



Contribution ID: 103

Type: Oral

## NA62 Trigger System

*Wednesday, 23 May 2012 08:55 (20 minutes)*

The NA62 trigger system works with 3 trigger levels. The first trigger level is based on hardware, central trigger unit, so called L0 Trigger Processor (L0TP) and Local Trigger Units (LTU), which are located in the experimental cavern. Other two trigger levels are based on a software, computer farm, located on a surface. The L0TP receives information from 12 triggering sub-detectors asynchronously via ethernet, processes it, and then transmits a final trigger decision synchronously to each sub-detector through Trigger and Timing Control (TTC) system. The interface between L0TP and TTC system, which is used for trigger and clock distribution, is provided by Local Trigger Unit board (LTU). The LTU can work in two modes: global and stand-alone. In the global mode, the LTU is providing an interface between L0TP and TTC system. In the stand-alone mode, the LTU can fully emulate L0 processor and such provides independent runs for each sub-detector for testing or calibration purposes. In addition to emulation functionality there is implemented functionality which allows to synchronize the LTU with the L0 processor and the TTC system. For testing and debugging purposes there is implemented Snap Shot Memory (SSM) interface which can work in an input or an output mode. The trigger system will be permanently monitored by reading counters, with considerable built-in redundancy, at regular intervals.

**Primary author:** Dr KRIVDA, Marian (University of Birmingham)

**Co-authors:** Dr LAZZERONI, Cristina (University of Birmingham, UK); Dr LIETAVA, Roman (Comenius University, Bratislava, Slovakia); Dr BLAZEK, Tomas (Comenius University, Bratislava, Slovakia); Dr CERNY, Vladimir (Comenius University, Bratislava, Slovakia)

**Presenter:** Dr KRIVDA, Marian (University of Birmingham)

**Session Classification:** Front End, Trigger, DAQ and Data Management

**Track Classification:** S4 - Front End, Trigger, DAQ and Data Management