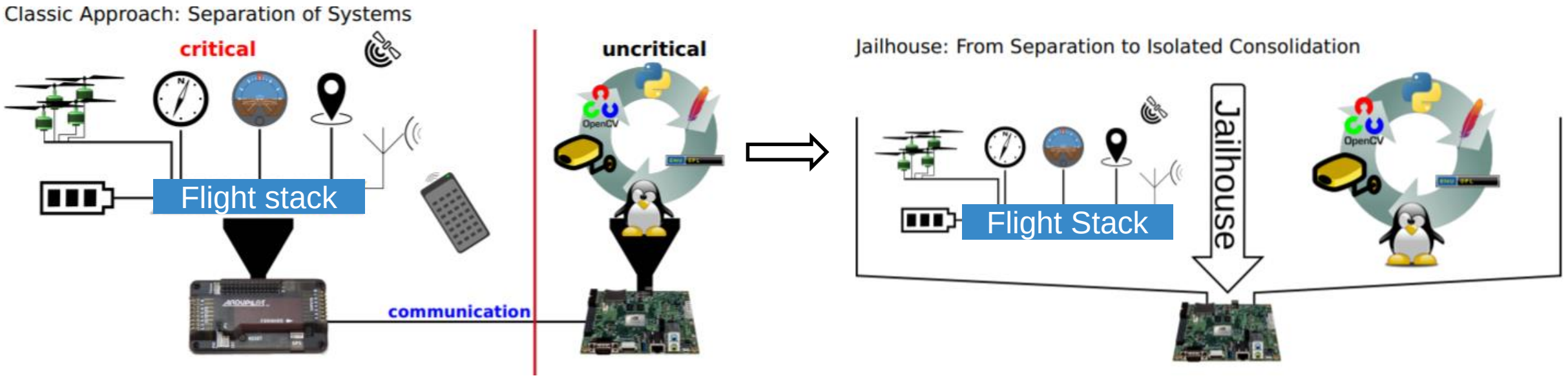
Jailhouse on i.MX

- Support Linux + Baremetal
- Support Linux + Linux
- Support all i.MX8/8M SoCs
- Source:
  - https://source.codeaurora.org/external/imx/imx-jailhouse/
  - Ask: https://community.nxp.com/

- Mass Production, Immersive 3D audio solution
- https://www.nxp.com/design/i-mx-developer-resources/immersiv3d-audio-framework-software:IMMERSIV3DAF

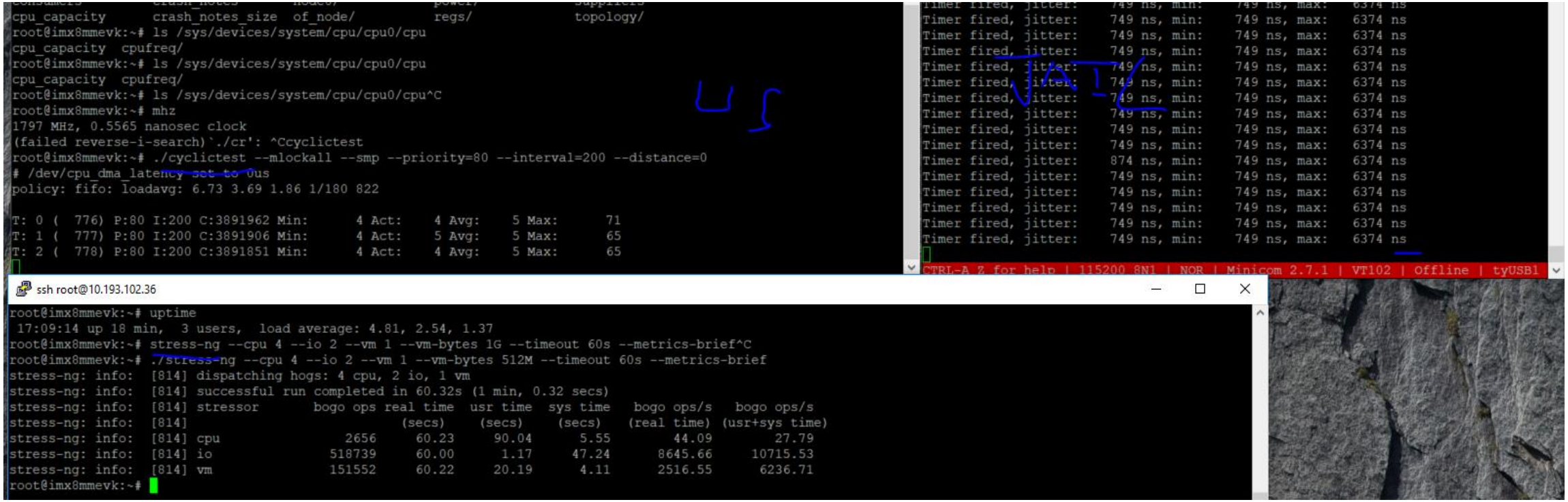
Jailhouse for Drones/Robot?

- Consolidate workloads
- Use high performance A cores to run more complicated computing.



Realtime test on i.MX

- 5.4.47 RT kernel(core0/1/2) + jailhouse(gic-demo jitter test on cpu3, no cache color) + stress-ng 60s.



Jailhouse on the wheels

- But still new stuff need to enable:
  - New hardware
  - Cache coloring
  - Boot Jailhouse before Linux
  - MPAM
  - Zephyr on Jailhouse
  - Virtio
  - GDB
  - Livepatch?
  - ….
- Welcome Contribution to Jailhouse Hypervisor

Q&A

SECURE CONNECTIONS
FOR A SMARTER WORLD