ABSTRACT

Given the time critical applications, the use of PXI and cRIO platforms in the accelerator complex at CERN, require the integration into the CERN timing system. In this paper the present state of integration of both PXI and cRIO platforms in the present General Machine Timing system and the White Rabbit Timing system, which is its successor, is described. PXI is used for LHC collimator control and for the new generation of control systems for the kicker magnets on all CERN accelerators. The cRIO platform is being introduced for transient recording on the CERN electricity distribution system and has potential for applications in other domains, because of its real-time OS, FPGA backbone and hot swap modules. The further development intended and what types of applications are most suitable for each platform will be discussed.

GMT on PXIe

The purpose of the General Machine Timing System is to trigger and synchronize equipment and software actions, timed according to the accelerator cycles.

The GMT has several interfaces to other subsystems of the accelerator control system:

White Rabbit

The White Rabbit node is an open hardware design, accessible from CERN’s open hardware repository.

Engineers at University of Zürich have used the WR design to build a cRIO module.

White Rabbit on cRIO

Hardware

- cRIO platform
- FPGA module
- cRIO-WR module
- cRIO-DI card

Software

- LabVIEW RT
- Drivers in LabVIEW FPGA for the control and readout of the White Rabbit timestamp
  - Code optimised for minimum delay and jitter

Wood Rabbit on PXI

Hardware

- PXI Express crate
- Timing receiver card
  - CTRP (PMC format)
  - PXI trigger card
  - cPCI carrier

Software

- Hypervisor
  - Run Linux and LabVIEW RT simultaneously
  - Communication through shared memory
- Linux (SLC6)
- GMT drivers (libCTR)
- LabVIEW RT
- DAQmx

CONCLUSION

Future developments point in the direction of having Linux RT on both cRIO and PXI platforms, which will allow the coexistence of the software in a single OS, simplifying the configuration, operation and maintenance of the Hypervisor systems, increasing their performance. We have also tested and optimised the FPGA driver to get optimum performance, which resulted in a minimum of 850 ns delay and 2 ns jitter.