ATLAS TDAQ SysAdmins SoC status report

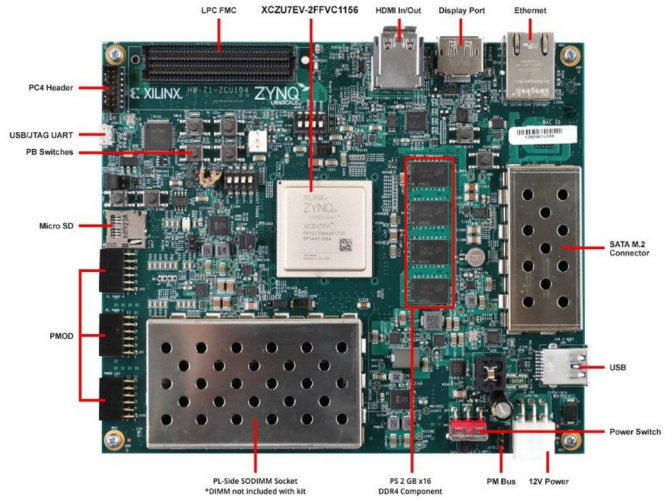
SoC Interest Group Meeting - November 21st, 2021

ATLAS TDAQ SysAdmins

-

Quentin Duponnois on behalf of ATLAS TDAQ SysAdmins

Our setup



Configuration

- USB-JTAG FT4232H
- Dual Quad-SPI flash memory
- MicroSD Card

Memory

- PS DDR4 64-bit Component
- Quad-SPI flash
- Micro SD card slot

Control & I/O

- 4x directional pushbuttons
- DIP switches
- PMBUS, clocks, and I2C bus switching
- USB2/3

Expansion Connectors

- FMC LPC (1x GTH)
- 3 PMOD connectors
- PL DDR4 SODIMM Connector – 64 bit

Communication & Networking

- USB-UARTs with FT4232H JTAG/3xUART Bridge
- RJ-45 Ethernet connector

Display

- HDMI 2.0 video input and output (3x GTH)
- DisplayPort (2x GTR)

Clocking

- Programmable clocks, System clock, user clock
- Jitter attenuator

Power

- 12V wall adaptor or ATX

We are using Zynq UltraScale+ MPSoC **ZCU104** Evaluation Kit to test the configuration. For more information: https://www.xilinx.com/products/boards-and-kits/zcu104.html

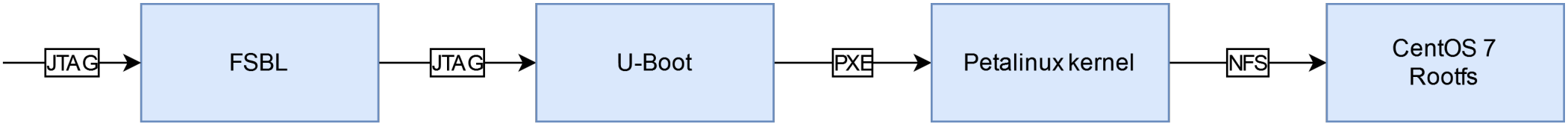
For the NFS, FTP server is hosted on an mini PC (OptiPlex 7060) and it's also our Serial/JTAG connection to the board.

JTAG/Serial

NFS/TFTP

Already done

- 1st Step: *Classic Boot process*



- FSBL: First Stage Boot Loader, is the Xilinx official Secondary Program Loader
- Das U-Boot: Open Source Bootloader, with network capability

  More information about Das U-Boot http://www.denx.de/wiki/U-Boot/

- PetaLinux, software created by Xilinx

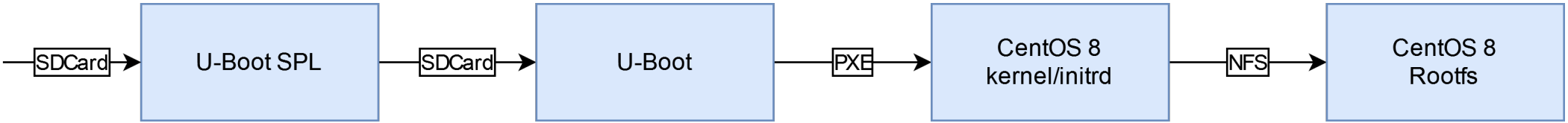
For the PXE part we currently overwrite the value of the “next server” provided by the DHCP, by setting the “serverip” environment variable in u-boot.

The build tool was the one provided by Xilinx: Vivado, Petalinux

This build was easy, but the build chain is very heavy (~60Go), and didn’t support CentOS kernel.

Current state

- 2nd Step: *Modified Boot process*



- U-Boot SPL: Secondary Program Loader created by the u-boot community
  it's lighter than FSBL, and it's built during the build of U-Boot (easiest to automate)
  more information about U-Boot SPL on ZynqMP:
  https://lucaceresoli.net/zynqmp-uboot-spl-pmufw-cfg-load/

- Das U-Boot: Open Source Bootloader, with network capability

- Centos 8 kernel (version 4.18) can be recompiled to support Xilinx SoC

Currently, we are working on the CentOS kernel/initrd configuration to find a way to mount the rootfs from SDcard or NFS.

Build tool

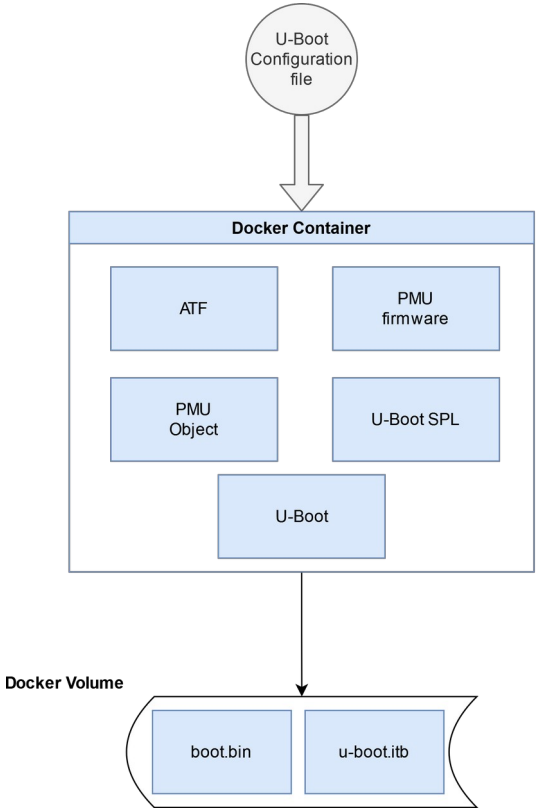
To build U-Boot and U-Boot SPL, the ATLAS TDAQ SysAdmins team has created a Docker container to easily share this tool.

You just have to provide an U-Boot configuration file and it will create:

- boot.bin (U-Boot SPL, ATF, PMU)
- u-boot.itb (U-Boot)

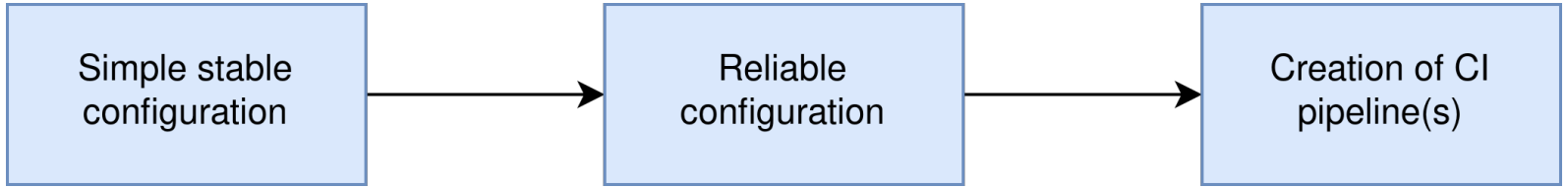
The tool builds all the components needed by Xilinx to load U-Boot:

- ATF (ARM Trusted Firmware)
- PMU firmware and objects
- Compile U-Boot SPL and U-Boot



The tool is available on gitlab for testing:
https://gitlab.cern.ch/soc/atlastdaqsysadmins_cc-uboot

Future steps



The goal of the SysAdmins team is to integrate the SoC inside our current environment (Puppet, ConfDB)

ConfDB is our configuration and deployment tool for image.

We also evaluate the possibility to build the U-Boot image and Kernel/initrd with Koji or rpmci provided by CERN

For any question: *atlas-tdaq-sysadmins@cern.ch*

Reminder “This project is planned to be end for phase 2”