Split Boot v2

Simple and Reliable Network-Based Booting for Serenity-S1 and other Boards with ZynqMPs Devices

Marvin Fuchs (KIT)
on behalf of the Serenity Consortium

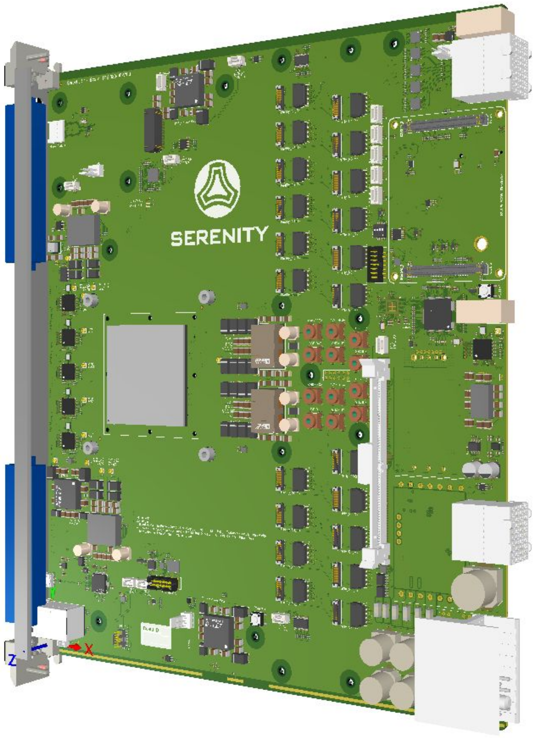
marvin.fuchs@kit.edu

Serenity-S1 ATCA Card

SERENITY

Serenity-S1 is the successor of Serenity-Z1.2 and Serenity-A1

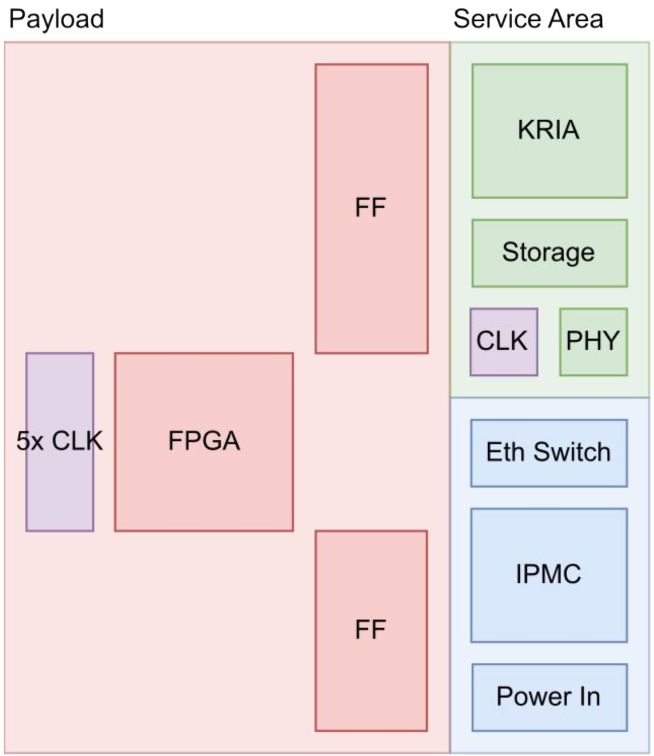
- Will be used in HL-LHC upgrade
- More than 700 cards will be used in various CMS systems
- Derived from successfully tested and evaluated Serenity-Z and Serenity-A boards
- Developed in git-controlled Altium project in a collaboration of multiple institutes led by the Imperial College London and KIT



https://indico.desy.de/event/28053/contributions/97454/attachments/63827/78124/MT_DTS_2021_CMS_TT.pdf

[See T. Mehner et al.,TWEPP 2023]
https://indico.cern.ch/event/1255624/contributions/5443848/

Serenity-S1 ATCA Card

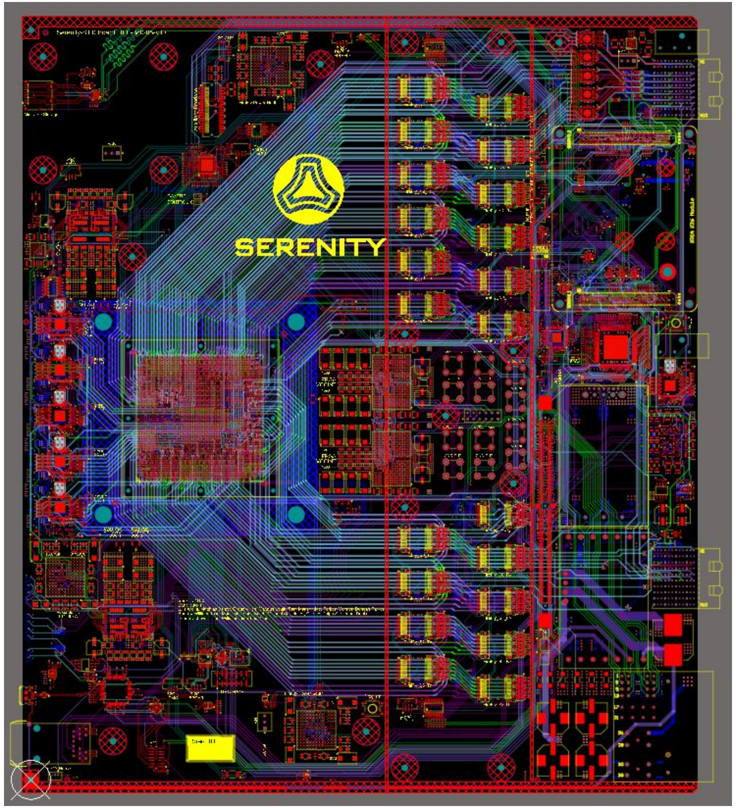


- Board Infrastructure
  - Xilinx KRIA K26 SoM
  - Clock, Power, Ethernet PHY
  - SD card, SSD
- ATCA Infrastructure
  - Backplane Connectors
  - IPMC (OpenIPMC DIMM Module)
  - Power Input
  - Ethernet Switch
- Payload
  - Samtec FireFly Optical Transceivers
  - AMD Xilinx VU13P FPGA
  - Clocks

[See T. Mehner et al.,TWEPP 2023]
https://indico.cern.ch/event/1255624/contributions/5443848/

Pilot Production

- We are still waiting for the first 12 PCBs
  - 2 PCBs will be assembled at KIT with FPGA socket
- Initial tests will be run at KIT
  - Power supply test
  - Slow control test
  - Copper SerDes loopback test
- Extended tests will be run at CERN and Imperial College London
  - Temperature cycle test
  - Optical tests
  - SerDes lane performance characterization



[See T. Mehner et al.,TWEPP 2023]
https://indico.cern.ch/event/1255624/contributions/5443848/

Serenity in CMS HL-LHC Level-1 Trigger

SERENITY

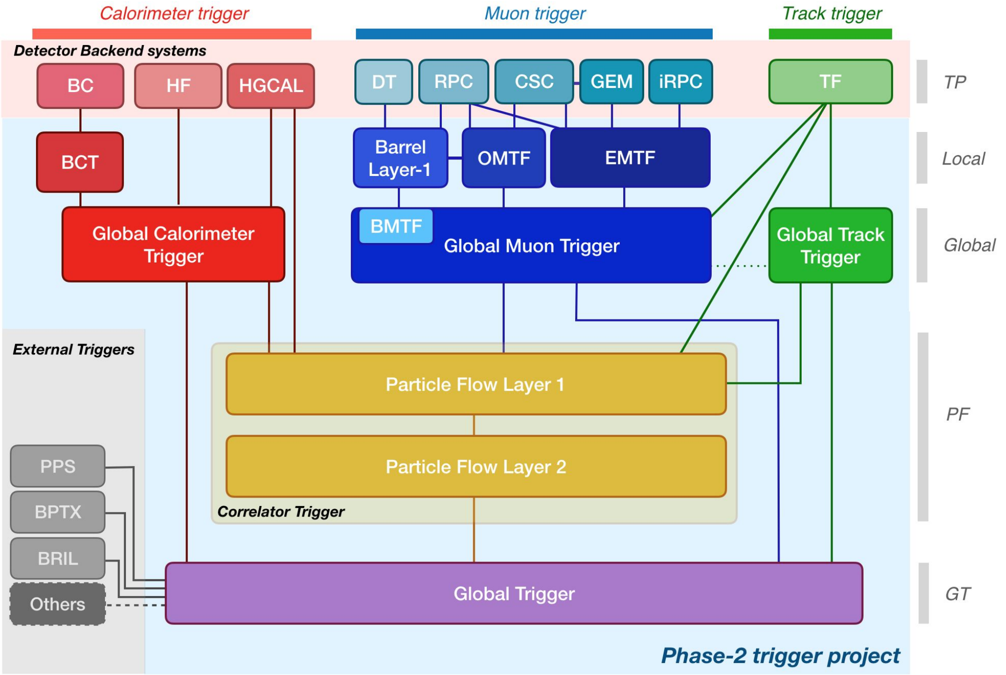
Four complementary trigger types

- Calorimeter Trigger
- Muon Trigger
- Track Trigger
- Particle Flow Trigger

Serenity boards intended to be used in

- HGCAL
- DTC, GTT
- Correlator L1, Correlator L2
- GT
- MTD, BRIL

More than 700 cards in total



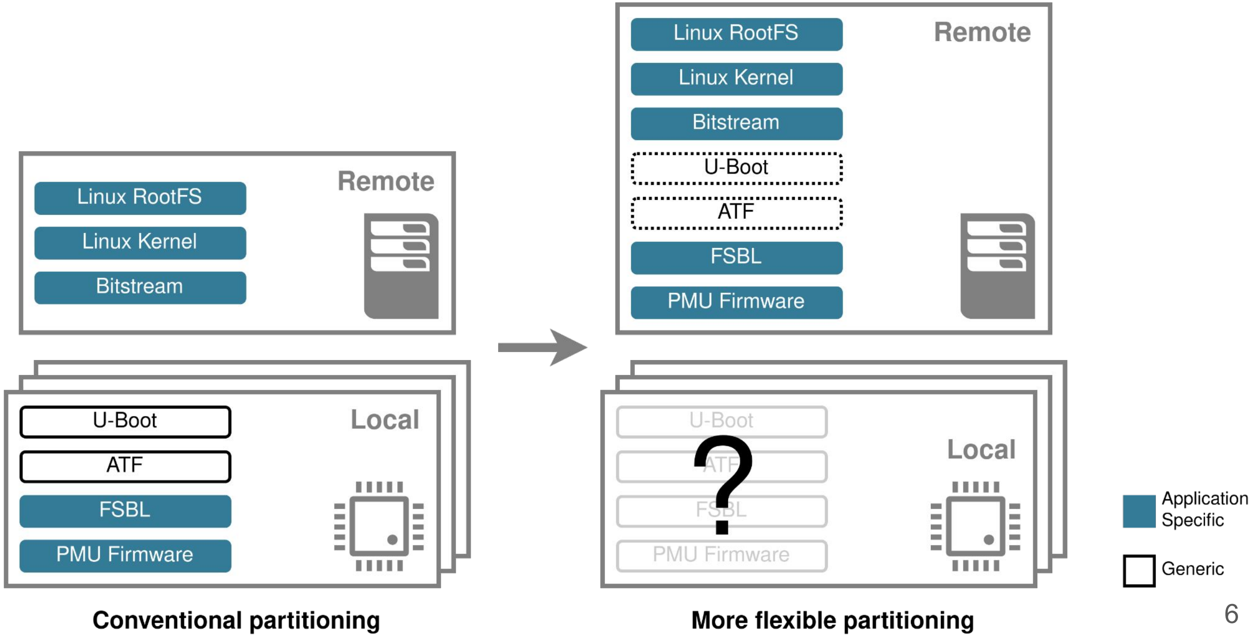
https://lss.fnal.gov/archive/2020/pub/fermilab-pub-20-712-cms.pdf

[See T. Mehner et al.,TWEPP 2023]
https://indico.cern.ch/event/1255624/contributions/5443848/

Why Split Boot?

Challenge: Distributed systems with many ZynqMP devices are difficult to deploy and update due to application-specific boot code in PMU Firmware and FSBL

Idea: Move project specific settings (e.g. PS / PL interface) to remote. Base configuration comparable to BIOS / UEFI on desktop PC (fundamental boot capabilities to enable network boot)



Recap: Split Boot v1

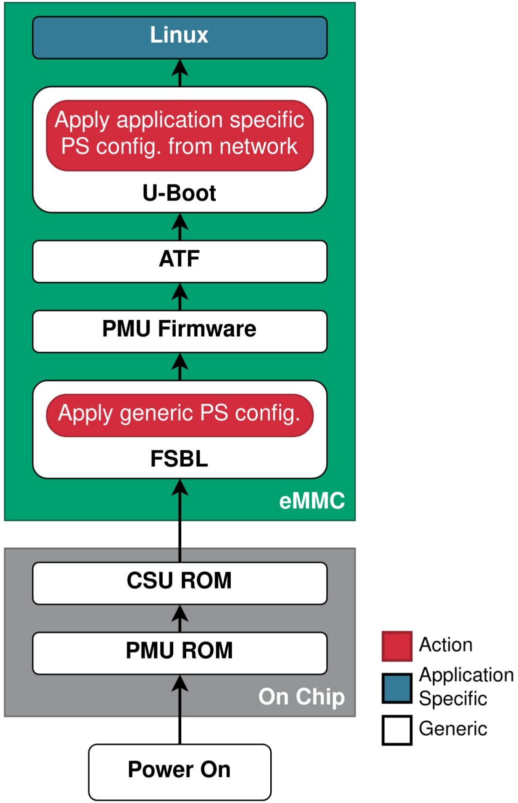
Idea: Load a generic PS-configuration in the FSBL and expand it with **application-specific** information from the network in **U-Boot**.

Advantages:

- Fully network based booting
- Easy and quick to update application specific files and configurations, potentially for many boards, in a single location
- Easy and quick to replace broken boards

Limitations:

- Not everything can be reconfigured (RAM, network interface, …)
- Custom files mean additional effort when switching toolset versions
- Usually the PMU Firmware is not created independent of a project



https://arxiv.org/abs/2301.05642
https://gitlab.cern.ch/split-boot/split-boot-example-zcu102
https://indico.cern.ch/event/1139381/contributions/4788797/attachments/2436369/4172790/2022-05-03_SoC_Interest_Group_Meeting_Update_ZynqMP_activities.pdf

New: Split Boot v2

Idea: Use a generic software stack (QSPI) to **synchronize** the main software stack (eMMC) with the network at boot time. Then, trigger a **soft reset** and boot the main software stack.

Advantages:

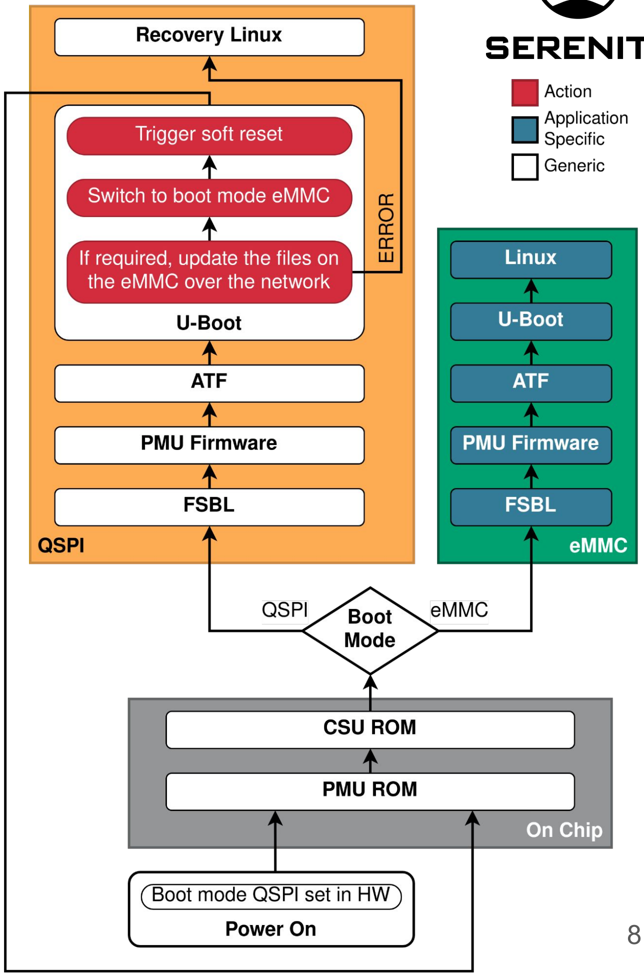
- Same advantages as with Split Boot v1
- More reliable and stable
- No custom files, tools or workflow needed
- No additional difficulties when changing Vivado versions
- Always recoverable (Recovery Linux)

Limitations:

- Two boot media are needed
- About 17 seconds longer boot (incl. 2 sec. for user interrupt)

https://gitlab.cern.ch/split-boot/split-boot-v2-example-kv260

SERENITY



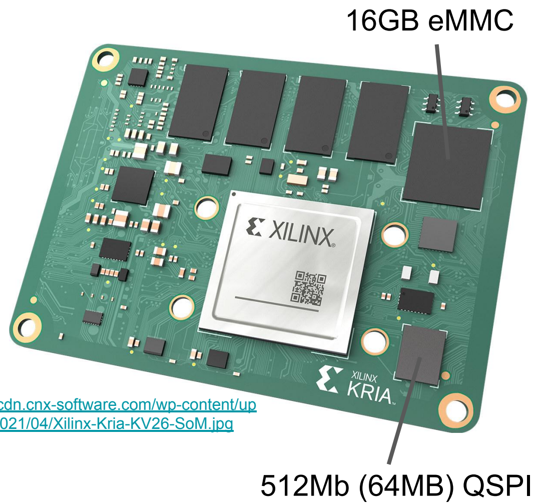
Split Boot v2 on Kria K26 SoM

The Kria SoM meets all requirements, but Split Boot v2 works **on all ZynqMP** devices with two sufficiently large boot media

- Kria K26 SoM has QSPI flash and eMMC storage on board
- Full software stack fits in QSPI flash, despite relatively small size
- External boot media such as an SD card can also be used

Example eMMC Usage	
boot.bin	~ 9.1M
boot.scr	~ 2.8K
uboot_rt_env_init.cfg	~ 155
uboot_rt_env_main.cfg	~ 47
CentOS File System	~ 3.3G
total	**~ 3.4G (~20%)**

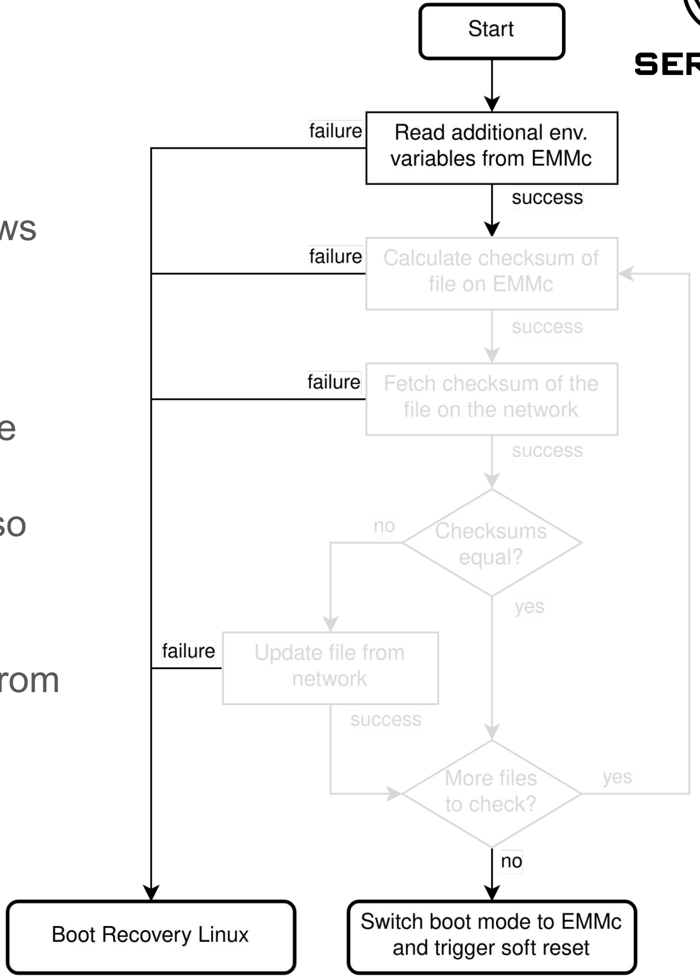
Example QSPI Usage	
FSBL	~ 500K
PMU Firmware	~ 500K
ATF	~ 150K
U-Boot	~ 9.0M
Devicetree Blob	~ 50K
boot.scr	~ 3K
Linux Kernel + INITRAMFS	~ 29M
Bitfile	~ 7.5M
total (boot.bin)	**~ 38M (~60%)**



https://cdn.cnx-software.com/wp-content/uploads/2021/04/Xilinx-Kria-KV26-SoM.jpg

Integration into U-Boot

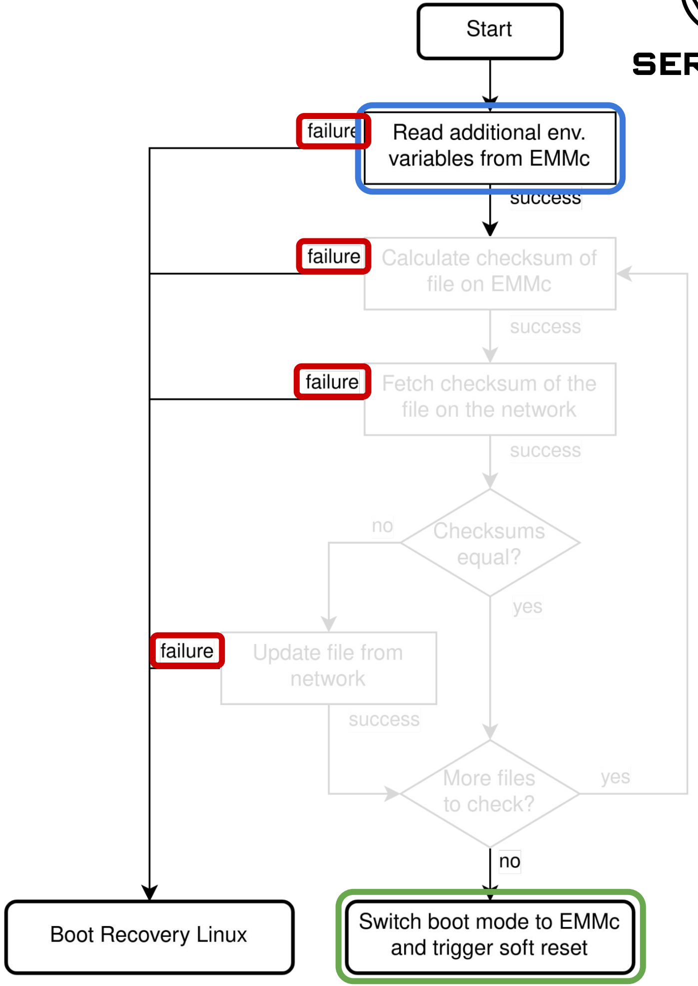
- U-Boot is the first stage in the boot process, which allows easy loading of files from the network
- Procedure implementable in U-Boot script
  - custom `boot.scr`
- U-Boot environment is extended with a configuration file
  - e.g. IP address of the TFTP server
- The EEMI interface of the PMU Firmware is extended so that it can be instructed to switch the boot mode
- The system boots into recovery Linux if any step fails
- If errors occur, they are recorded in a shared memory from where they can be read out with the Recovery Linux



Integration into U-Boot

Three custom U-Boot commands

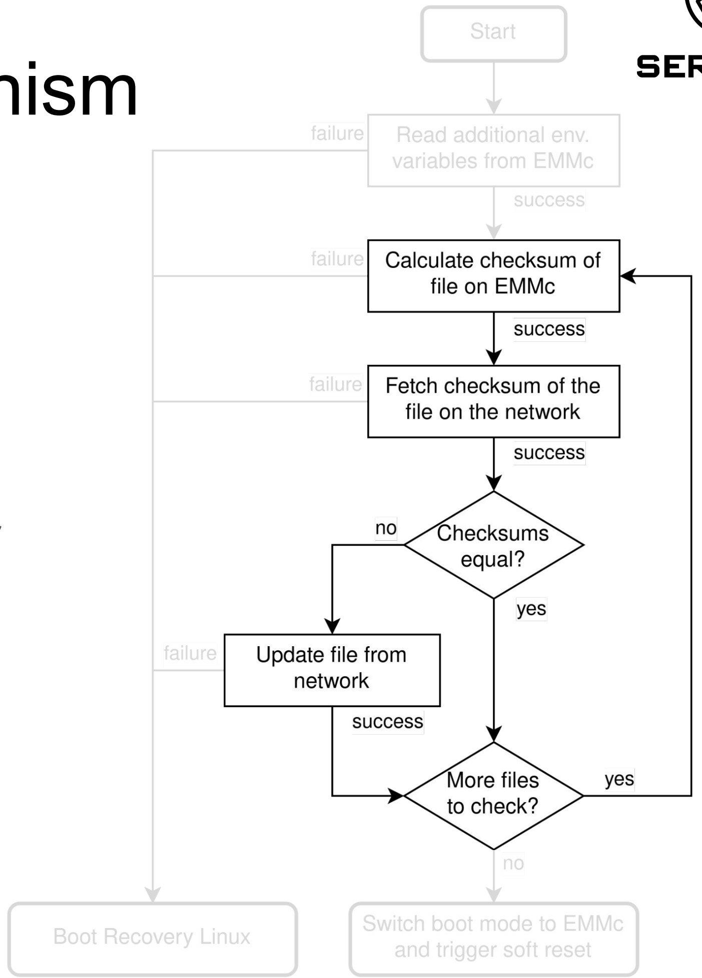
- `file2env`
  - To add environment variables from file
- `print2mem`
  - To pass error messages to the Recovery Linux
- `switchbm`
  - To switch the boot mode and trigger a soft reset



The Automatic Update Mechanism

Idea: Only update files that are not up to date anymore

- DHCP and TFTP capabilities of U-Boot are used to download files
- Checksums are used to save bandwidth when validating files
- Download of complete files only it the checksums differ
- For each file to be checked for actuality, the server holds the pre-calculated CRC32 checksum in a file
    - E.g. `boot.bin` and `boot.bin.crc32`



The Automatic Update Mechanism

To prevent the system from entering a state from which it cannot be recovered without physical access, **not all files** should be automatically updated

- The file `uboot_rt_env_init.cfg` is used to configure the QSPI based U-Boot
  - If it is corrupted, it may be impossible to boot the recovery Linux
  - Updates to this file should be very rare
- It is possible to update all files on QSPI and eMMC manually from Linux

Minimum content TFTP Server
`boot.bin`
`boot.bin.crc32`
`boot.scr`
`boot.scr.crc32`
`uboot_rt_env_main.cfg`
`uboot_rt_env_main.cfg.crc32`

Minimum content eMMC
`boot.bin`
`boot.scr`
`uboot_rt_env_init.cfg`
`uboot_rt_env_main.cfg`

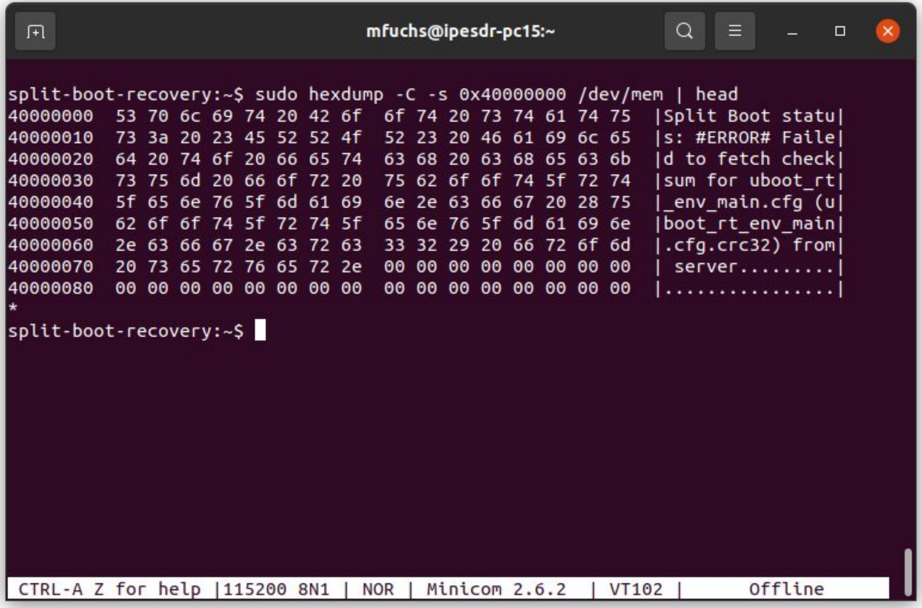
Only the minimum required files are listed here. Additional files can easily be added.

Recovery Linux

A minimal PetaLinux is used to provide the easiest and most comprehensive access to the system in case of problems

- Stored entirely in the QSPI Flash
  - ~30MB in size
- Temporary INITRAMFS as Rootfs
- Access to QSPI and eMMC is possible
- SSH access
- Can be booted manually from U-Boot
  - `ZynqMP> bootm ${kernel_addr_r}`

This image can also be customized if some specific drivers etc. are needed

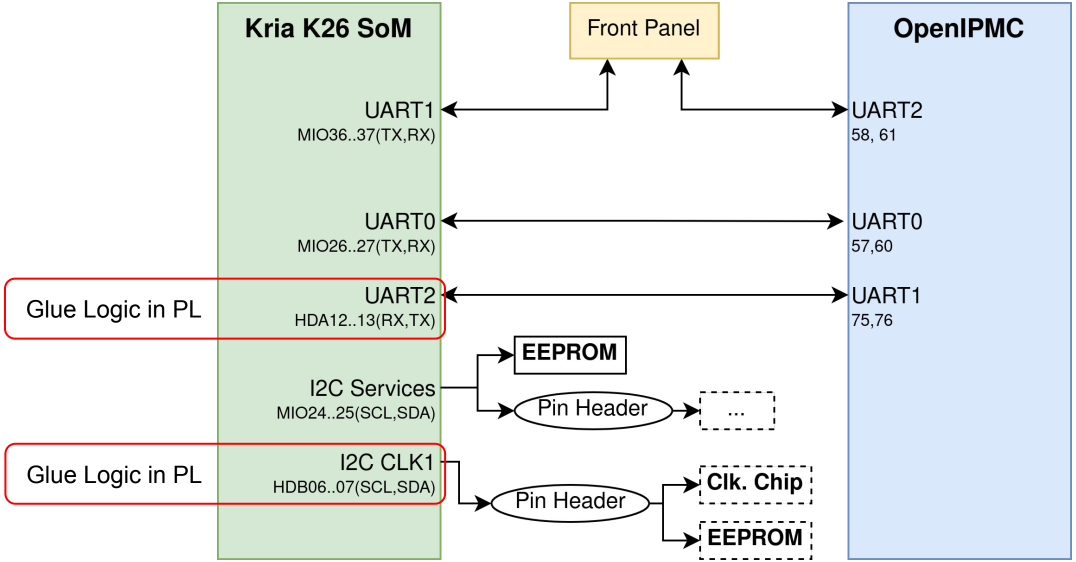


Reading an error message from U-Boot in the shared memory

Usage of the PL in the Initial QSPI Boot Phase

SERENITY

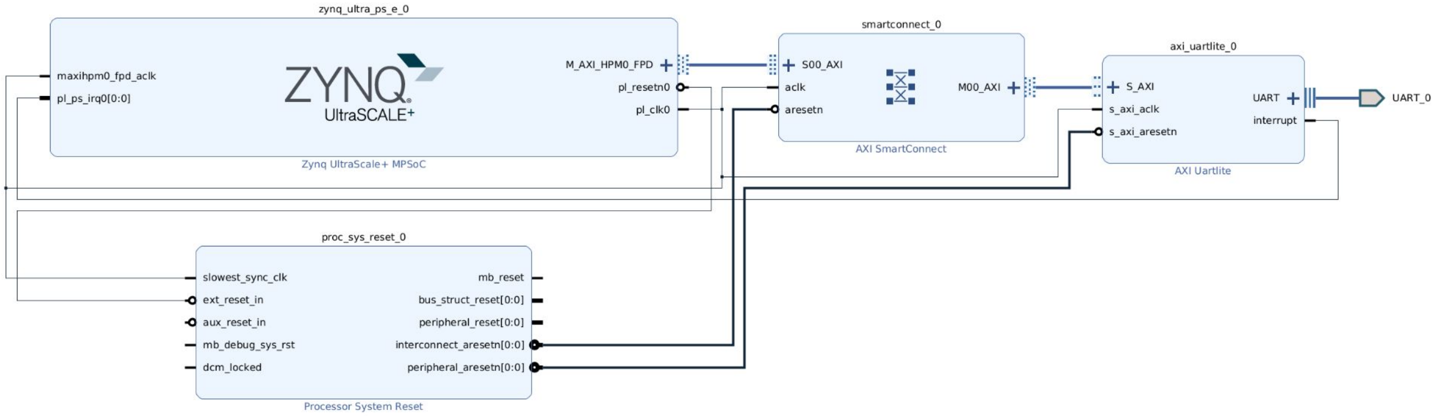
- Early access to devices sometimes necessary
  - Configuring clock chips
  - Reading / Writing EEPROMs
  - …
- Glue Logic
- Example: Serenity-S1
  - UART connection to request information from IPMC
  - PL-I2C to e.g. configure clock chip or read EEPROM



Kria SoM and IPMC connections on the Serenity-S1

Usage of the PL in the Initial QSPI Boot Phase

- Dedicated PL configuration can be used in the first phase of the boot process (QSPI boot)
- Stored e.g. in QSPI image
- Loaded e.g. by the FSBL
    - Usable by FSBL and U-Boot
- Temporarily usable until soft reset



Example: Temporary PL configuration to communicate with the IPMC module via UART on Serenity-S1

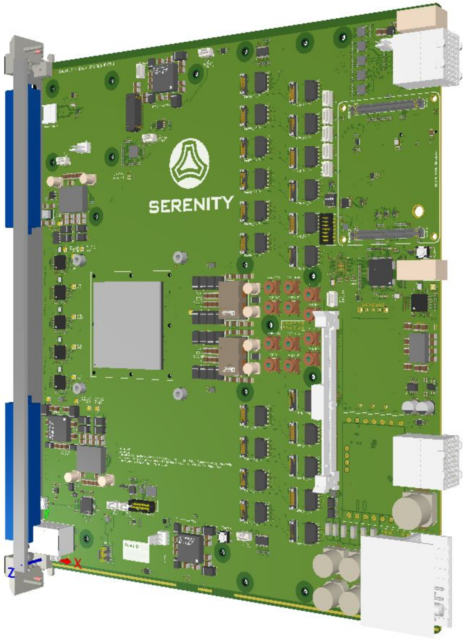
Conclusion

Serenity-S1

- Developed mainly for CMS HL-LHC Level-1 Trigger
- More than 700 units expected to be produced in the long term
- PCB currently in production (pilot production run)
- Tests planned and prepared but not yet executed

Split Boot v2

- New development that takes a different approach than Split Boot v1
- More reliable and stable
- ZynqMP always recoverable
- Easier to update Vivado toolset version



Feel free to browse our Split Boot v2 example Project: https://gitlab.cern.ch/split-boot/split-boot-v2-example-kv260

Thank you for your interest!