

Use of Triple Modular Redundancy (TMR) Technology in FPGAs for the Reduction of Faults due to Radiation in the Readout of the ATLAS Monitored Drift Tube (MDT) Chambers

J. Dubbert¹, M. Fras¹, H. Kroha¹, O. Reimann¹, R. Richter¹, B. Weber¹

¹Max-Planck-Institut für Physik, Föhringer Ring 6, 80805 München, Germany

The readout of the ATLAS MDT chambers uses a XILINX Virtex-II-2000 FPGA as the main processor for data transmission, channeling data from up to 432 drift tubes via optical fibers to the Readout Drivers (ROD) in the experimental hall. The FPGA is sitting on the Chamber Service Module (CSM), which in turn is mounted onto the MDT chambers in the ATLAS cavern. The FPGA is thus exposed to a considerable rate of strongly ionizing tracks, in particular in the highest- region of the end-caps of the ATLAS muon spectrometer. The resulting SEUs will not only corrupt user data but may also change the firmware code running on the FPGA, which may lead to malfunction of the chip. A common mitigation scheme for SEUs is Triple Modular Redundancy (TMR). Functional blocks are implemented three times in parallel and fed by the same inputs. A majority voter uses the three independent outputs to determine the correct logic value, even if one of the blocks suffers from an upset. This provides good protection against an SEU in the user data and some protection against an upset in the configuration memory. The drawbacks of TMR are increased usage of logic and routing resources, increased power dissipation, and harder timing closure.

We used the TMRTool software package supplied by the XILINX corporation to apply TMR to the most critical parts of the design. Individual modules can be configured to be triplicated or to be untouched by the TMR process. In addition, the software takes care of some FPGA-specific elements for which SEUs are particularly problematic. In our case the code generated with TMRTool uses about 92% of the FPGA logic resources, whereas the normal usage is about 41%. The TMR firmware was tested in the laboratory and at a cosmic ray test facility. It worked well in both cases.

A supplementary technology, called scrubbing, consists in continuously re-writing the configuration memory from a source, which is highly immune to SEUs. The upset configuration bits are thus permanently overwritten by the correct values. While TMR is a good mitigation for a few SEUs, scrubbing continuously corrects wrong bits, preventing an accumulation of SEUs. In Xilinx Virtex-II devices, this can be done while the FPGA is running normally. For the CSM we use a self-hosted scrubbing unit, which is sitting in the same FPGA that it is re-configuring. Benefits and disadvantages of this scheme will be presented.