A time service for EPICS

specification for replacement of drvTS

Babak Kalantari
Timo Korhonen

Fall 2005

Paul Scherrer Institute

Problems & weaknesses of *drvTS*

- Lack of fail-over mechanism for HW-assisted clock
- Master clock could NOT synchronize with external world (NTP)
- Hard-coded HW interface routine names (totally HW dependent)
- Very little diagnostics (just *TSreport*) and no controls over clock parameters
- No way to force an IOC to maintain a direct clock (NTP sync) in presence of a master clock
- If NTP server is not available at boot time IOC looks just for a master for ever and NOT for an NTP server (should look for both)

Solutions & enhancements

- Task-switching mechanisms: automatic for fail-over, manual to resume a sync clock
- Implementation of a PLL to keep master clock IOC in sync with NTP
  (clock frequency correction is communicated via UDP messages periodically)
- Provide a clean way of HW interface to define interrupt & error handling routines for timing boards (like a device support)
- Monitoring & controls of clock parameters via soft device supports
- Possibility to force an IOC to maintain a direct clock (NTP sync)

Status & progress

- Man power: 0.2 person (Babak) + 0.1 person (Timo)

  (regular discussions, reviewing specs, development & tests) they have also other tasks to fulfill !

- Done:
  - Reverse engineering
  - Writing the specs
  - Implementation & tests:

  HW fail-over, task-switching mechanism, diagnostics & controls, master PLL sync with NTP

- To be done:
  - Implementation & tests: Plug-in timing HW interface supports, force direct slave
  - Publishing the specs to the EPICS community (EPICSWIKI)
  - Rewriting the whole timestamp software support
  - Merge with General Time (need a meeting to sort out the details)

Open issues

- What to do when we have two separate subnets and IOC’s on both may need synchronized clocks ? (e.g. at SLS we have a beamline net and a machine net)

- Do we ever support auxiliary master IOC concept and if yes, how?

Difficulties

- The operating IOC’s may be affected during test & development and screw up their clocks
- Testing some behaviors and reactions are time consuming (PLL, stability, etc.)
- To test each development an IOC core has to be built
- .... and lack of time …