Firmware development

How do we collaborate on firmware?

Common repository

HDL style guides

Strategies for collaborative development

Common f/w repository

- L1Calo has discussed and tried to set up a common code management system before
  - Synchronicity, CVS
  - Never universally adopted
  - Each developer used local copies of code
- We need to set something up now
  - Developers at different institutes working on same modules/FPGAs for upgrade
  - Need long-term solution for maintaining existing hardware (until ca. 2020)
- Questions:
  - What system do we use?
  - What do we put on it?
 - How do we use it for collaboration?
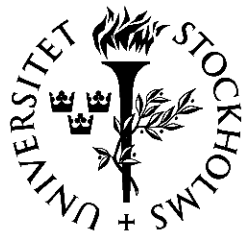
What system do we use?

- SVN : popular version control system for software and for firmware
- CERN has a central SVN service
- Is it suitable for us? Easy enough to use?
- Are there other attractive options?
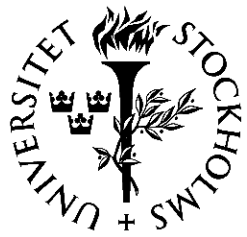
What do we put in it?

- At a minimum:
  - HDL libraries
  - Constraint files (e.g. UCF)
  - Test bench framework
  - Documentation
  - Notes on synthesis, implementation settings
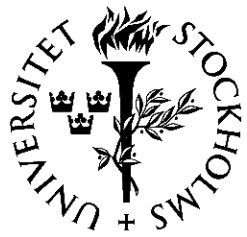- Also libraries of general-purpose VHDL cores
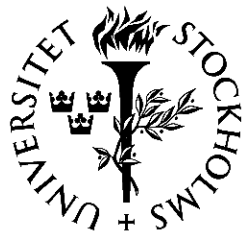
VHDL style guide

- Important for unified code development
- Draft 0.4 distributed in 2009 (copy in agenda)
- Should revisit, and finalize
- Should develop templates for top-level designs and test benches
  - Common top-level entity declaration for different firmware types on same FPGA
  - Signal names matched to documentation

How do we collaborate?

- Suggestion for organization:
  - Identify person responsible for each FPGA design
  - That person makes sure the design is complete and functional, responsible for synthesis/implementation and simulation
  - Other collaborators can develop VHDL cores for use in the design, but "responsible" person imports those cores into design
- Active support/discussion among developers
  - Wiki? Mailing list? Forum?
- Other ideas?

Discuss....