Linux for Xilinx Zynq Ultrascale+ based Embedded Systems in the CMS DAQ Network

Authors: Keyshav Suresh Mor, Petr Zejdl and Marc Dobson
Contact: keyshav.suresh.mor@cern.ch

Motivation: CMS Phase-II Upgrade
- Upgrade of the CMS detector and electronics for HL-LHC (High Luminosity LHC) in 2024
- Extensive use of embedded systems (SoC) in the new electronics
- Devices capable of running firmware grade Linux OS
- For control, configuration and monitoring
- System requires network connection
- Mostly based on Xilinx Zynq Ultrascale+ SoC
- Deployed at a scale of ~1000 devices
- The scale poses challenges for the integration of SoCs in CMS
- Hardware is not uniform due to the detector layout
- Different software requirement and software life cycle
- OS selection and support
- Network, system administration and scaling challenges

Operating System Issues
- Many Linux versions available for ARM (reference Linux from Xilinx: FATALinux, Yocto, Arche, CentOS)
- Hardware developers may prefer various OS distributions or versions
- Linux System Administration may prefer to support only single OS version (reducing manpower)
- Can existing knowledge be re-used and pooled by using the same OS as PCs at CERN?
- CentOS (RHEL based Linux)
  - Not supported by Xilinx, but
  - Kernel can be used from the Xilinx toolchain
  - Preferably CentOS default kernel could be used with Xilinx specific drivers (requires porting)
  - CentOS root file system remains unchanged

System Administration Issues
Integration Issues
- Devices need to be integrated into the CMS experiment technical and control network
- Network specific settings: IP addresses, DHCP, DNS, NTP
- Sufficient bandwidth for the primary task of control and supervision of the CMS electronics and for services (NFS, logging, etc...)

System Administration
- Centrally administered OS with regular updates and security patches. CentOS for common knowledge with PCs?
- Central Configuration Management System as for PCs: Puppet
- Mechanism to update Root file system or Board specific packages with Puppet
- Same user database (ldap, kerberos) across all platforms

Hardware Issues
- Reliable, fault tolerant booting mechanism
- Automatic failover to golden image in case of failure to boot
- HW address (MAC) in standard EEPROM for all board designs
- Reliable and fault tolerant mechanism to update files on SD card (FSBL, U-Boot, and maybe firmware)

Xilinx Zynq Ultrascale+
- Zynq devices tightly integrate the programmable logic (PL) with the processing system (PS).

CMS DAQ Proposed Solution
- Benefit from common knowledge across CMS/CERN and use a common approach for all the issues
- Minimize the manpower required for software development, integration and administration
- Use centrally managed Linux distribution for SoC hardware at CERN

Scalings Issues

Power On Sequence
- Platform Management Unit (PMU)
  - Supplies power and down peripherals
  - Manages clocks, resets and initialization PLLs
  - Manages sleep modes, monitors the system
- First Stage Boot Loader (FSBL)
  - Loads the FSBL from a boot device into On-Chip Memory (OCM)
  - Enables ARM CPU

Files used for booting Linux on Xilinx Zynq devices
- PL Bitstream (.bit): The bitstream is used to program the programmable logic which would be eventually controlled by the processing system (PS).
- Hardware Description File (.sdt): File which describes the hardware, register and memory offsets being programmed by the bitstream, used to generate device tree.
- First Stage Boot Loader (fsbl.elf): Initiates peripherals and memory before handing over control to the ARM Trust Firmware, which then loads the U-Boot in the OCM.
- U-Boot (.uboot.elf): Intermediate bootloader which loads the device tree and the kernel in the memory.
- Device Tree (.system.dtb): Device tree describes underlying hardware, register and memory offsets (via a BDF) to the Linux kernel when the kernel executes.
- Linux Kernel (.image.ub): Linux kernel is responsible for providing device tree for the configured underlying hardware and activating services and features supported by the hardware. Linux Kernel works as interface between uCLinux and hardware.
- Root Filesystem is the file system where the driver, files, software and services used by the user are located and installed. init is the first process which executes after kernel finishes executing.