2ⁿᵈ SoC workshop CERN – Added Notes – June 11, 2021

https://indico.cern.ch/event/996093

Added Notes – June, 11th:

- A. D. Tinoco Mendes. Report on SoC Activities in the CMS HGCAL:
  - Using Trenz TE0820 2CG/3EG/4CG in ~6 different test system types (using ~3 different carrier boards) but also a bigger one for GTY access (integrated in Serenity ATCA board)
  - Incrementally added the ability to trigger external systems (like lasers) and accept and timestamp external trigger inputs, etc.
  - Eventually, there may be O(100) small systems performing production testing across the world.
    → Happy with Trenz environment/support :)

  - Petalinux used for BOOT.INI+uboot+kernel(and .ko) generation including ethernet MAC address discovery as part of Trenz "recipes". The sooner we got rid of PL the sooner we were happier :)
  - CentOS 7 root fs (thank you!)
  - Using SD card (no network dependency mandatory for some test sites) → While BOOT.INI and uboot could probably be built once in PL for a long time, we'd really like to have a non-PL kernel build path for security; we must be able to access to a custom kernel command line otherwise uio (see below) will not work. Moving from SD card to eMMC could be interesting but not a priority.

  - Firmware and support software files built in CI
  - We keep vivado docker containers in our repo that can be shared.
  - Builds are run in a dedicated CI runner machine.
  - Using own set of py/tcl build scripts and vivado-in-docker that produce .bit .dtbo (from automated customisation of dtsi with project-specific device-tree additions) and uHAL .xml files (for software) → In the works: wrapping it all in .rpm distributed on EOS site along with a small tool to load .bit and .dtbo and install .xml files in canonical paths.

  - Software access to AXI devices with IRQs (in production) and DMA (demonstrated) via uio+uHAL.
  - Test system DAQ system based on a couple of ZMQ patterns so that Zynq is mostly a server for configuration and data-taking, but data is processed in large computer outside (analyse, make plots, etc).
    → Compiled in the Zynq; we are really eager to use cross-compilation or any other solution that would allow for a docker solution for CI integration. Docker would definitely make it simple to compose containers with things like uHAL, ROOT, as needed. Eventually, .rpm deployed along the fw .rpm.

  - Testing: presently by hand.

→ Long term goal for CI flow to include installing fw/sw .rpms and run test suite on real hardware. (Like lpGBT team did!)

- ○ Untouched: we have not looked at anything related to fallback boot, encrypted booting, using the realtime cores, etc.