北京郵電大學
Beijing University of Posts and Telecommunications

RROS

Bootloader初探

-以uboot为例

李弘宇
北京邮电大学

Background: How does a computer boot

- Boot process
  - X86:
    - Bios->grub->mbr/gpt->kernel
  - Arm:
    - Bootrom(optional)->uboot->kernel/image/dtb
  - Risc-v:
    - Bootrom(optional)->bootloader->rustsbi/opensbi->uboot->kernel/image/dtb
  - Others
    - Bootrom -> EEPROM -> sdcard/start.elf -> sdcard/kernel.img
    - Bootrom -> sdcard/bootcode.bin -> sdcard/start.elf -> sdcard/kernel.img

Background: How does a computer boot

- Boot process
  - X86:
  - Arm:
  - RISC-V
- Concept
  - Bootloader: grub, uboot
  - Uboot = Bios+grub
- Some tips
  - Don’t be a “Language lawyer”.

Why bootloader exists

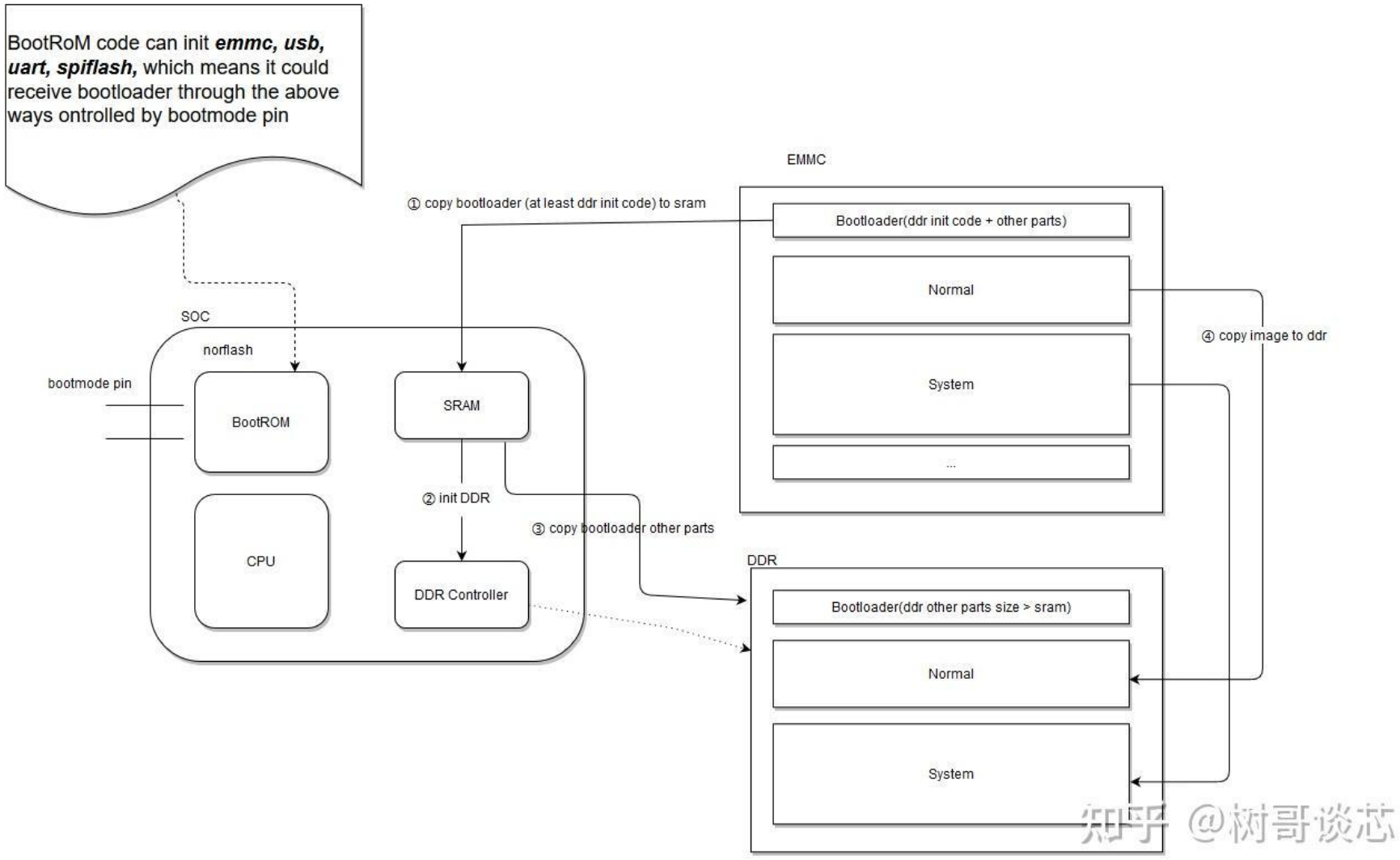
- This needs to start with how to run a program in a CPU at the very beginning
  - Long long ago, when the digital world “Computer” has nothing exist, its first citizen “CPU” needs to initialize everything, including memory.
  - But unfortunately, CPU needs to know the memory layout for running the instructions.
  - Here comes a confliction: **CPU needs the information of memory layout to initialize memory to a specific memory layout**.
  - A nature idea: stage by rocket launch
    - The loaded segment of code gradually grows up

Background: memory device

- RAM
  - SRAM: Cache
  - DRAM, SDRAM, DDR SDRAM(DDR5): Memory
- ROM
  - ROM, PROM, EPROM, EEPROM: Bios/Bootrom
  - Flash EEPROM memory:
    - **Nor flash** (Burn code): XIP, address
    - Nand flash (USB flash disk, SD/TF): block, IO

Why bootloader exists

- This nee ginning
  - Long ist, its
    first c ry.
  - But u ning the
    instru
  - Here **y layout**
    to ini
  - A nat
    - Th



Why bootloader exists

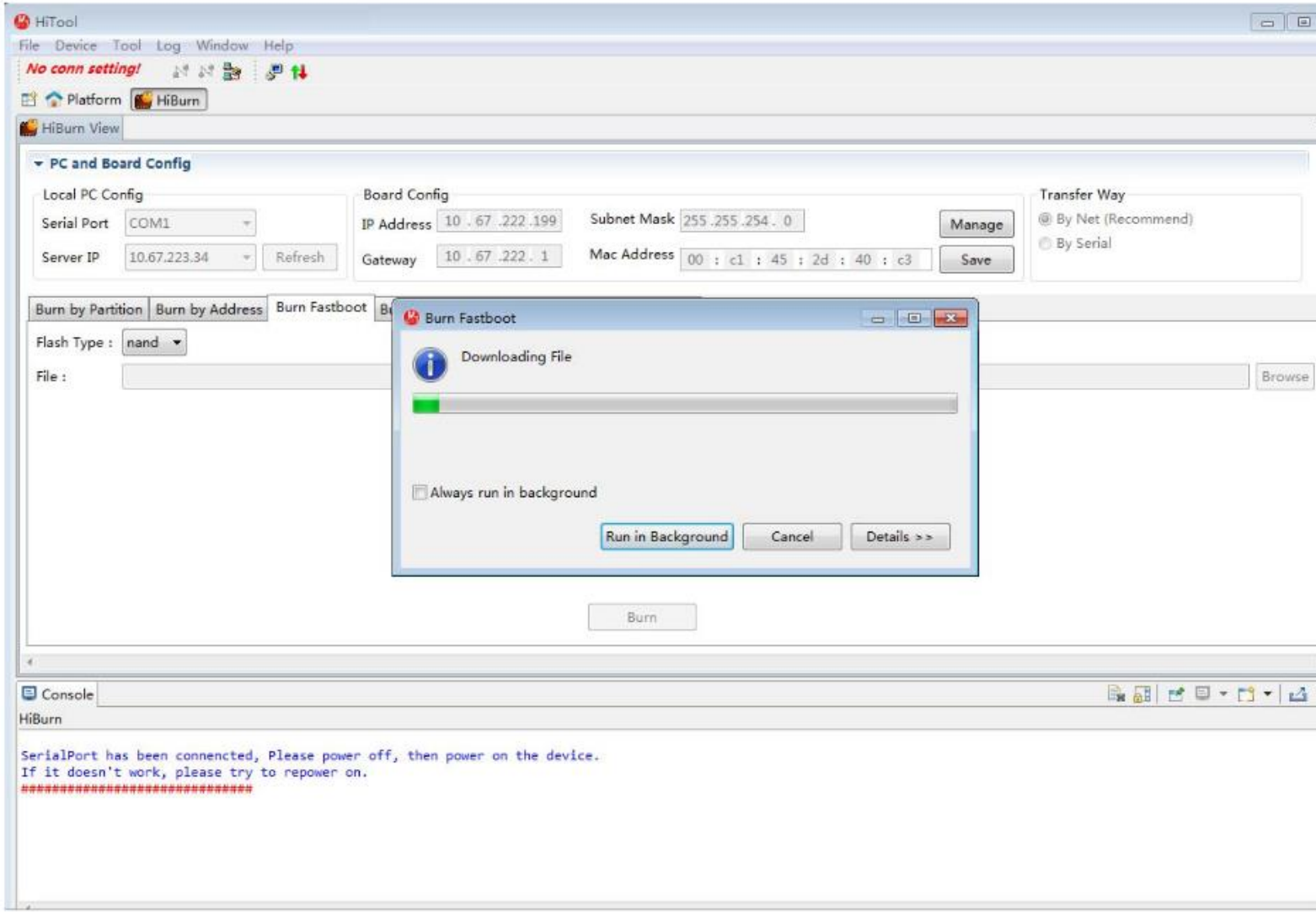
- This needs to start with how to run a program in a CPU at the very beginning
  - Long long ago, when the digital world “Computer” has nothing exist, its first citizen “CPU” needs to initialize everything, including memory.
  - But unfortunately, CPU needs to know the memory layout for running the instructions.
  - Here comes a confliction: **CPU needs the information of memory layout to initialize memory to a specific memory layout**.
  - A nature idea: stage by rocket launch
    - The loaded segment of code gradually grows up

An overview about uboot

- uboot
  - Basic instructions
  - Boot from network
- ~~Uboot source code~~
  - ~~bootrom/uboot~~
- Typical uboot use cases
  - Disk-Disk copy by uboot
  - Uboot OTA (uboot hot update)
  - Multiple OS in single Uboot
  - Multiple uboot in single machine

Basic instructions

- Uboot
  - How to burn uboot



Basic instructions

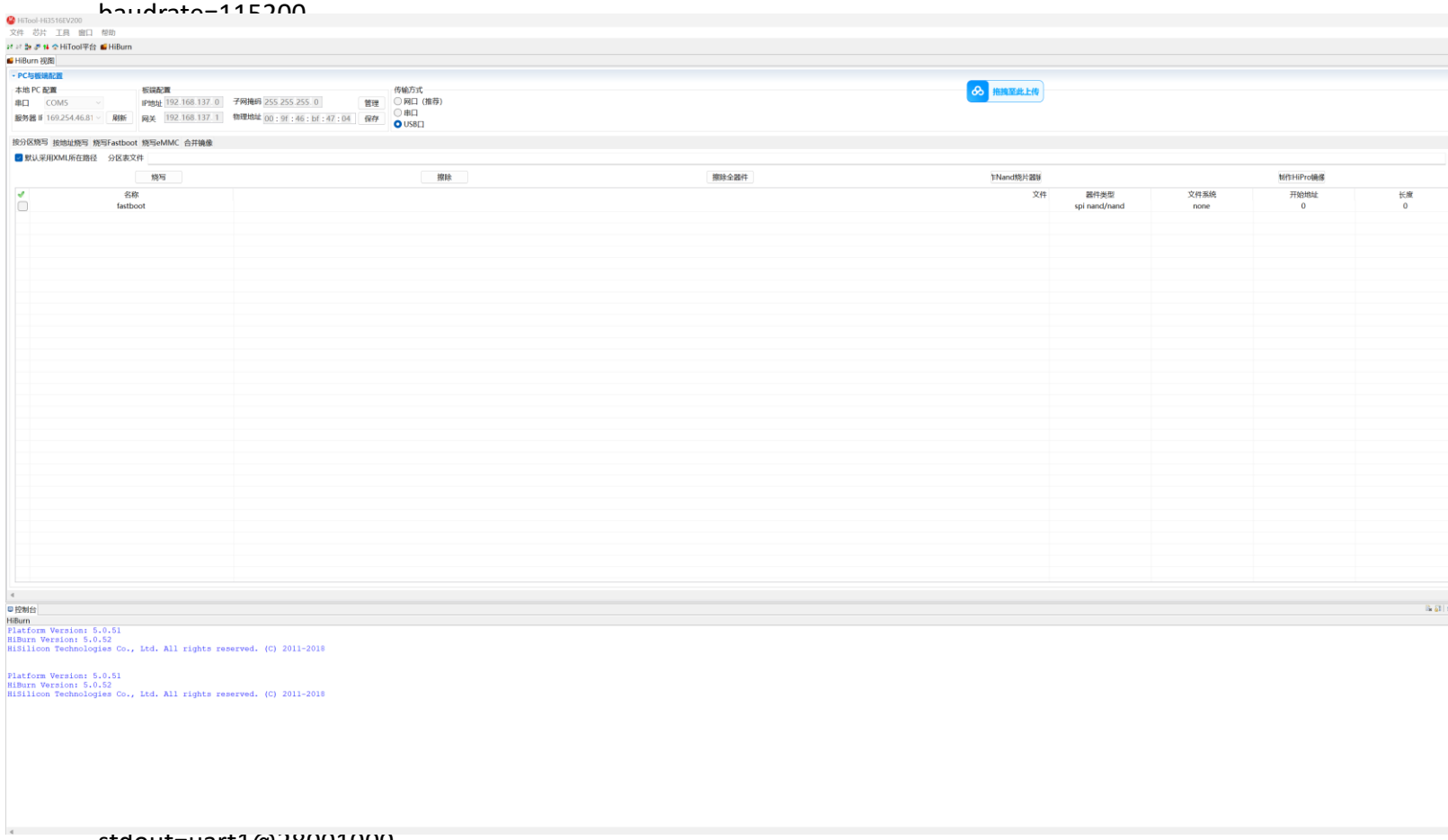
- Uboot
  - How to burn uboot
  - How to config kernel/image/dtb in uboot
    - cmd
    - Gui

```
arch=arm
baudrate=115200
board=D2000
board_name=D2000
boot_ft=run bootcmd_ft;run load_fdt_ft;run load_kernel_ft;bootm 0x90100000 -:- 0x90000000
bootargs=console=ttyAMA1,115200 earlycon=pl011,0x28001000 root=/dev/nvme0n1p1 rootwait rw init=/init
bootcmd=nvme scan; ext4load nvme 0:1 0x90200000 /boot/Image; ext4load nvme 0:1 0x90100000 /boot/d2000-devboard-dsk-fixed.dtb; booti 0x90200000 - 0x90100000
bootcmd_ft=setenv bootargs 'console=ttyAMA0,115200 earlycon=pl011,0x28001000 root=/dev/sda5 rw rootwait'
bootdelay=1
cpu=armv8
distro_bootcmd=run boot_ft
eth1addr=3c:6a:2c:3c:6a:2d
ethact=ethernet0@2820c000
ethaddr=3c:6a:2c:3c:6a:2c
fdtcontroladdr=fae46878
gateway=192.168.1.1
gatewayip=192.168.1.1
ipaddr=192.168.1.100
load_fdt_ft=ext4load scsi 0:1 0x90000000 d2000_ok.dtb
load_kernel_ft=ext4load scsi 0:1 0x90100000 uImage-d2000
netmask=255.255.255.0
serverip=192.168.1.101
stderr=uart1@28001000
stdin=uart1@28001000
stdout=uart1@28001000
vendor=phytium

Environment size: 958/4092 bytes
```

Basic instructions

- Uboot
  - How to burn uboot
  - How to config kernel/image/dtb in uboot
    - cmd
    - Gui



arch=arm
baudrate=115200
stdout=uart1@28001000
vendor=phytium

Environment size: 958/4092 bytes

Boot from network

- Boot from network
  - Why boot from network?
    - Update bootloader without operating the machine manually
  - How to boot from network
    - Uboot has the tftp function to upload file based on the serial

```
ethact=ethernet0@2820c000
ethaddr=3c:6a:2c:3c:6a:2c

gateway=192.168.1.1
gatewayip=192.168.1.1
ipaddr=192.168.1.100
netmask=255.255.255.0
serverip=192.168.1.101
```

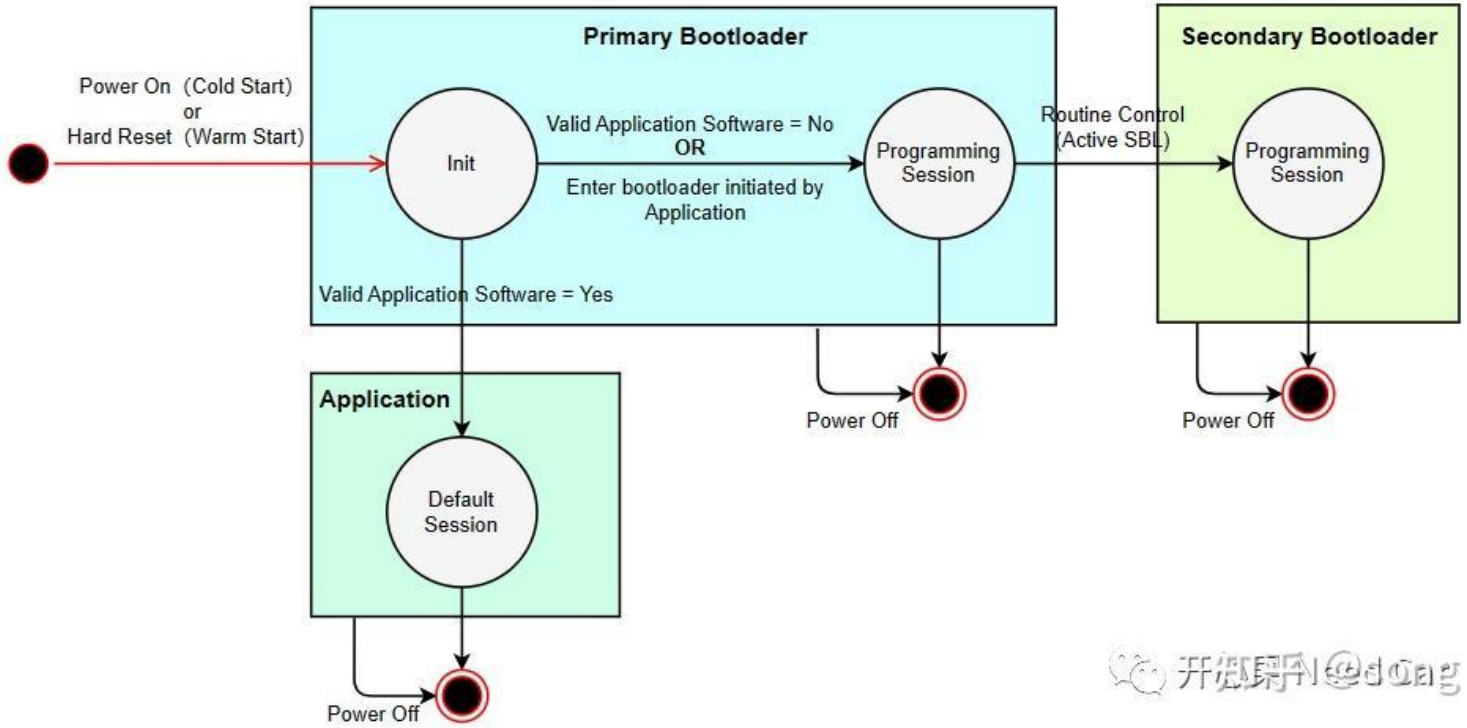
```
tftpboot 0x90200000 Image
tftpboot 0x90100000 d2000-devboard-dsk-fixed.dtb
tftpboot 0x93000000 uInitrd
booti 0x90200000 0x93000000 0x90100000
```

Disk-Disk copy by uboot

- Save image
  - dd if=/dev/sda_copy_source | ssh user@192.168.1.2 (client ip) "dd of=/backup.img “
- Restore image
  - dd if=/backup.img | ssh user@192.168.1.3 (server ip) "dd of=/dev/sda_copy_destination"

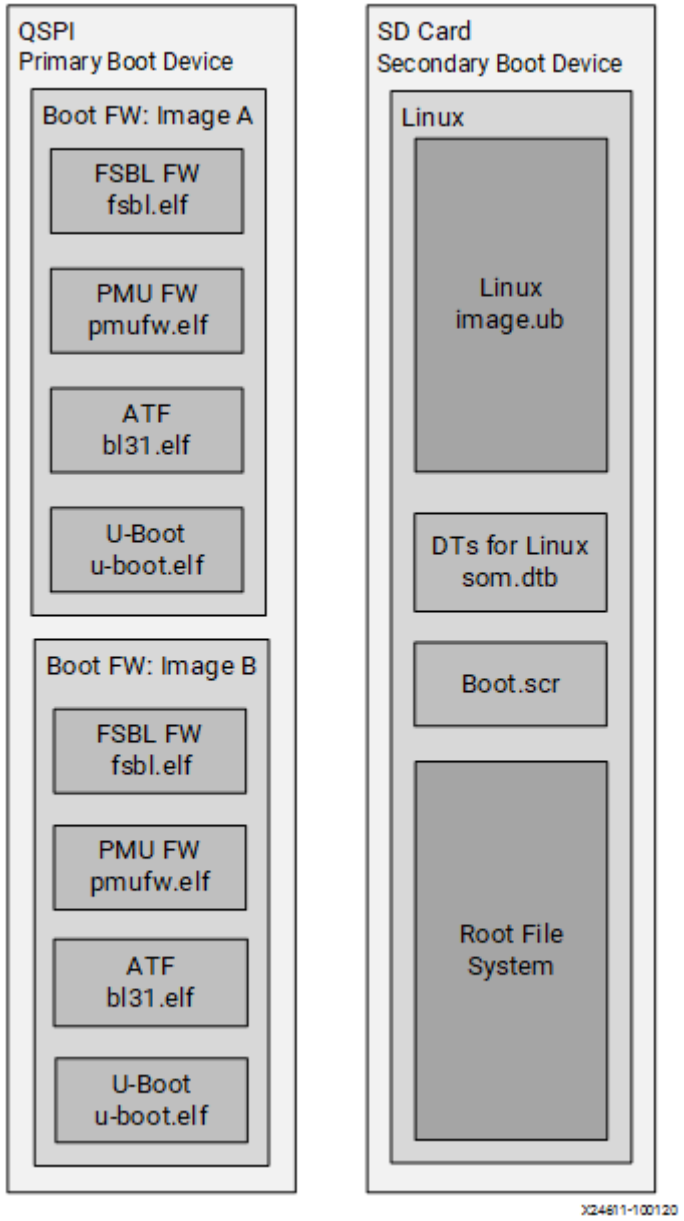
Uboot OTA

- Background
  - Software update may break



Multiple OSes/Uboot

- Background
  - Software update may break
- How to use OTA



Reference

- Uboot/bootloader
  - Riscv: https://zhuanlan.zhihu.com/p/482381637
  - BIOS/UEFI: https://www.zhihu.com/people/mikewolfwoo
  - UBOOT network: https://zhuanlan.zhihu.com/p/115377569
  - Uboot OTA: https://bootlin.com/pub/conferences/2022/elce/opdenacker-implementing-A-B-system-updates-with-u-boot/opdenacker-implementing-A-B-system-updates-with-u-boot.pdf

动态 回答 510 视频 17 提问 1 **文章** 328 专栏 5 想法 247 收藏 16 关注

Backup

Multiple OSes

- uboot
  - Basic instructions
  - Boot from network
- Uboot source code
  - bootrom/uboot
- Typical uboot use cases
  - Disk-Disk copy by uboot
  - Uboot OTA (uboot hot update)
  - Multiple OS in single Uboot
  - Multiple uboot in single machine

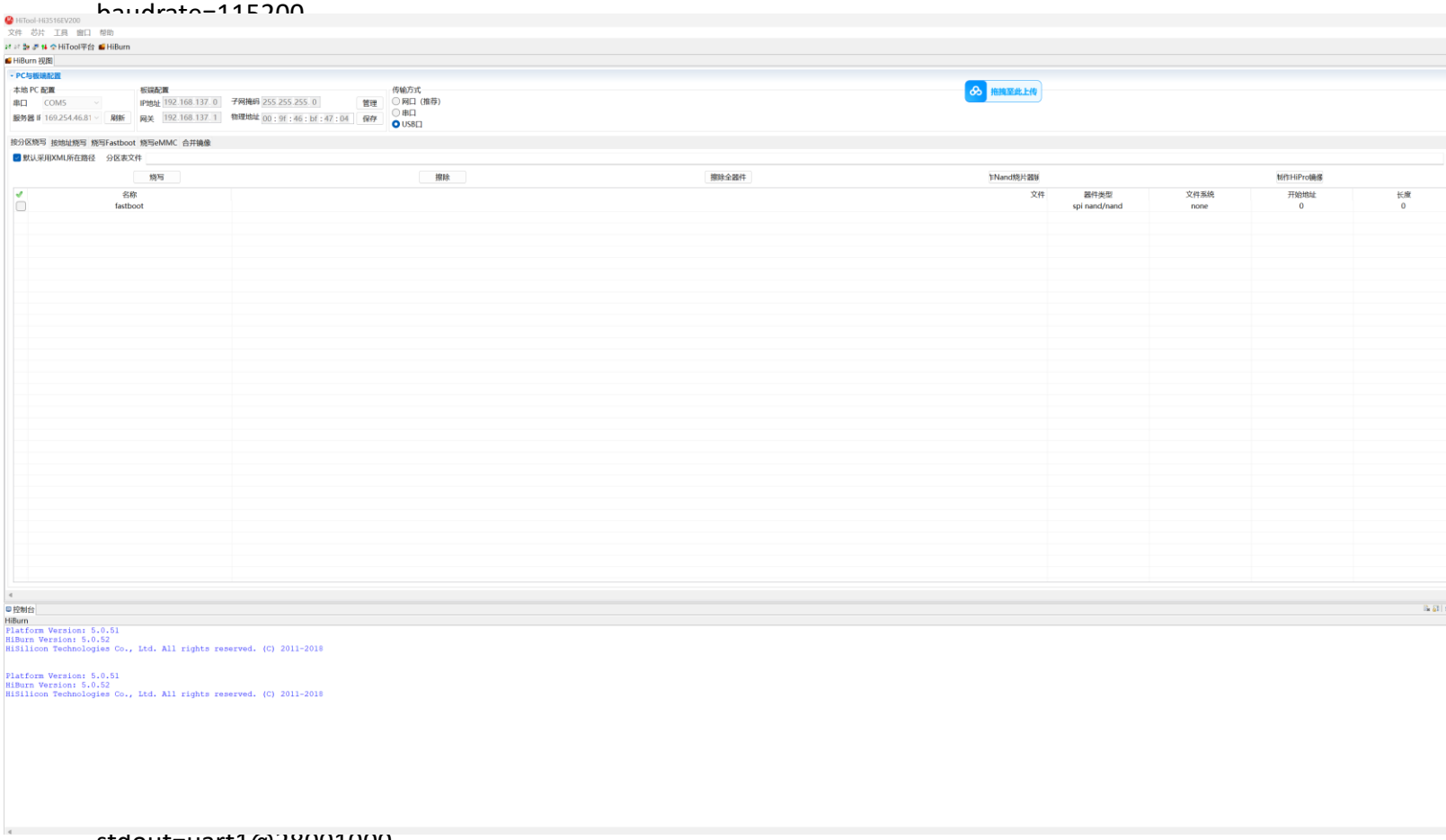
Multiple Uboot

- uboot
  - Basic instructions
  - Boot from network
- Uboot source code
  - bootrom/uboot
- Typical uboot use cases
  - Disk-Disk copy by uboot
  - Uboot OTA (uboot hot update)
  - Multiple OS in single Uboot
  - Multiple uboot in single machine

Basic instructions

- Uboot
  - How to burn uboot
  - How to config kernel/image/dtb in uboot
    - cmd
    - Gui

arch=arm
baudrate=115200



stdout=uart1@28001000
vendor=phytium

Environment size: 958/4092 bytes

Uboot功能

- uboot基本使用方法
  - uboot boot from network
- uboot源码初探
  - bootrom/uboot启动顺序
- uboot经典案例
  - 基于 uboot的硬盘对拷
  - 对 uboot OTA 升级（A/B机 uboot热更新）
  - 单 uboot多OS切换
  - 多 uboot备份

参考链接

- uboot基本使用方法
  - uboot boot from network
- uboot源码初探
  - bootrom/uboot启动顺序
- uboot经典案例
  - 基于 uboot的硬盘对拷
  - 对 uboot OTA 升级（A/B机 uboot热更新）
  - 单 uboot多OS切换
  - 多 uboot备份