Quench detection firmware security

R. Denz for WP7
Firmware security - development

- Firmware development
  - Generic FPGA code only, no vendor specific libraries
  - Dedicated tools supporting code development including automatic error correction e.g. SIGASI™ plug-in for Eclipse™ IDE
  - Strict version control (also for all hardware designs) → SVN/GIT
  - Exhaustive documentation → Confluence™ collaboration software during development → technical specification / user manual after completion stored in EDMS
  - Partially automatic firmware generation using standard, already verified building blocks e.g. for ADC readout
  - Core of the FPGA code is the finite state machine (FSM) ensuring the proper execution of the code
    - FSM is protected by watchdogs (hardware and/or software)
  - Device configuration data are transmitted continuously and stored in the LHC logging database (defined as a QPS signal)
Firmware security - verification

- Functional and type testing
  - System is regarded as a black box and response to external stimuli i.e. analog input signals verified with respect to the functional specification
  - Testing and development of test systems typically done by team members not involved in the firmware development process

- Code verification
  - Performed by external specialist(s); in the QDS case within a long term collaboration with AGH Krakow
  - Verification of code integrity, identification of potentially dangerous constructs, analysis of the synthesis process (not always perfect, results may depend on the version of the tool chain)

- During operation
  - Verification of device configuration, signal integrity checkers (already successfully in use during RUN2), automatic self checks (→ part of QPS_OK signal → affects power and beam permit)