SH-Mobile ARM zboot
Japan Technical Jamboree 37
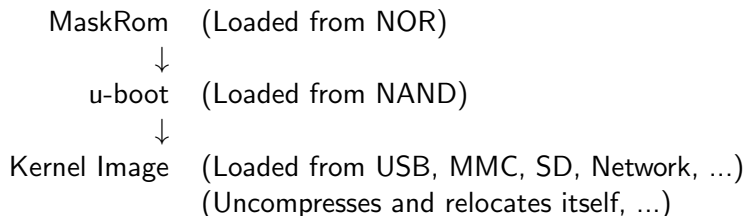
Simon Horman <simon@horms.net>
Horms Solutions Ltd., Tokyo

20th May 2011

SH-Mobile ARM zboot

- SH-Mobile ARM is an ARM platform developed by Renesas
- Target for this work is the AP4EB and Mackerel boards
- Default boot loader is u-boot
- Would like to boot directly to linux;
- Or use Linux as a boot loader (kexec)

Motivation

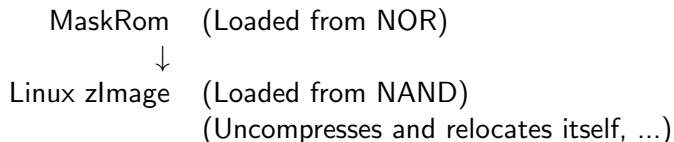- Prove Hardware
- Provide options for customers
- Upstream first policy

Boot Sequence with u-boot on NAND

MaskRom (Loaded from NOR)
↓
u-boot (Loaded from NAND)
↓
Kernel Image (Loaded from USB, MMC, SD, Network, ...)
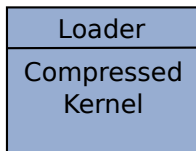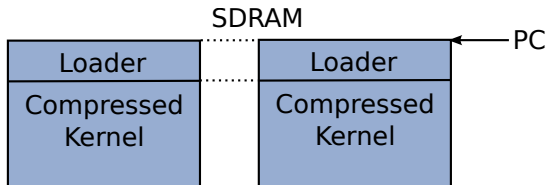(Uncompresses and relocates itself, ...)

Goal 1: Boot zImage from NAND

- NAND zboot support has been merged into 2.6.38-rc1
- Success...
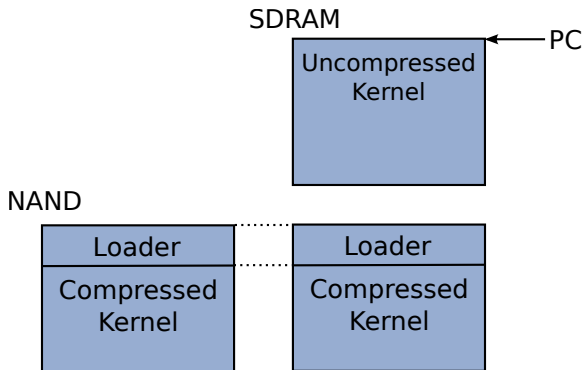 ...but end users generally can't flash the NAND.

MaskRom   (Loaded from NOR)
↓
Linux zImage   (Loaded from NAND)
               (Uncompresses and relocates itself, ...)

Loader

Compressed
Kernel

- MaskRom copies the zImage to SDRAM;
- And and jumps to it

SDRAM

Uncompressed
Kernel

PC

NAND

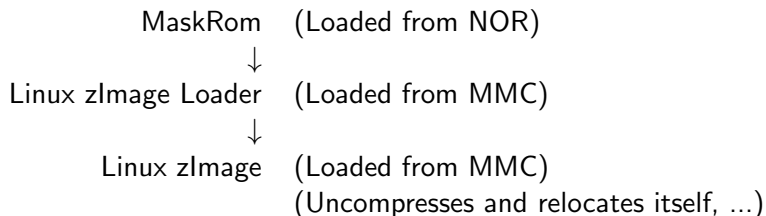| Loader	Loader
Compressed Kernel	Compressed Kernel

- zImage Loader uncompresses the Kernel;
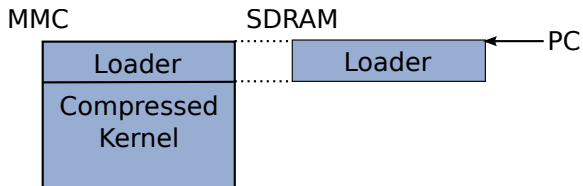- And jumps to it

Goal 2: Boot zImage from MMCIF

- MMCIF is a hardware block that accesses MMC cards
- MMCIF zboot support merged into 2.6.39-rc1
- A hardware-hack is required,
  unfortunately this is not generally available.
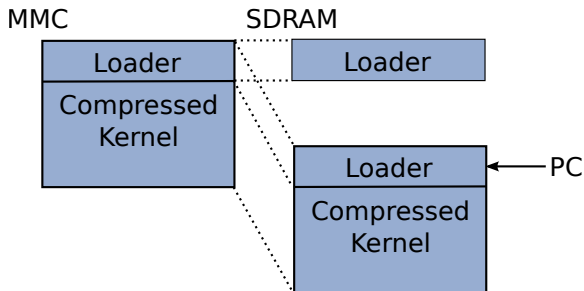
Boot Sequence with zboot on MMCIF

```
          MaskRom  (Loaded from NOR)
             ↓
Linux zImage Loader  (Loaded from MMC)
             ↓
    Linux zImage  (Loaded from MMC)
                  (Uncompresses and relocates itself, ...)
```
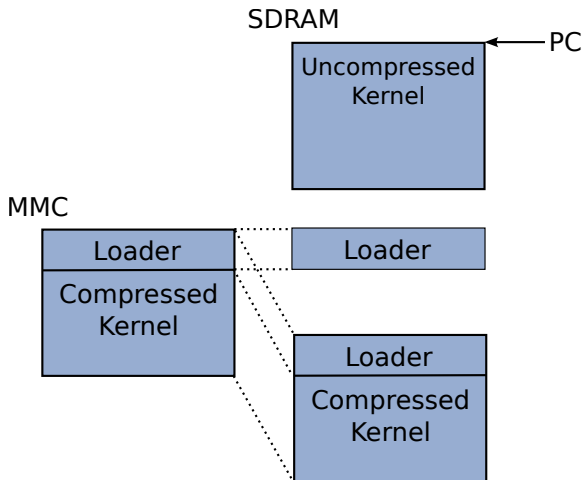
- MaskRom copies the zImage Loader to SDRAM;
- And jumps to it

- zImage Loader copies the entire zImage to SDRAM;
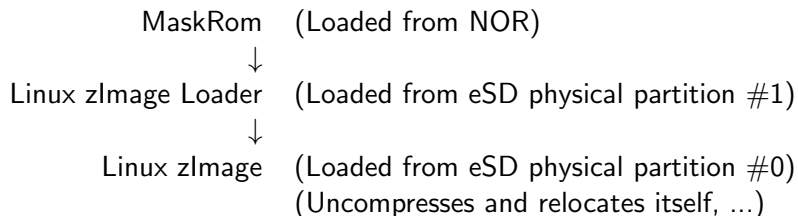- And jumps to the next instruction in the loader in the new location

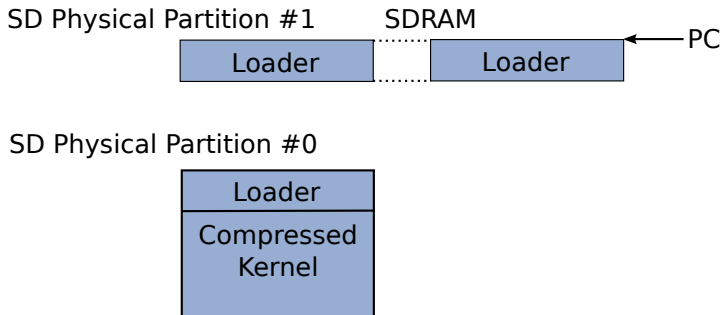- zImage Loader uncompresses the Kernel;
- And jumps to it

Goal 3: Boot zImage from SDHI

- SDHI is a hardware block that accessed SD and MMC cards
- Complete — Aiming merge in 2.6.40-rc1
- An eSD card is required,
  unfortunately this is not generally available.

Boot Sequence with zboot on SDHI

MaskRom (Loaded from NOR)
         ↓
Linux zImage Loader (Loaded from eSD physical partition #1)
         ↓
Linux zImage (Loaded from eSD physical partition #0)
(Uncompresses and relocates itself, ...)

zboot from SDHI: Loaded (1)

SD Physical Partition #1   SDRAM

| Loader | | Loader | ← PC |

SD Physical Partition #0

| Loader |
| Compressed Kernel |

- MaskRom copies the zImage Loader to SDRAM;
- And jumps to it

SD Physical Partition #1   SDRAM

| Loader | | Loader |

SD Physical Partition #0

| Loader | | Loader | ← PC |
| Compressed Kernel | | Compressed Kernel |
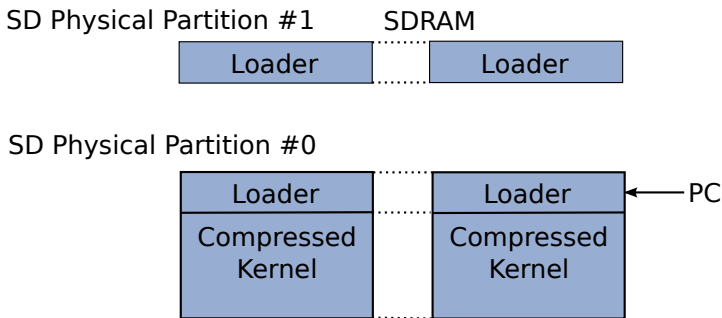
- zImage Loader copies the entire zImage to SDRAM;
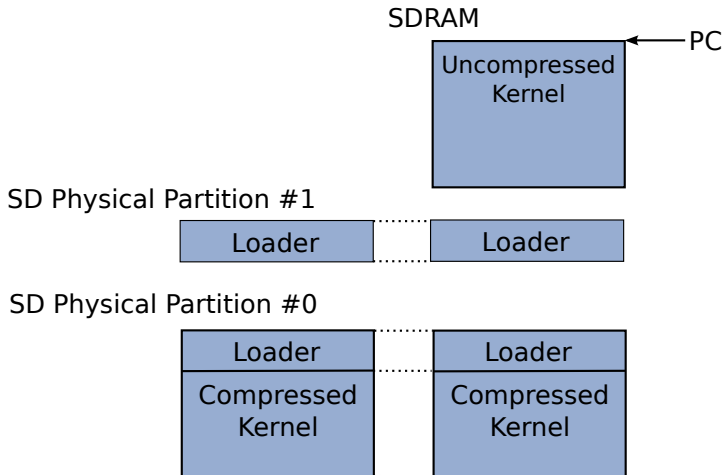- And jumps to the next instruction in the loader in the new location

zboot from SDHI: Loaded (2) (cont)
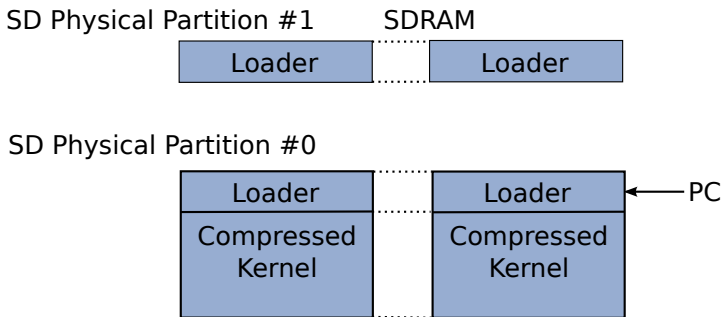
- Physical partition #0 is used as the maximum size of physical partition #1 is 128kb.
- Partition #0 requires a partial copy of the zImage that is in partion #1
- A better boot protocol, possibly a self-contained loader in partition #0 would be nice – a progression of the appraoch taken for MMCIF

zboot from SDHI: Uncompressed

SDRAM

Uncompressed Kernel

PC

SD Physical Partition #1

Loader .... Loader

SD Physical Partition #0

Loader .... Loader

Compressed Kernel .... Compressed Kernel

- zImage Loader uncompresses the Kernel;
- And jumps to it

Further work

- zboot on APE5REV

Questions