

DSP

Digital Signal Processors (DSPs) are microprocessors with the following characteristics:

1. Real-time digital signal processing capabilities. DSPs typically have to process data in real time, i.e., the correctness of the operation depends heavily on the time when the data processing is completed.
2. High throughput. DSPs can sustain processing of high-speed streaming data, such as audio and multimedia data processing.
3. Deterministic operation. The execution time of DSP programs can be foreseen accurately, thus guaranteeing a repeatable, desired performance.
4. Re-programmability by software. Different system behaviour might be obtained by re-coding the algorithm executed by the DSP instead of by hardware modifications.

In the pixel detector, a single Master and four Slave DSPs reside on the board and are utilized for the control and coordination of on- ROD operations, as well as performing high-level tasks such as data monitoring and module calibration. Once configured, the ROD FPGAs handle the event data-path to ATLAS Level-2 without further assistance from the DSPs. ROD reset and the VME interface are handled by an FPGA referred to as the Program Reset Manager (PRM). At power-up, the PRM activates the host-to-ROD VME interface and resets the Master DSP (MDSP). Once the MDSP has booted, configuration information is transmitted for the Controller FPGA as well as the data-path FPGAs as the Formatter, the Event Fragment Builder and the Router. The farm of Slave DSPs (SDSPs) are loaded with boot code from the host via the MDSP.

The MDSP is a Texas Instruments 6201 integer DSP running at 160 MHz with two internal 64 kB blocks of memory and an additional 32 MB of (slower) external memory. The MDSP has ROD and BOC registers connected to one of its External Memory InterFaces (EMIFs) thereby allowing any ROD or BOC registers to be set from the host via the MDSP. The MDSP runs software to perform system functions while the FPGA performs real-time functions. Module configuration is performed by the MDSP using its multi-channel buffered serial ports (SP0 and SP1); configuration data is passed to the MDSP from the host. In calibration mode the MDSP serial ports are also used to send triggers. During normal ATLAS running the trigger and event description information (Level-1 ID, bunch-count ID and triggertype) is supplied to the ROD by the TIM. The TIM trigger is detected inside of the Controller and expanded into the trigger codes required by the Pixel and SCT modules. The trigger code is then sent out via a 48-wide mask gate and propagated on to the modules.

The SLAC group has four main areas of involvement in the overall DSP effort:

1. Standalone DSP teststand work and development/maintenance of documentation. SLAC maintains a teststand setup where offsite work can be performed easily. The setup is used as a template for other standalone lab setups at CERN and LBNL.
2. Optimization of the code. One of the major successes of the NewDSP default code is the speed with which tasks can be performed. Low-level optimization work is performed in order to further streamline the memory management, and code setup.
3. Validation of the TOT Calibration. Each of the performed scans of the detector, utilising the NewDSP code, need to be accurately validated. This involves detailed comparisons between similar scans taken while running against the OldDsp and NewDsp code, robustness tests of the NewDsp and interpretation of any results. Any differences or mistakes uncovered need to be understood and then the problems solved.
4. S-curve fit validation and TOT calibration on the DSP. First one begins by comparing the offline fit to the fit performed with the NewDsp. The scope of the work is then to understand the S-curve fitting and improve on the basic criteria (speed, quality, exit criteria etc) of the fit. The S-curve fit validation is then to perform a pixel-to-pixel comparison between a root fit and DSP fit for Threshold mean, sigma and χ^2 . The goal being, partly, to understand whether the χ^2 can be used to qualify the S-curve fitting.

Here is a link to the TWiki for the PixelDSP group: <https://twiki.cern.ch/twiki/bin/view/Atlas/PixelDsp>

Here is a link to the TWiki for the SiliconROD group: <https://twiki.cern.ch/twiki/bin/view/Main/AtlasSiliconRodGroup>

I provide some instructions linked below for running tests of the DSP code.