

LHCb Silicon Tracker DAQ and DCS Online Systems

D. Esperante, P. Rodríguez, A. Büchler, A. Keune, A. Bay, F. Blanc, M.O. Bettler, G. Conti, V. Fave, R. Frei, N. Gueissaz, G. Haefeli, J. Luisier, R. Muresan, T. Nakada, M. Needham, L. Nicolas, M. Knecht, A. Perrin, C. Potterat, O. Schneider, M. Tran, C. Bauer, M. Britsch, W. Hofmann, F. Maciuc, M. Schmelling, H. Voss, U. Straumann, J. Anderson, N. Chiapolini, V. Hangartner, C. Salzmann, S. Steiner, O. Steinkamp, J. van Tilburg, M. Tobin, A. Vollhardt, B. Adeva, J. Fungueiriño Pazos, A. Gallas, A. Pazos Alvarez, E. Pérez Trigo, M. Pló Casasús, J. Saborido, P. Vázquez, A. Gong, V. Iakovenko, O. Okhrimenko, V. Pugatch (on behalf of the LHCb collaboration)

Abstract—The LHCb experiment at the Large Hadron Collider (LHC) at CERN in Geneva (Switzerland) is designed to perform precision measurements of b quark decays. The Silicon Tracker (ST) plays a crucial role in reconstructing particle trajectories and consists of two silicon micro-strip detectors, the Tracker Turicensis upstream of the LHCb magnet and the Inner Tracker downstream. The radiation environment and the magnetic field represent new challenges for the implementation of a Detector Control System (DCS) and the data acquisition (DAQ).

The DAQ has to deal with ~272K analog read-out channels, 2K read-out chips and real time DAQ at a rate of 1.1 MHz with data processing at TELL1 level. The TELL1 real time algorithms for clustering thresholds and other computations run on dedicated FPGAs that implement 13K configurable parameters per board, in total 1.17M parameters for the ST. After data processing the total throughput amounts to about 6.4 GB from an input data rate of ~337 GB per second.

A finite state machine based hierarchical control system is the fundamental of the DCS and allows distributed control access and multi-platform use. The implementation of the DCS system for two sub-detectors requires a design which can be used for TT and IT which have a different hardware mapping. With the DCS an operator is able to control the power supplies, to program the read-out chips and to monitor online the state of all the hardware in the read-out chain. It features as well a monitoring of temperature and humidity readings and can take automated

actions on warnings or alarms. To guarantee safe operation a completely independent, hardware-based system is used for the 'vital' alarms to ensure redundancy.

Index Terms—Control system, data processing, hierarchical finite state machines.

I. INTRODUCTION

The LHCb detector [1] placed in the LHC at CERN is a single-arm forward spectrometer to study rare b decays and CP-violation. The LHC is a proton-proton collider working at a bunch crossing rate of 40 MHz. The luminosity in the LHCb experiment will be limited to $2 \cdot 10^{32} \text{ cm} \cdot \text{s}^{-1}$. LHCb features excellent tracking and particle identification capabilities and covers an acceptance out to 250 mrad x 300 mrad around the LHC beam axis.

The Silicon Tracker (ST) [2] is part of the tracking system of the LHCb experiment. It consists of two detectors, the Tracker Turicensis (TT) and the Inner Tracker (IT). Both of them use silicon micro-strip technology with long read-out strips, giving a total active area of 12 m² with ~272k channels. The same front-end chip (the Beetle [3]) and read-out link [4] are used both in the IT and the TT. The ST data acquisition system (DAQ) is able to read events at a maximum rate of ~1.1 MHz, which leads to ~337 GB/s at the Front-End (FE) before real-time processing in the TELL1 boards [5]. After processing the total throughput to the DAQ network is expected to be about 6.4 GB/s.

The TT is a ~150 cm wide and ~130 cm high planar tracking station that is placed upstream of the LHCb 4 Tm dipole magnet and covers the whole acceptance of the experiment (Fig. 1). It is made up of 280 silicon modules with 512 strips each read-out by four Beetle chips. About 340 sensors perform the environmental monitoring of the detector.

The second of the two detectors is the IT. It consists of three stations covering a 125 cm wide and 40 cm high cross-shaped region (Fig. 2) downstream of the magnet and close to the beam pipe. It consists of 336 silicon modules with 384 strips read-out by three Beetles each. The environmental monitoring is performed by ~400 sensors.

The overall detector is controlled by the Experiment Control

Manuscript received May 23, 2009.

D. Esperante, P. Rodríguez, B. Adeva, J. Fungueiriño Pazos, A. Gallas, A. Pazos Alvarez, E. Pérez Trigo, M. Pló Casasús, J. Saborido, P. Vázquez are with the University of Santiago de Compostela, Department of Particle Physics, Campus Sur, Santiago de Compostela 15782, Spain (e-mail: daniel.esperante@usc.es).

A. Keune, A. Bay, F. Blanc, M.O. Bettler, G. Conti, V. Fave, R. Frei, N. Gueissaz, A. Gong, G. Haefeli, J. Luisier, R. Muresan, T. Nakada, M. Needham, L. Nicolas, M. Knecht, A. Perrin, C. Potterat, O. Schneider, M. Tran are with the LPHE, EPFL Lausanne, 1015 Lausanne, Switzerland.

C. Bauer, M. Britsch, W. Hofmann, F. Maciuc, M. Schmelling, H. Voss are with the Max Planck Institut für Kernphysik, 69117 Heidelberg, Germany.

A. Büchler, U. Straumann, J. Anderson, N. Chiapolini, V. Hangartner, C. Salzmann, S. Steiner, O. Steinkamp, J. van Tilburg, M. Tobin, A. Vollhardt are with the Physik-Institut der Universität Zürich, 8057 Zürich, Switzerland.

A. Gong is with the Dept. of Engineering Physics, Tsinghua University, Beijing, China.

V. Iakovenko, O. Okhrimenko, V. Pugatch are with the National Academy of Sciences, Institute for Nuclear Research, 03680 Kiev, Ukraine.

System (ECS). The ECS is a highly hierarchical, distributed and heterogeneous expert system that controls the detector hardware.

Finally, the DSS is a separate reliable and fast real-time response system whose which protects the equipment and prevents situations leading to severe alarms.

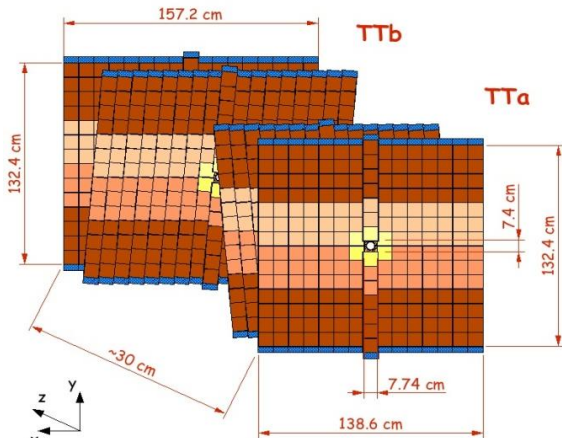


Fig. 1. The TT station.

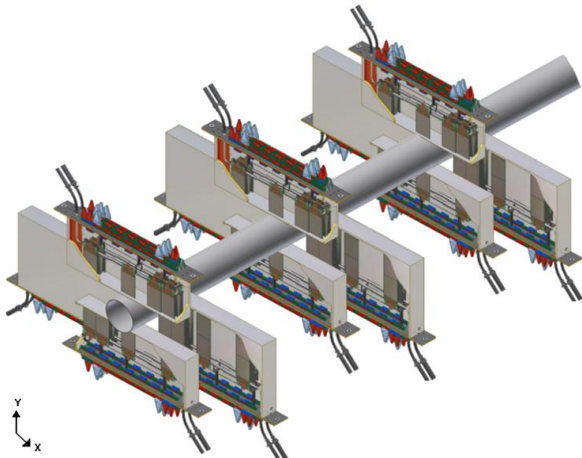


Fig. 2. The three IT tracking stations.

II. THE ST ELECTRONICS OVERVIEW

The DAQ read-out electronics is made up of three building blocks (Fig. 3):

1. The L0 FE electronics, the Beetle.
2. The Service Boxes.
3. The TELL1 processing boards.

The silicon sensors are read-out by the Beetle chip. This is a radiation hard device placed on the read-out hybrid, bonded to the silicon sensors. The chip performs a fast amplification and shaping of the input signals and stores them in an analogue pipeline which will be read-out with a fixed latency of $4\mu\text{s}$. The analogue output data consists of 128 channels plus 16 headers which are digitized in the Service Box. The data is transmitted in serial mode over four different ports. The Service Box which houses the FE digitizing and control electronics is located close to the detector and hence rated for radiation tolerance ($\sim\text{Krad}$ s). The Digitizer Boards (DB) perform an 8-bit digitization of the Beetle data. The digitized data of each port is then serialized in the DB by the CERN

GOL (Gigabit Optical Link) and transmitted optically to the TELL1. In addition, a Control Board (CB) [6] is housed in each Service Box. The CB carries out the slow and fast control of the FE electronics. It implements the SPECS [7] slow control interface and also drives the Timing and Trigger Control signals coming from the TFC system [8] to the FE hybrids and DBs.

The last of the blocks is the TELL1 real-time processing board. The data from the Digitizer Boards are deserialized and further processed in the TELL1. The TELL1 electronics is placed behind a shielding wall, 60 m from the interaction point to minimize the amount of electronics prone to the radiation effects.

To preserve synchronization over large distances LHCb has a global clock and fast signal transmission network, the Timing and Fast Control (TFC). The TFC is driven by the Readout Supervisor (RS) [9], responsible for distributing the LHC clock and scheduling the trigger decisions.

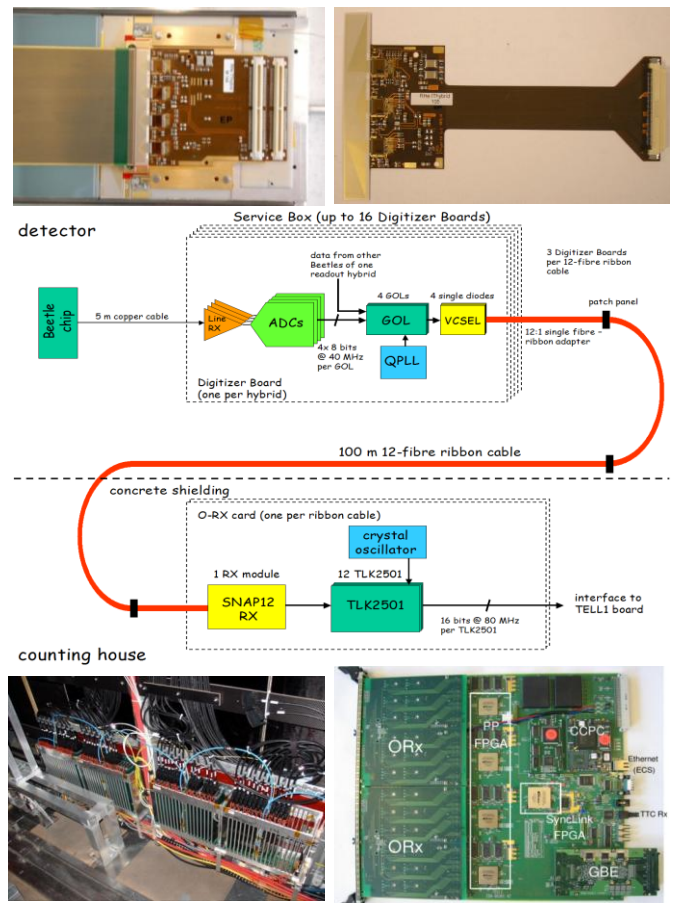


Fig. 3. The top pictures from left to right are the TT and IT hybrids housing the Beetles. The middle picture is the scheme of the read-out link. The bottom left picture shows the four service boxes corresponding to an IT half-station and the right one is the TELL1 board.

III. THE TELL1 PROCESSING STAGE

The TELL1s are custom developed boards for being used by different detectors in LHCb, each with its own data processing algorithms. The ST is one of the LHCb most demanding detectors in terms of data processing. The main

processing units are five large FPGAs which can handle ~ 60 Goperations/second per Tell1 board.

Fig. 4 shows a simplified block diagram of the ST TELL1s. Two optical receiver mezzanine cards provide an input for 24 optical links from the service boxes. Each optical link corresponds to the digitized data from one Beetle. Each of the “PP-FPGAs” processes the data from 6 Beetles at a maximal event rate of ~ 1.1 MHz. After this stage the “Synch-Link” FPGA receives the fast signalling from the TFC system via “TTCrx” and distributes the synchronization signals (clock, trigger and event identification) to the PP-FPGAs. It is also in charge of collecting and merging the data fragments from the “PP-FPGAs”, to assemble multiple events into the so-called multi-event packets (MEPs), and to perform Ethernet and IP framing before transmission to the read-out network. The read-out network is interfaced by a four-port Gigabit Ethernet card (GBE) delivering 4 Gbit/s data bandwidth. The fast control signals received via “TTCrx” chip are the LHC clock (locked at 40.079 MHz), the reset, the L0 trigger and the data IP destination address.

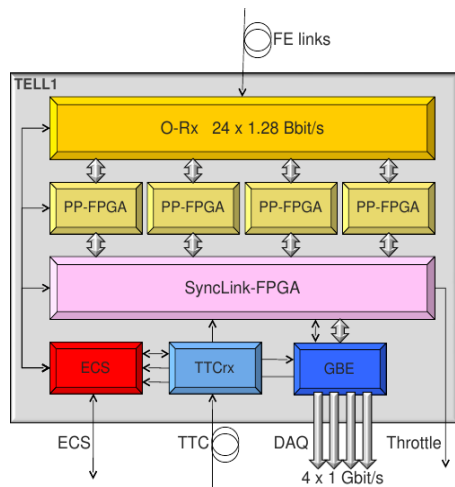


Fig. 4. TELL1’s simplified block diagram

In addition to the zero suppressed (ZS) data, the TELL1 can send the raw data to the network. The maximum achievable data throughput is some tenths of KHz. This feature is essential for testing the TELL1 algorithms as explained later.

A. Synchronization

The first step is to synchronize the data arriving from the FE links to a common clock domain as the different links have different arrival times. In addition, the incoming data are verified to be from the same event by checking the Pipeline Column Number (PCN) time stamp sent in the Beetle header. Afterwards the data are tagged with the experiment wide event ID. Finally the ST specific data headers are created (error flags, error counters, status flags...). The links synchronization relies on the fact that the incoming FE event data size is fixed (35 clock cycles at 40 MHz). Due to optical transmission failures, caused by low power VCSEL laser transmitters in the Digitizer Board, it was necessary to improve the performance of the data synchronization by using a simple but robust

algorithm. This permits filtering spurious glitches in the “Data Valid” (DV) signal from the deserializers (Fig. 5):

1. Two counters count the high and low period of the DV signal (DV is in 80 Mhz domain). Once the DV turns high, the low counter is cleared to zero. When the DV is low, the low counter increases; only after the counter is bigger than 4, is the high counter cleared. With this, a high glitch is removed after 4 cycles, and the one cycle low gap between consecutive events is not affected.
2. An event is only valid after high counter is bigger than 65 and if the high counter is bigger than 70, a forced end of event is generated and the high counter is cleared to zero.

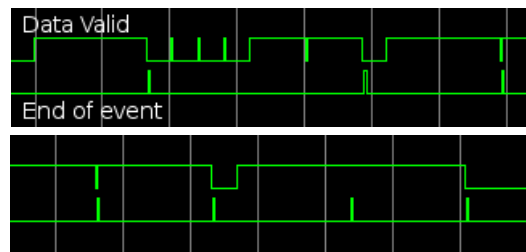


Fig. 5. TELL1’s input synchronization simulation of “Data Valid” and “End of event” signals. The simulation shows a normal event, few noise glitches, a “split” event, few consecutive events and a conglutinated event are simulated.

B. Pedestals

After synchronization pedestal subtraction is performed. The pedestal is the mean value over some period of time of the amplitude registered by one strip with no physical signal (the average noise level). The pedestals may vary with time so an automatic update procedure during data-taking has been implemented. They can be approximated by averaging over a large number of events assuming low detector occupancy. For each event, the difference between the actual value and the pedestal value is computed for subsequent calculations.

C. Linear Common Mode Suppression (LCMS) algorithm

After pedestal correction, the ADC count of the first of each 32 read-out strips is also corrected for header cross-talk, as it is adjacent to a header strip with a value much deviant to the read-out strips. Here, linear common mode noise suppression is performed per set of 32 read-out strips, as is shown in the schematic (Fig. 6). After the noise reduction steps the ADC counts above a certain hit threshold are then formed into clusters, which are stored for further analysis.

D. Derandomizer

Derandomization after CM suppression is needed because the subsequent steps of clusterization and zero suppression need a processing time larger than the minimum spacing between L0 accepted events. The buffer size is dimensioned for 64 events.

E. Zero-suppression and clusterization

Data compression is applied to the CM corrected data using detector specific hit finders and clusterization procedures. In a first step, the seeding strips are found using strip individual

Signal/Noise thresholds and neighboring hits are combined into a cluster. The chosen data format allows encoding up to four strips in a cluster. Larger clusters are split. In a last step, the cluster position is calculated by a weighted average with 2 bit inter-strip precision.

After zero suppression the event size is variable. Derandomizing buffers are employed to average the data rate and the data processing time. To prevent overflows, each buffer can generate a throttle signal acting on the TFC system.

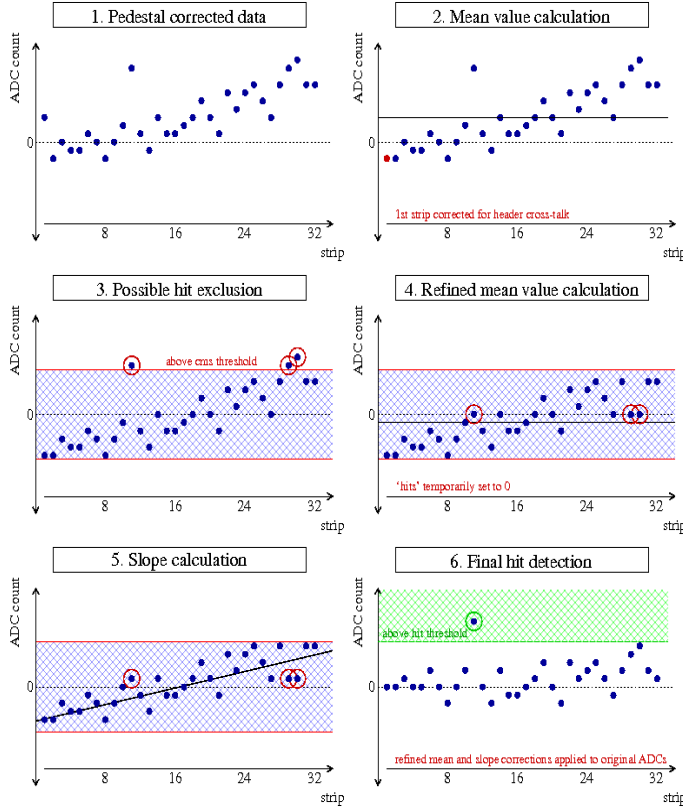


Fig. 6. LCMS algorithm.

F. Linking and data transmission

Finally the data from the different processor channels are linked and then encapsulated. To cope with varying event size at the output to the Ethernet network, an external 2 MB large derandomizer buffer is attached to the SyncLink- FPGA. To reduce the Ethernet overhead, the event data are assembled in MEPs prior to transmission to the network.

G. The TELL1 emulator

As rule of thumb high percent of errors in digital real-time processing are caused by programming mistakes. Similarly to any software development VHDL implementations are also susceptible to errors. Due to the overall complexity of the algorithms implemented in the TELL1 detailed VHDL simulations are difficult to perform. As only the zero suppressed (or clusterized) output data will be kept and used for further analysis it is necessary to verify the functionality of the FPGAs. Thus, a mechanism to test if the FPGAs are doing what it is expected is needed. For this reason, a TELL1 emulator performing the same algorithms has been developed.

Using noise raw data, bit-perfect emulation of the FPGA algorithms has been achieved after several iterations, which has been invaluable for the validation of the detector's read-out prior to the first LHC collisions. This tool has not only helped to find bugs in the TELL1 firmware but it's also an essential instrument to pre-test the tuning of the strip thresholds before using them in the TELL1 algorithms (Fig. 7).

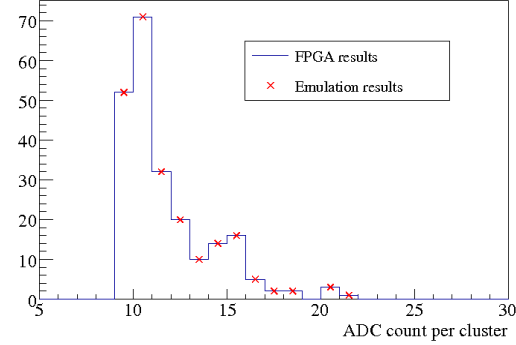


Fig. 7. Emulator vs. TELL1 results. The figure shows perfect matching between the emulator and the TELL1 when looking for clusters.

The emulator will also be run online in the monitoring computer farms. During the experiment data taking not only zero suppressed data will be transmitted by the TELL1 but also regularly scheduled raw data banks. These raw data will be used by the emulator to verify online the correct working of the detector and the algorithms. Together with the event data, a copy of the pedestal is sent in order to exactly replicate the TELL1 computations. To verify the proper working of the pedestal update specific runs are performed prior to data-taking.

H. ST TELL1 thresholds tuning and operation

Each TELL1 processes the data of 3072 FE channels and the ST uses a total of 90 TELL1. A total of 2128 out of 2160 optical links are used (~98%). Running the whole detector synchronously maximizing the efficiency of recorded data and minimizing zero-suppression overflows is a challenging task.

The TELL1s have to process all the FE data with minimal data loss and limited latency. The data is further transmitted to the high level event building software which assembles the event from the data received from all sub-systems. Failure to receive in time the data from any TELL1 would produce incomplete events with the consequent fatal degradation of the experiment physics performance. A second order consequence is that event building latency increases keeping the limited computational and network resources busy for longer. The processing of the TELL1 data is a time critical issue on which relies the correct functioning of the DAQ network.

There are about 13K parameters (basically thresholds) to configure per TELL1 which have different impact in the TELL1 output data rate. The thresholds to set are:

1. Pedestal Mask (per channel): mask off dead channels.
2. Pedestal value (per channel): not relevant as the pedestals are automatically updated.
3. CMS Threshold (per channel): step 3 of Fig.6.

4. Hit threshold (per channel): for hit detection (Fig. 6).
5. Cluster sum threshold (per Beetle): the sum of the total deposited charge of maximum 4 channels.
6. Spill-over sum threshold (per Beetle): to check if sum of the cluster charge exceeds the spill over threshold.
7. Header correction (4 values per board): step 2 of Fig. 6.
8. High threshold (per TELL1 board): high threshold for the Beetle headers.
9. Low threshold (per TELL1 board): low threshold for Beetle headers.

The number of thresholds per board amounts to 12342, giving a total of 5,924,736 for the TT and 5,184,144 for the IT. The aim pursued during threshold tuning was to reduce the rate of noise clusters to at least 10^{-4} with minimal efficiency loss. A noise rate of $\sim 4 \cdot 10^{-5}$ was achieved setting individual strip thresholds. The set thresholds were calculated in a S/N basis. A value of S/N of 3 was used for the hit threshold, 4 for the cluster sum, 5 for the CMS and 10 for the high and low threshold. The 3 first are the settings that determine the data quality and output data rate. During normal running at nominal luminosity the estimated event data size is of ~ 3.1 Kb (TT) and ~ 2.7 Kb (IT) per TELL1. At the nominal rate of 1.1 MHz the total throughput for the ST detector is be ~ 6.38 GB/s.

IV. THE ECS

The ECS is capable of controlling and configuring the detector electronics infrastructure in different running modes, monitors all the relevant environmental parameters and is able to perform automatic corrective actions ensuring the safe running of the detector. The CB provides slow control (I2C interfaces, PT1000 temperature readings...) and fast control signaling for the ST front end (FE) electronics using the SPECS field-bus system. Other LHCb specific hardware interfaces such as the CCPC [10] are used for remote control and monitoring, this is the case for the TELL1. In the middleware layer specific software interfaces such as DIM [11] are used as well as industry standards such as OPC.

The ST ECS software is based on the hierarchical Finite State Machines concept. It has been implemented using the SMI++ toolkit [12] and the PVSS SCADA [13]. The controls hierarchy is structured in the first layer following a functional basis and in the underlying layer in a geographical division manner. This allows a similar layout for the IT and TT detectors.

A. Hardware

All the ST hardware is under the scope of the ECS. The number of devices to be configured and monitored is several thousands. A careful and planned design of the FE hardware was performed so a closely similar control system could be developed for IT and TT. The electronics design has been driven by a well defined hardware partitioning and by the development of a common Control Board for IT and TT service boxes. Table I shows the quantity of devices that comprise the ST and need to be configured, whereas table II shows the number of parameters that need to be monitored.

Standard industry solutions were adopted for the high voltage (CAEN) and low voltage (Wiener MARATON) power supplies as well as for the cooling station. The CAEN HV and MARATON LV power supplies use TCP/IP over Ethernet to connect to the ECS. The cooling station is controlled using MODBUS.

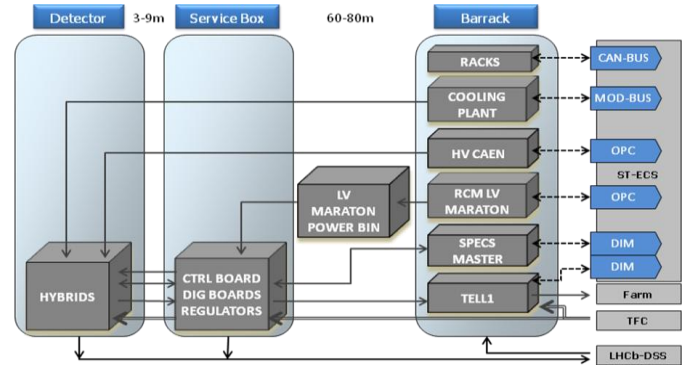


Fig. 8. Silicon Tracker equipment scheme

	IT	TT		IT	TT
Beetle	1008	1120	PT1000	373	328
GOL	1008	1120	HMX2000 ¹	12	12
SPECS	48	48	DAQ-FE status	336	280
DCU	408	316	LVR voltage	336	280
Delay25	24	24	LVR INH	420	376
TTCrx	24	24	LVR OCM	1008	840
LVR	1080	852	MARATON V	48	48
CCPC-TELL1	42	48	MARATON I	48	48
CAEN ch	84	152	CAEN V	84	152
MARATON ch	48	48	CAEN I	84	152
Cooling station	1	1	CCPC-TELL1	420	480

Table I: devices to be configured. Beetle chips, CERN GOL serializer, DCU ADC, CERN Delay25 chip, CERN TTCrx chip, low voltage regulators (LVR) inhibit, CCPC-TELL1s, CAEN and MARATON channels and the cooling station.

Table II: Parameters monitored. Temperatures, humidity, FE PLLs status, LVR (low voltage regulators) voltage, inhibit and over-current signals, MARATON voltages and currents, CAEN voltages and current and CCPC-TELL1 registers status.

Custom hardware was developed in the ST to interface the FE and back-end equipment. For the FE electronics radiation and magnetic field hardware had to be built. The FE devices, i.e., hybrids, digitizer boards and service box low voltage regulators (LVR), are interfaced through the CB, which houses two SPECS mezzanines. The SPECS system is one of the LHCb adopted hardware interfaces to connect the FE hardware with the ECS. Whilst the SPECS is intended to interface the FE, the CCPC is used for the TELL1 back end electronics. The CCPC is an on-board credit card-sized and disk-less PC running a Linux kernel attached over the network to a boot server. The racks and crates in the barrack, where the DAQ and detector electronics is housed, use the CANBus and CANOpen standards and they are based in the ELMB system [14]. The scheme of Fig. 8 summarizes the overall architecture

¹ HMX2000 is a MEMS humidity sensor from Hygrometrix Inc.

of the Silicon Tracker equipment. The TFC system provides real-time timing and trigger signaling to the DAQ electronics. A way to publish the DSS information to the sub-detectors is foreseen so the control system can react.

A partitioning for the TELL1 DAQ links was defined early in the detector design [15]. This grouping was extended to the LV, HV, FE and control electronics during the hardware design and development. As a consequence the partitioning of the devices matches in all domains and a coherent control of the system is feasible.

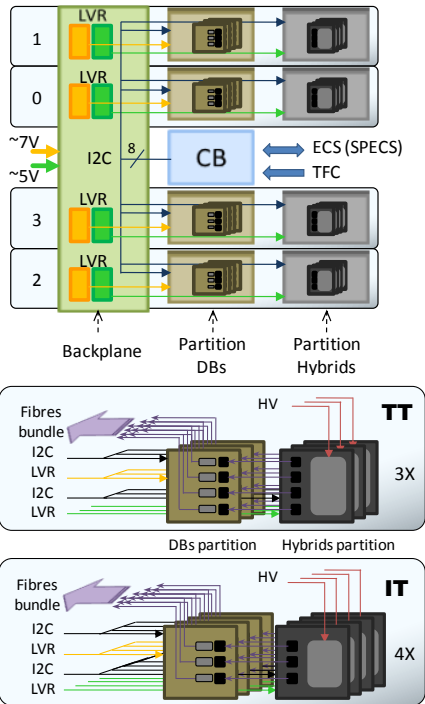


Fig. 9. IT and TT FE partitioning. The partitioning of the service box FE elements into four groups is shown in the upper picture. The bottom picture shows the IT and TT partitions, made up of three and four boards respectively.

B. Middleware

OPC and DIM are the most popular middleware interfaces used in LHCb.

DIM (Distributed Information Management) [11] is a high level client/server communication protocol for distributed and mixed environments that provides a network transparent inter-process communication layer. A DNS (DIM Name Server) provides transparency (i.e. a client does not need to know where a server is running). Servers “publish” their services by registering them with the name server. The DIM protocol is used in LHCb for the remote control of the CCPC (DIM *ccserv* server) and SPECS (*SpecsServer*) devices as depicted in figures 10 and 11.

OPC (OLE for Process Control) is an open standard that specifies the communication of real-time process data, alarms and events, historical data, etc. It is used in the ST for interfacing the CAEN and Wiener MARATON power supplies. The ELMB system also provides an OPC server on top of CANOpen.

DIP is essentially a robust information “one-way”

distribution service based on the publish/subscribe paradigm that supports an on-change data exchange. It allows small amounts of real-time data to be exchanged between systems that don’t need very low latency.

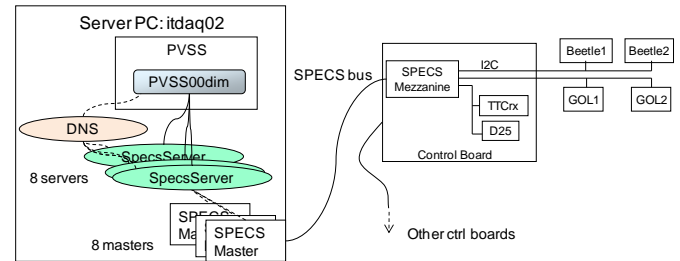


Fig. 10. Deployment of the DIM SpecsServers (IT). One server per SPECS master running the control computer giving a total of 4 for IT and 4 for TT.

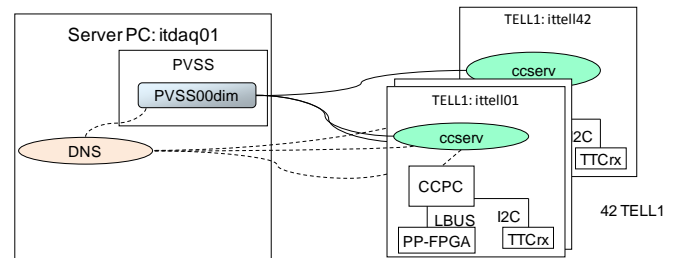


Fig. 11. Deployment of the CCPC DIM servers for the TELL1s (IT). One server per TELL1 running in the CCPC board.

C. Control software

The control software consists of a large complex system that configures and monitors in (soft) real-time the ST equipment. The supervision layer is provided by the PVSS SCADA [13] and the control layer by the SMI++ FSM toolkit. Some other tools have also been built on top of PVSS in order to provide extended functionalities.

PVSS has been chosen by CERN to be the SCADA system for all LHC experiments. Besides the typical SCADA functionality (data acquisition, logging, alarm handling...), PVSS can run in a highly distributed manner, it has multi-platform support (WNT and Linux), it is device oriented and it has an advanced scripting facility. It contains an internal “real-time” database, based on the object oriented *datapoint* concept. The *datapoint* is the basic data container of a variable. Finally, PVSS is capable of running scripts, which are triggered by an update of a *datapoint*.

The control system is built as a hierarchy of sub-systems distributed over several computers. The used toolkit (SMI++) which combines finite state machines (FSM) and rule-based programming, allows for the description of the various sub-systems as decentralized deciding entities, reacting in real-time to changes in the system, thus providing for the automation of standard procedures and for the automatic recovery from error conditions in a hierarchical fashion. Modelling systems with FSMs allows simple and accurate design of sequential logic and control functions, as long as the system can only be in a limited (finite) number of states. They permit sequencing commands and automating behaviours. They also reduce the number of parameters exported and managed in the SCADA supervisory layer.

The adopted tree-like hierarchy represents the structure of the sub-detector and hardware components. This tree is composed of two types of nodes: “Device Units” (DU) which are capable of “driving” the equipment to which they correspond and “Control Units” (CU) which correspond to sub-systems and can monitor and control the sub-tree below them. A similar layout has been developed for the IT and TT hierarchies, despite of the different detector geometries.

The control hierarchy allows a high degree of independence between components, for concurrent use during integration, test or calibration phases, but it also allows integrated control, both automated and user-driven, during physics data-taking. The system can either be run as an autonomous system or can be driven by the top LHCb overall ECS. The control tree is first divided into four standard domains that every LHCb sub-detector has to implement, namely DAQ, DCS, DAI and HV. This way, the top LHCb control node can easily take control of the detector via these standard domains.

While one and only one hierarchical control tree is meant to take the control of a detector it may be useful to have other views of some parts of the sub detector components. The FSM tool provides the possibility to link nodes already part of the main control tree under an alternate control tree providing a root tree which is not part of the main control tree. This alternate control view of a set of components based upon a different segmentation of the detector gives a way to control a sub part of the detector as an autonomous sub detector with all the necessary components to run stand alone. An application of this feature is the possibility to take data with only a sector of a detector. Another use case is the automatic actions performed by separate nodes to ensure the safe running of the detector.

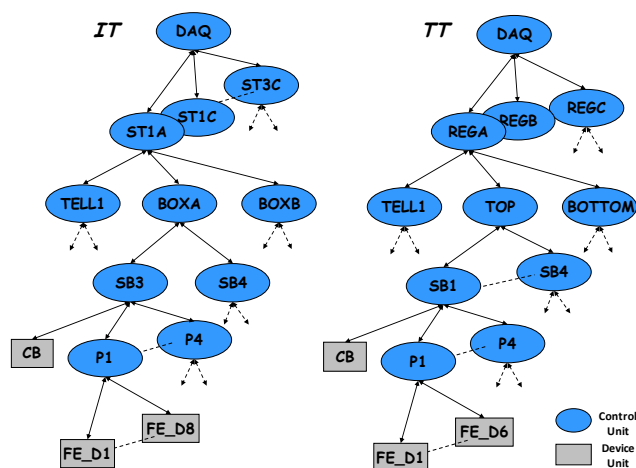


Fig. 12. IT(left) and TT(right) simplified DAQ domain hierarchies.

All the monitored parameters need to actively inform the operator about any value out of the acceptable working range. For that purpose PVSS offers functionalities for real-time data handling, alert messaging and a standard PVSS alarm panel. In addition, the monitored data is stored in the archiving database.

D. The databases

The storage of data is a built-in element of any control system. Different types of data related to the control process need to be stored: configuration data of the devices, samples of the measurements, etc. Three databases for different purposes are used in the LHCb experiment:

1. The archiving database [16].
2. The configuration Database [17].
3. The conditions database [18].

The archiving database contains all the monitoring data gathered from the devices as listed in table II (voltage readings, temperatures, etc.). Besides the parameters in the table other values such as the ones that provide the DUs status, are stored to keep track of them over time. In order to reduce the volume of data that is sent to the database a smoothing is applied to the archived data. For the “analog” readings a value dependent smoothing (a percentage basis to the relative value) is used, and for the integer values just old/new comparison filtering is applied.

The configuration database contains all data needed to configure the hardware software for the different running modes (i.e. high voltage settings, pedestal settings, trigger settings, etc.). HEP experiments need to operate in running modes other than the normal physics data taking. Detectors need to run stand-alone in order to perform calibration or testing activities in order to fix any failure, to improve its performance or just to understand its behavior under certain circumstances. At other times, when running under the LHCb global control, it may be necessary to take special runs for detectors integration, alignment or even to perform “cosmics” data taking. In addition, the behavior of the hardware can vary due to the aging effect induced by the radiation and new settings may be needed from time to time. Some flexibility and re-configurability of the devices is hence mandatory. Different hardware settings must be loaded from a configuration database depending on the running mode (“physics”, “cosmics”, “calibration”, etc). The parameters downloading from a configuration database is synchronized in order to allow for the coherent operation of the experiment for a given run type[19]. This configuration data is grouped into recipes. A recipe is a collection of settings (values or/and alert parameters) corresponding to a number of *datapoints* in PVSS and identified by a unique name.

The conditions database contains time varying data that might be necessary for later processing, e.g., the reconstruction of the events. These data can be subset of the monitoring data read from the hardware (e.g. high voltage readings if changed by more than “n” volts) or some of the hardware configuration settings that have been used (e.g. FE settings used by a particular run).

E. Interaction with LHCb

The four common domains are the entry points for the LHCb overall ECS to control the sub-detectors (Fig. 13). For stand-alone operation the sub-detectors need some resources that are not part of the ST infrastructure itself but LHCb wide.

An online partition with all the necessary resources for performing full detector activities needs to be allocated, that is, a TFC partition for driving the fast control signaling as well as a slice of resources to collect the data, store it and display online results. These resources are dynamically assigned to the detector when a stand-alone run is performed. The way this is implemented is depicted in the figure 13: a parallel “IT” node takes control of the four top entry domains (DAQ, DCS, DAI and HV). This node implements all the functionalities in stand-alone mode.

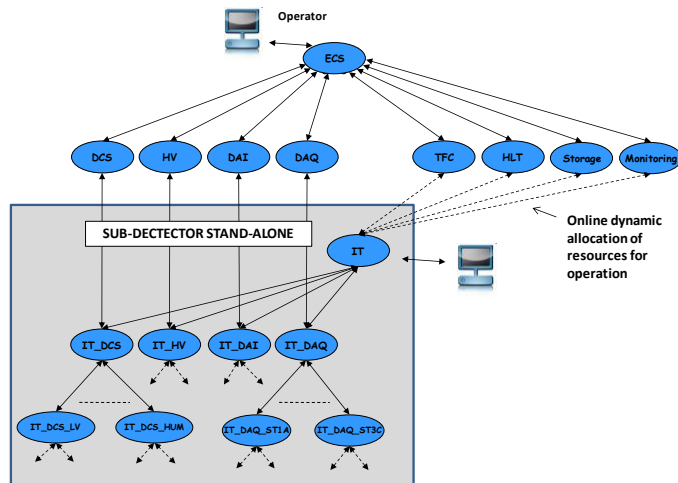


Fig. 13. IT interconnection with LHCb.

The ECS allows automatic scans on hardware parameters. An example consists of pulsing the detector channels with different delay times of the calibration signal (Fig. 14).

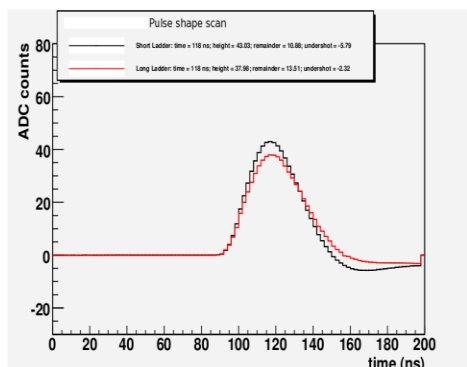


Fig. 14. Beetle signal output shape scanning through the tes-pulse signal delay. The red shape is for two silicon sensor detectors and the black for single sensors.

In order to ease the use of the system for shift operators the ECS provides intuitive User Interfaces (UI) and exhibits a common look and feel.

V. THE DSS SYSTEM

The Detector Safety System (DSS) is a parallel hardware oriented system which will take the appropriate action to protect the hardware against potential causes of damage, usually by powering off the equipment. It's a redundant system that ensures a real time response in case the ECS failed.

The DSS checks a few critical parameters with dedicated

sensors: temperature, cooling status, gas status and cooling water flow switches. The logic output of all these sensors is routed to a Programmable Logic Controller (PLC), which runs a closed-loop control cycle. It exchanges information with the ECS to inform about the status and actions taken.

VI. CONCLUSIONS

The online DAQ and ECS systems for the ST have been developed and deployed at the LHCb experiment. Both systems were tested in stand-alone configuration and included into the LHCb hierarchy. The detector has been successfully run during the first injection tests of the LHC machine and first data collected. During the system commissioning and detector running many lessons were learned and errors found that provided feedback to improve the detector performance.

REFERENCES

- [1] “LHCb Reoptimized Detector Design and Performance”, Technical Design Report. CERN/LHCC-2003-030. ISBN 92-9083-209-6.
- [2] H.Voss, “The LHCb Silicon Tracker”, Nucl. Instr. and Meth. A 549 (2005) 44-48.
- [3] N. Bakel et al., “The Beetle Reference Manual”, LHCb note 2001-046.
- [4] Vollhardt et al., Proc. LECC 2005, CERN/LHCC-2005-038 (2005) 187-191.
- [5] G. Haefeli, et al., “TELLI Specification for a common read out board for LHCb”, IPHE note 2003-002.
- [6] D. Esperante, A. Vollhardt, “Design and development of the Control Board for the LHCb Silicon Tracker”, LHCb note 2007-153.
- [7] D. Breton, D. Charlet, P. Robbe, I. Videau, “SPECS : A serial protocol for experiment control system in LHCb”, 10th ICALEPCS, Geneva (2005).
- [8] Z. Guzik, R. Jacobsson, B. Jost, “Driving the LHCb front-end read-out”, IEEE-TNS vol. 51, n° 53, June 2004.
- [9] R. Jacobsson, B. Jost, Z. Guzik, “Readout supervisor design specifications”, LHCb note 2001-012.
- [10] F. Fontanelli, G. Mini, M. Sannino, Z. Guzik, R. Jacobsson, B. Jost, and N. Neufeld, “Embedded Controllers for Local Board-Control”, IEEE-TNS, vol. 53, n°3, June 2006.
- [11] C. Gaspar, M. Donszelmann, “DIM – A Distributed Information Management System for the DELPHI Experiment at CERN”, 8th NPSS Real Time Conference, Vancouver, 1993.
- [12] B. Franek and C. Gaspar, “SMI++ Object-Oriented Framework for Designing and Implementing Distributed Control Systems”, IEEE-TNS, vol. 52, n°4, August 2005.
- [13] PVSS II von ETM, <http://www.pvss.com>
- [14] B.Hallgren et al., “The Embedded Local Monitor Board (ELMB) in the LHC Front-end I/O Control System”, Proc. LECC 2001, CERN/LHCC-2001-005 (2001) 325-330.
- [15] M. Needham and O. Steinkamp, “Updated channel numbering and read-out partitioning for the Silicon Tracker”, LHCb note 2007-137.
- [16] M. Gonzalez Berges, “The high-performance database archiver for the LHC experiments”, 11th ICALEPCS Knoxville (2007).
- [17] L. Abadie, C. Gaspar, E. van Herwijnen, R. Jacobsson, B. Jost, N. Neufeld, “The LHCb configuration database”, 10th ICALEPCS, Geneva (2005).
- [18] M. Clemencic, N. Gilardi, J. Palacios, “LHCb Conditions Database”, LHCb note 2006-017.
- [19] F. Calheiros, P. Golonka, F. Varela, “Automating the configuration of the Control Systems for the LHC experiments”, 11th ICALEPCS, Knoxville (2007).