LHC Collimators Low Level Control System

Alessandro Masi and Roberto Losito

***Abstract*—The low level control system (LLCS) of the LHC collimators is responsible for accurate synchronization of ~500 axes of motion at microsecond level. Stepping motors are used in open loop ensuring a high level of repeatability of the position. In addition, a position survey system based on Resolver and LVDT sensors and operating at approximately 100 Hz, verifies in real-time the position of each axis with some tens of micrometers accuracy with respect to the expected position. The LLCS is characterized by several challenging requirements such as high reliability, redundancy, strict timing constraints and compactness of the low level hardware because of the limited space available in the racks underground. The National Instruments PXI platform has been proposed and evaluated as real-time low level hardware. In this paper the architecture of the LHC collimators LLCS is presented. The solution adopted for implementing motion control and positioning sensors reading on the PXI platform are detailed.**

***Index Terms*—LHC collimators, LVDT, PXI, real-time control.**

I. Introduction

THE Large Hadron Collider (LHC) is the next particle accelerator under construction on the CERN site [1]. The LHC will mainly accelerate and collide 7 TeV proton beams but also heavy ions such as lead. It is undergoing installation approximately 100 m underground, in the 27 km circumference tunnel, that previously housed the Large Electron Positron Collider (LEP). The LHC design is based on superconducting twin-aperture magnets which operate in a superfluid helium bath at 1.9 K.

In the LHC an unprecedented energy (~ 350 MJ) will be stored in the colliding beams. Safety considerations are therefore critical for the operation of the machine. At full energy, a mis-steered beam can in fact inflict considerable damage on the installation.

More than 100 collimators will therefore be installed, primarily to protect the machine from uncontrolled particle losses, but also to absorb energetic particles travelling outside of the nominal beam core and to reduce noise for the LHC experiments.

The LHC collimators consist of two 1-meter long blocks (jaws) of different materials (graphite, copper, tungsten) that have to be positioned precisely with respect to the beams circulating in the collider [2].

The required accuracy of positioning is 20 μm, one tenth of the beam size, i.e., the rms of the beam particles distribution at nominal energy.

Also required is the possibility to position any jaw within a well defined angle with respect to the nominal beam trajectory.

Manuscript received May 10, 2007; revised December 3, 2007.

The authors are with CERN, Geneva 1211, Switzerland (e-mail: alessandro.masi@cern.ch).

Digital Object Identifier 10.1109/TNS.2007.914206

To satisfy this requirement, each jaw can be moved on both ends by stepping motors. In order avoid the excitation of dangerous mechanical vibrations on the long block, especially in the case of graphite, it is necessary to perfectly synchronize the two motors on each jaw to better than 1 ms.

Jaws in different collimators need to respect movement functions sent by a central supervisory application which is not part of the LLCS. Each jaw must follow its given function within the already stated tolerance of 20 μm. With the nominal motor speed of 400 steps per second, and a reduction factor of 2 mm/revolution, this means that any motor (more than 500 overall) in the LLCS has to be synchronized within 10 ms all along the operation of the LHC.

Each motor is equipped with a resolver to detect lost steps in order to check the consistency of the actual movement with the desired movement function. An LVDT (Linear Variable Differential Transformer) is installed on each axis to measure the absolute position of each jaw extremity. The LVDTs also enable measurement of the distance between two jaws within the same collimator. Overall, the LLCS will have to control nearly 500 resolvers and more than 700 LVDTs.

A further difficulty is posed by the environmental conditions of the LHC. Due to the high level of radiation expected in the proximity of the collimators (several MGrays/year), no electronics can be embedded in the sensors or in the motors.

Furthermore, radiation safe alcoves for the installation of the driving electronics can be up to 800 m far from the collimator itself, generating a complex matching problem for conditioning electronics. Over such long distances the impedance of the cable may in fact transform the impedance of the sensor from inductive to capacitive. Standard off-the-shelf conditioners do not generally provide a sufficient tuning range to preserve a satisfactory stability and accuracy of positioning for cable distances from 20 meters up to 800 meters as required for the LHC collimators.

For early detection of any lost step the resolver reading must be fast and synchronous to the movement. LVDT readings are required to be fast in order to minimize the risk of undetected errors in jaw positioning.

Due to the large number of motors and sensors installed, a further requirement is the ability to remotely check the integrity of motors and sensors, and to perform an automated remote calibration of the system.

In conclusion, the requirements in terms of the real-time behavior of the LLCS are:

1) synchronization of motors within the same collimator to better than 1 ms;
2) synchronization of motors of different collimators to better than 10 ms;
3) an accuracy of positioning and reading of 20 μm in any condition and at any temperature;

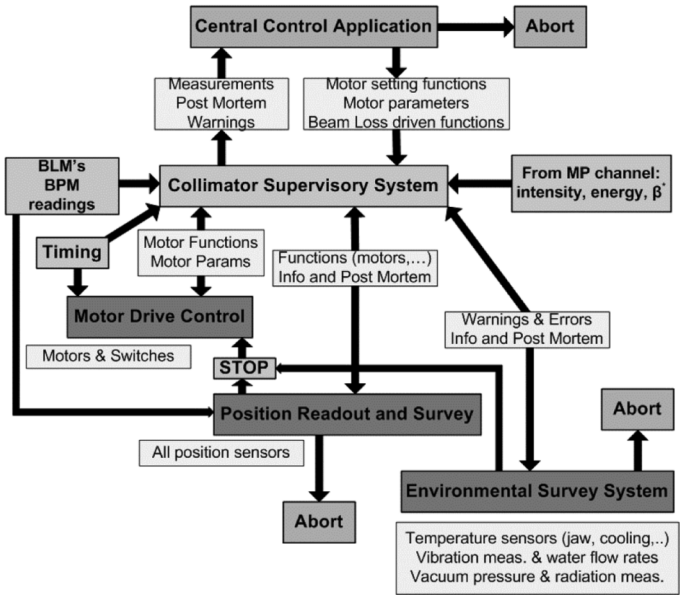

Fig. 1. The layout of the Collimators Control System.

4) frequency of monitoring of the resolvers: up to 400 Hz;
5) frequency of monitoring of the LVDTs: 100 Hz.

The solutions adopted are detailed in Sections III–V, with special focus on the real-time aspects of the solution.

II. The Architecture of the Collimators' Control System

Due to the large number of collimators to be installed in the LHC machine, and the strict accuracy, speed and accessibility requirements of the overall system, it was preferable to separate the different functionalities on different levels (low level, supervisory level and application level) [3].

Additionally, the high level of reliability required for the control system to efficiently protect the elements of the LHC necessitates separating the function of motion steering from the function of motion survey and interlocking on different physical machines.

A layout of the collimators' control system is sketched in Fig. 1. The low level functions are divided in three modules:

- **Motor Drive Control (MDC)** is the module which receives the commands from the supervisory system about the movement to be executed by the collimator, verifies the consistency of the order with the specificity of the particular collimator it is controlling, and performs the movement. It provides interlocks to eventually abort the beam in the LHC in case a problem occurs during the movement (e.g., unexpected status of one of the end switches). It governs all the functions of the movement (motors, end switches), and verifies that the motors do not lose steps (by reading the resolvers).
- **Position Readout & Survey (PRS)** is the module responsible for checking that the actual position corresponds to the desired one within a well defined tolerance, depending on the beam energy and the jaw position, received by the supervisory system. The position is read on LVDTs with an accuracy of better than 20 μm and at rate of up to 100 Hz. In case of problems, it aborts the beam.
- **Environmental Survey System (ESS)** reads all environmental parameters (temperatures, vacuum gauges etc.,…) and can trigger a beam abort within specific conditions. Since it has no specific real-time features, this system will not be described in details in Sections III–V.

The coordination and synchronization among all the collimators in the machine is divided into two levels. All the time critical actions are performed by an application called **Collimator Supervisory System (CSS)** running on several Linux PCs (referred to as "Gateway") installed closed to the low level racks. Every Gateway will manage about 30 collimators through data and commands sent to the low level applications via the CERN middleware framework FESA (Front End Software Architecture) [4]. All time critical actions will be triggered on the low level system by pulses sent through dedicated optical fiber links between the Gateway and the different low level systems of the collimators supervised by the CSS. The different Gateways will be synchronized among them via the standard CERN timing system [5]. Every Gateway will have a timing receiver installed in it, which provides not only the synchronization signals, but also information about energy of the circulating beams, beam type etc The CSS will also have a fast link with the low level system of the beam loss monitors installed close to the collimators. It is foreseen in fact, following the successful experience of the collimation system of the Tevatron at Fermilab [6], to implement automated beam-loss based alignment to speed up the setup of the collimation system before ramping the energy to 7 TeV.

The interface to the operators in the control room constitutes the last module of the system and is called the **Central Control Application (CCA)** [7]. The role of the CCA is not only to create an efficient graphical environment to allow control of the ~ 120 collimators in one screen, but also to provide special functions such as trimming, which will be developed on the LSA framework presently under development at CERN [8].

Section III will analyze in details the requirements, design, implementation and results of tests of the first two low level modules, the MDC and the PRS.

III. The Choice of the Real-Time Platform and the Role of the Middleware in the Low-Level System

The choice of the real-time platform to be used to implement all the functionalities of the low level system was driven not only by technical requirements and cost, but also by the need for compatibility with the standard middleware, Common Middleware (CMW) [9], developed and used at CERN for the harmonization of the control systems of the LHC accelerator. CMW is based on CORBA and has been developed for UNIX-like real-time environments, including Linux and Lynx-OS. The CMW server normally runs directly on the front-end and can be directly accessed by applications and interfaces. Other platforms not compatible with UNIX (e.g., PLCs) can still be connected to CMW via a Gateway, which is a PC running Linux or LynxOS with which the front-end (e.g., the PLC) establishes a private communication with a platform specific protocol (e.g., for Siemens PLC

the CMW server retrieves data on the PLC using the fetch-write protocol).

Several platforms were benchmarked against the requirements of the collimation project, of which VME and the National Instruments PXI were selected for a more detailed study. Alternative platforms such as PCI or Compact PCI did not offer any particular advantage with respect to these two, neither from the point of view of the technical performance, nor cost.

VME had the advantage of being the standard real-time platform in use at CERN, therefore efficient internal support is available. This platform however would require either the purchase of extremely sophisticated and expensive commercial cards or the time consuming task of developing customized cards to suit the specific needs of the collimation project. In addition, the present real-time operating system in use at CERN on VME platforms is LynxOS, for which drivers are not often supplied with new cards. A considerable investment in manpower to write the new drivers had therefore to be taken into account.

By using the National Instruments PXI platform running LabView RT it has been possible to take advantage of the availability of a wide range of cards, all provided with their drivers. Several grades of motor controllers and I/O cards were tested before purchase, enabling an optimization of the cost of the overall system whilst avoiding unnecessary over-sizing.

The PXI platform was tested in the CERN SPS during three 24-hour runs with a real beam in November 2006 during which it performed satisfactorily and reliably. Following the recommendation of an internal review committee held in December 2006, the PXI platform running LabView RT was chosen for the implementation of the MDC and of the PRS in January 2007.

Since Labview RT runs on Pharlap, which is not a Unix-like OS, CMW could not be compiled due to its incompatibility with several necessary CORBA functions. To avoid a considerable investment of manpower in deploying CMW on Pharlap, it was necessary to run a CMW server on the PC Gateway and pass on to it all the data using Distributed Information Management (DIM) [10]. DIM is an open source middleware developed at CERN originally for the controls of the DELPHI experiment installed on the old LEP accelerator.

It is now widely used in the CERN experiments and is available on different platforms, including Windows and Linux. Starting from the Windows version, it was possible to establish a server running on Pharlap, while the client on the Gateway running LynxOS was compiled from the Linux version without any difficulty.

Fig. 2 shows the software architecture of the collimators control system. The two applications MDC and PRS, running on different machines, make available to the DIM server all necessary data and receive from it the commands from the CSS. The DIM client running on the PC Gateway fetches the data on the corresponding DIM server and makes them available to the CMW server, on which a FESA class is built, containing all the properties and methods to provide commands to the low level, and to preprocess the data for trasfer to the CSS. The FESA class also includes some coordination actions, for instance during calibration of the sensors, when the reading of the LVDT is correlated with the number of steps executed from the motor.

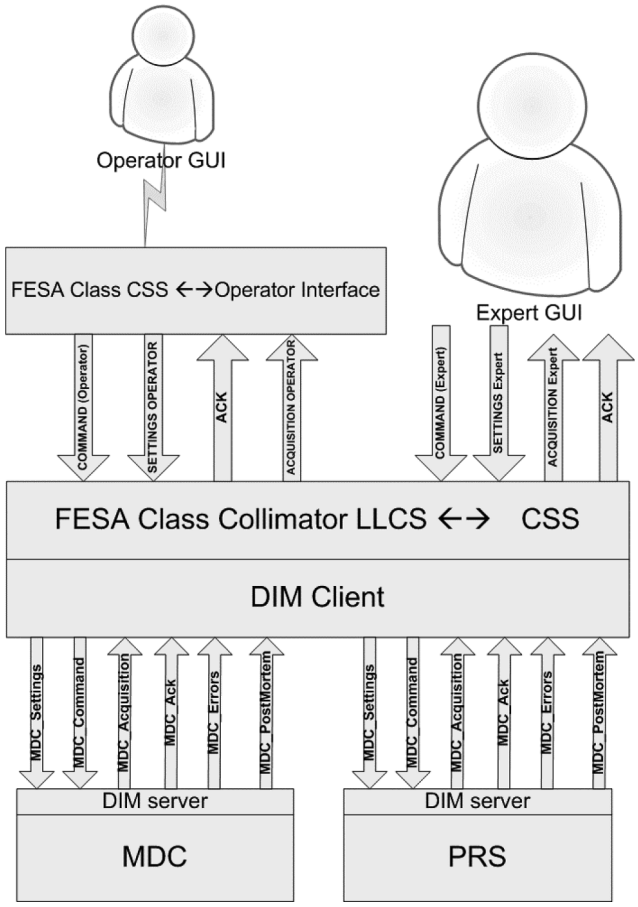

Fig. 2. The software architecture of the LHC collimators control system.

The implementation of the two low level modules in Labview RT is discussed in Sections IV and V. The remainder of this section highlights specific advantages and disadvantages of the PXI platform, clarifying its choice for this application.

PXI has the advantage of being easily synchronized using an external or internal 10 MHz signal that can be daisy-chained among the different chassis through pre-installed BNC connectors. For such a large installation as needed for the collimators, this is a critical advantage, since it requires the installation of only one oven stabilized 10 MHz quartz oscillator at each point of the LHC to synchronize the different clocks on which the generation of the steps for the stepping motors are based. In addition, several trigger lines are available on the backplane to further synchronize each card installed in the chassis.

The CPU in our configuration is an Intel® Core 2 Duo T7400 dual core processor at 2.16 GHz, which provides excellent performance in terms of speed of execution of real-time tasks. Even though the present version of LabView RT does not fully exploit the possibilities of parallel computing offered by the dual processor architecture, all the timing requirements have proved to be satisfied (once the applications are correctly optimized). The possibility to migrate to a full parallel version of LabView RT provides a safety margin to implement more sophistication in the program and to separate independent real-time tasks on different cores to increase the reliability of the system.

In addition, the system was purchased with 2 GB solid state disks instead of the hard disks that are supplied as standard.

Hard disks in fact are known to be one the first causes of failures of control systems in the CERN environment. The life-span of a solid-state drive is most often determined by the number of writes to a single storage location. Our application writes to the drive once every reboot (averaging once every few months). Therefore, this is not an issue. The MTBF of solid state drive installed (@ 25 deg. C) is > 4 million hours (i.e., > 456 years). The MTBF of an average mobile hard drive is ~ 300 K hours.

Each PXI chassis will control one, two or three collimators. MDC and PRS are implemented on different chassis to increase the reliability of the system. In fact, if the MDC fails, the PRS can still give information on the position of the collimator jaw, while if the PRS fails, with the MDC the jaws can still be returned to a predefined safe home position before the collimator is declared out of service. The PXI chassis have the additional advantage of being extremely compact and easy to connect to. All connections to digital signals pass through the NI Compact RIO modules, which offer the advantage of being compact, rugged and easy to mount, and provide additional features such as conditioning of RTDs.

Furthermore, all inputs/outputs are optically isolated and it is therefore possible to supply 24 V on all digital inputs/outputs, thus considerably increasing the overall noise immunity.

IV. The Motor Drive Control

A. General Description

The Motor Drive Control includes all the functions necessary to move the stepping motors of one, two or three collimators and verify that the movement has been executed as expected by reading the position of a resolver, mounted directly on the motor shaft.

The MDC application is responsible for:

- pre-processing the commands received from the CSS, for instance performing a coherence check and a protocol check;
- formatting data to be sent to the CSS, including acknowledgement of commands received and notifications of end of action;
- generation of trajectory set-points to be transferred to the FPGA motor controller card;
- interpolation of movement profiles sent by the CSS to make them smoothly executable by the controller;
- some of the interlocking functions;
- proper routing of the commands to the up to three collimators controlled by each chassis.

B. The Hardware Architecture

The hardware architecture of a MDC for 3 collimators is shown in Fig. 4.

The MDC PXI chassis hosts the Controller NI-PXI 8106 Core Duo, based on an Intel® Core 2 Duo Processor at 2.16 GHz.

Some of the MDC chassis will host the NI-PXI 6653 multichassis synchronization module which includes an OCXO to generate an extremely stable clock and the 10 MHz reference

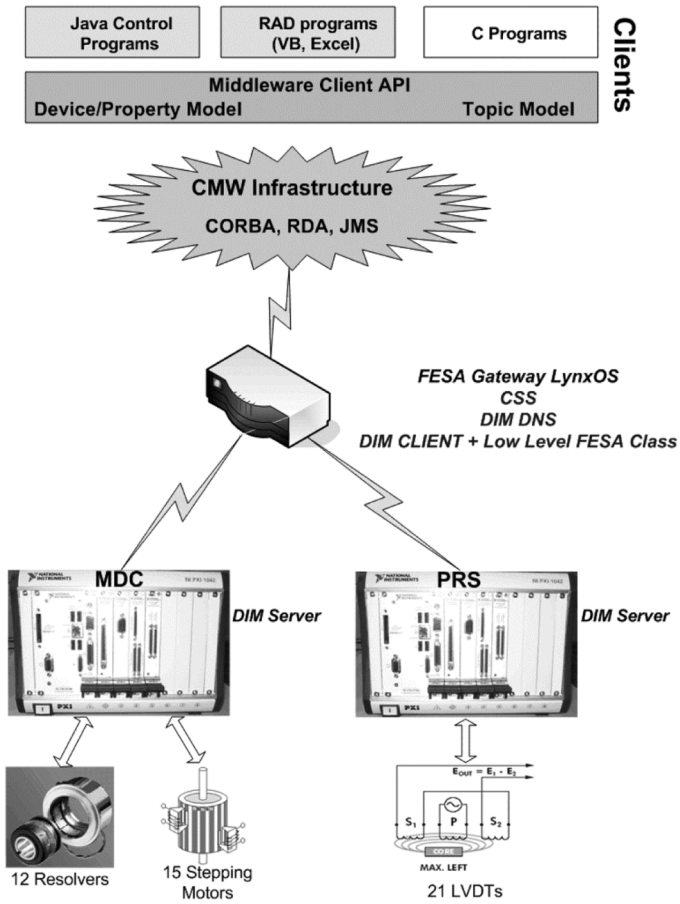

Fig. 3. Hardware architecture of the collimator control system.

signal (accurate to 45 ppb) that can be redistributed to all the other chassis geographically close to it to ensure low drift of the reference clocks of all the PXI chassis installed.

For the level of maximum relative drift required by the collimators application (< 1 ms from chassis to chassis), it is sufficient to rely on the stability of the 10 MHz signal generated by one NI PXI 6653 and redistributed to the other chassis.

Furthermore, for each collimator controlled by the chassis, we have installed a reconfigurable I/O card NI-PXI 7833R, based on a 3 MGates Xilinx FPGA providing 8 analog inputs, 8 analog outputs and 96 digital inputs/outputs. All inputs/outputs are completely reprogrammable by using either specific modules of Labview (Softmotion, FPGA) or directly in VHDL. Digital outputs were used for sending pulses to the STEP and DIRECTION inputs of an industrial driver for stepping motors, for the management of end switches and for interlocks, triggers, etc…. The analog inputs/outputs are used to generate the sinusoidal excitation signals and read the outputs of the 4 resolvers mounted on each collimator.

The digital inputs/outputs are connected to NI Compact Rio modules mounted on a Compact Rio passive chassis, which provide optocoupled ports, with the ability to excite the switches at 24 V thus considerably reducing the risk of spurious detections of the switch state. During a test in the real environment of a particle accelerator, EMI problems were encountered when directly connecting the long cables (up to 750 meters) coming from the

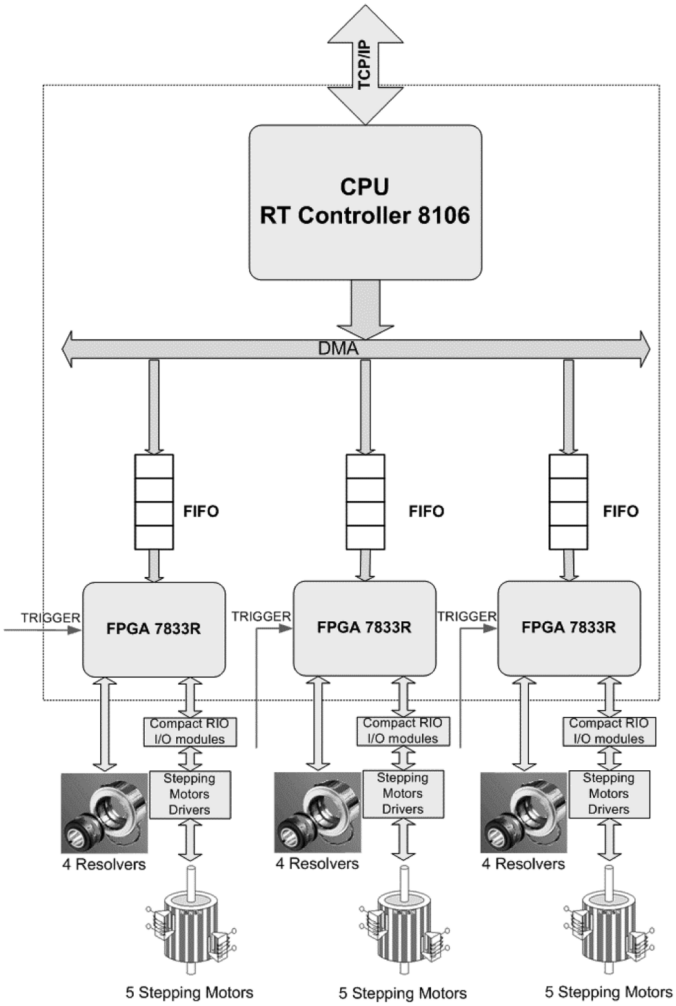

Fig. 4. The hardware architecture of the MDC.

tunnel to the FPGA, working at 5 V. The same is true for triggers, interlocks and any other digital signal.

Analog signals are dispatched and connected to the resolvers via custom CERN made Europa cards.[1]

These offer good modularity and ease of connection. EMC characteristics of the cards were simulated to ensure that no interference might be generated among the channels.

C. RT Tasks Architecture

Real-time features are requested for the MDC in the following tasks:

- the system should react within few ms (< 10) to a command received from the CSS;
- all the axes of a same collimator have to be synchronized to better than 1 ms. Any movement is started by an external trigger sent by the CSS and motor steps should start within a maximum delay of 1 ms, to be kept all along the trajectory (of up to 30 minutes and more);
- interlocks have to be generated as soon as unexpected conditions are detected (e.g., unexpected switch contact or lost steps).

[1]CEI 60297-3-101, CEI 60297-3-102, CEI 60297-3-103/IEEE 1101.1, IEEE 1101.10/11

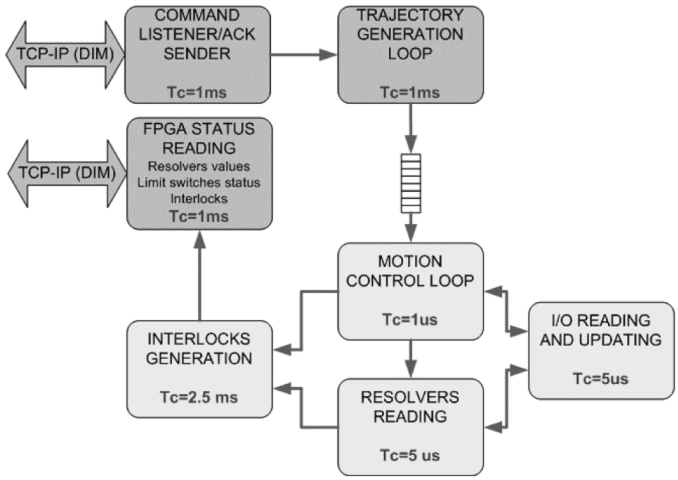

Fig. 5. The architecture of the real-time tasks of the MDC.

The architecture of the RT tasks executed on a MDC to control one collimator is shown in Fig. 5.

The DIM server is a thread running on the PXI controller at the same priority as the LabView RT MDC application. In Fig. 5 the tasks performed on the CPU and on the FPGA card are differentiated by gray shade (light-gray for FPGA, dark grey for CPU).

On the CPU, the ***command listener*** waits for commands or settings sent by the CSS via the related DIM service. As soon as the DIM service is updated, a LabView event is generated.

The timed loop used for the event recognition has a 1 ms cycle time and the highest priority to ensure a fast reaction time even when the CPU is preparing a motion profile. The most common commands consist of a simple displacement of the four collimator axes or of the execution of complex and synchronized motion profiles that are up to 20 minutes long. After checking the consistency with the interface protocol and the feasibility of the requested movement the profile setting is prepared in the trajectory generator loop and downloaded to the FPGA via a DMA FIFO. At the same time an acknowledge message is sent to the CSS to inform it that the MDC is ready to execute the requested profile. The NI Softmotion 2.0 library for Labview RT and NI RIO has been used to implement the motion generation for the 5 axes of the collimator. The working principle is shown in Fig. 6.

The ***trajectory generation loop*** also runs on the CPU and calculates the set-points for the control loop running on the FPGA, taking into account the jerk, acceleration and speed specified for the simple displacement or the time-position profile for each axis. A new set-point is generated per cycle.

The cycle for the generation is fixed at 1 ms to ensure that enough set-points are sent to the FPGA before a start trigger comes.

The ***control loop*** runs at a 1 MHz clock rate and generates the STEP and DIRECTION signals to drive each stepping motor driver in accordance with the set points fetched on the FIFO every ms. This architecture ensures a synchronization among the axes within the same card at μs level. The trigger delay response time was measured to be in average 84 μs with only few μs jitter. The skew after the execution of the same 15 minute profile on two different PXIs was a few hundred μs since the CLK/10 clock signal on the PXI RTSI bus is used for the FPGA clock. This is derived by the timing card PXI 6653 and it is characterized by a stability of 45 ppb.

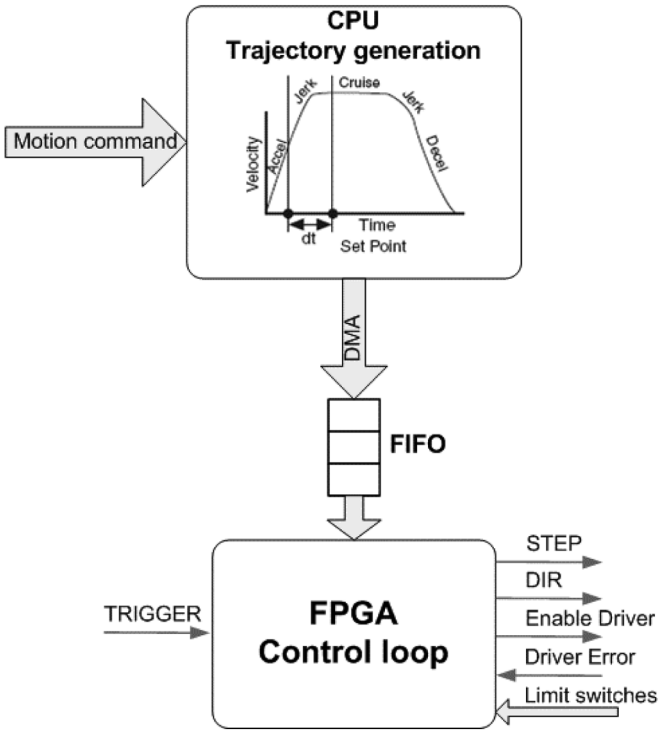

Fig. 6. Trajectory generation and execution.

The ***I/O reading*** task runs on the FPGS and updates the digital output signals and reads the input ones on the compact RIO modules as well as takes the samples of the resolvers' output signals and updates the analog outputs for the resolvers excitation. The update time is 5 μs both for the digital signals and for the analog one. This represents also the reaction time to a limit switch changing state.

The ***resolvers reading*** task is performed on the FPGA processing the arrays of the sine and cosine sampled signals from each resolver.

A demodulation scheme implemented with the CORDIC arithmetic has been developed to have a 13 bit resolution on the angle reading and 15 bit on the revolution measurement.

High noise immunity has been achieved using least-square FIR at 180 samples on both resolver output signals acquired.

The maximum reading frequency is 400 Hz that permits at the nominal motor displacement speed to check the resolver after each motor step demanded in order to detect immediately a step lost. Fig. 7 shows the details of the resolver decoding algorithm implemented. A Direct Digital Synthesizer (DDS) algorithm is used to generate the reference sine wave.

The *angle logic* block decodes the angle position using the two secondaries (cosine and sine) of the resolver. This process is based on the following steps:

1) Instantaneous values of resolver secondary voltages are used to calculate the angle position over a half-revolution. The conversion is done using the formula:

$$\text{angle\_half} = \arctan\left(\frac{\sin}{\cos}\right) \text{ if} \geq 0,$$

$$\arctan\left(\frac{\sin}{\cos}\right) + \pi \text{ otherwise.}$$

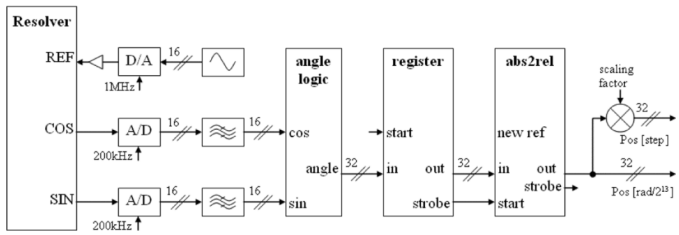

Fig. 7. The resolver to digital algorithm implemented on the FPGA.

2) The "COordinate Rotational DIgital Computer" (CORDIC) algorithm is used to perform the trigonometric calculation.
3) When both cosine and sine signals cross zero volts, the CORDIC algorithm may give unpredictable results for the angle. These values are filtered using a custom logic.
4) Transitions between $0 - \pi$ and $\pi - 2\pi$ are detected and counted to calculate the absolute position given in radians fixed point 32 bits with 13 bits for the fractional part.

In the Output Register an average of the instantaneous absolute position (256 samples) is performed on demand using the "start" signal which is delayed to compensate the input filters latency. The signal "strobe" indicates that a new position value is available at the output. At the end in the Absolute to Relative Converting the position is converted in integer step for an easier comparison with the controller position.

The ***interlocks generator*** task checks that no steps have been lost. It is implemented on the FPGA where the resolver reading is triggered by the steps generation. In this way the actual motor position is verified for each step. As soon as the difference between demanded and measured steps at a given moment exceeds a predetermined threshold, an interlock is generated (i.e the signal to dump the LHC beam).

Finally, an ***FPGA status reading*** on the CPU is performed with a 1 ms cycle time in order to collect at 100 Hz all the data required for a postmortem analysis in case of failure (e.g., resolvers data, switches status, axes controller position). The data sensors from resolvers and limit switches are sent to the CSS at low speed but within a delay of only 1 ms, the status of the command requested is updated.

V. Position Readout and Survey (PRS)

A. General Description

The PRS is responsible for checking the coherence of actual jaw positions of up to three collimators with those expected by the CSS. The absolute position of the jaws is measured by an LVDT installed on each axis (2 for each jaw). Two additional LVDTs measure the distance between different jaws of the same collimator. During the execution of a motion profile, the position readings are compared with threshold profiles previously sent by the CSS. The position survey is performed at a rate of 100 Hz to ensure, even at the maximum speed of 2 mm/s, the 20 μm tolerance specified for the project.

The functionalities implemented in the PRS are the following:

- pre-processing the commands received from the CSS: coherence check, protocol check;

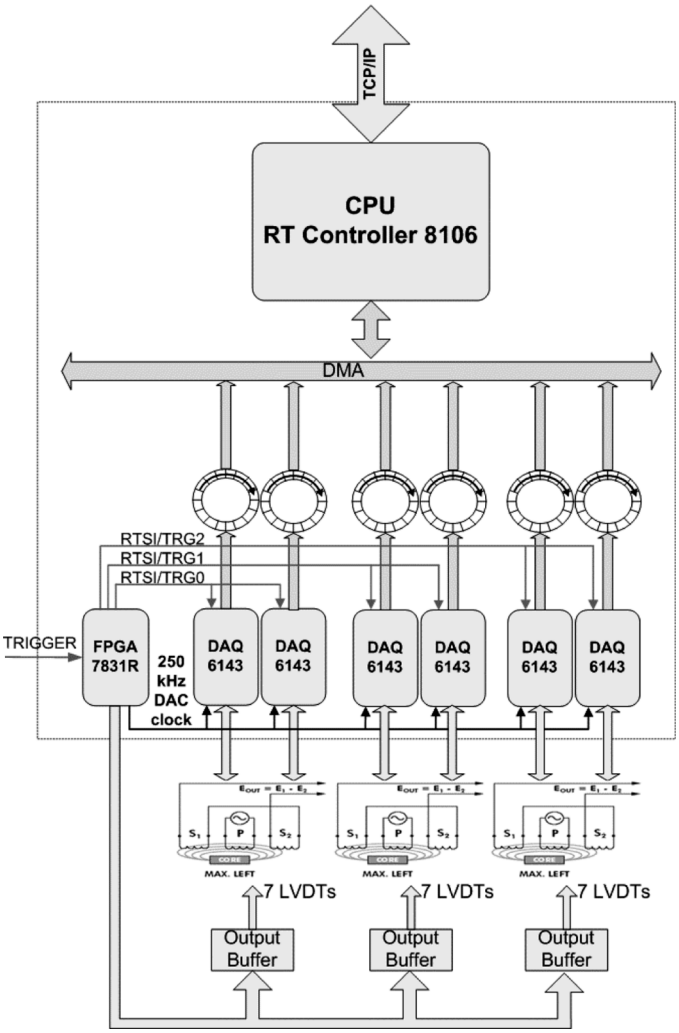

Fig. 8. The hardware architecture of the PRS.

- preparation of the upper and lower threshold profiles for the collimator axes and the two gaps;
- reading up to 21 LVDTs with 20 μm accuracy at a rate of up to 100 Hz;
- synchronization of the motion profiles survey process on the trigger received by the CSS.

B. The Hardware Architecture

The hardware architecture of a PRS for three collimators is shown in Fig. 8.

An RT controller 8106 based on an Intel® Core 2 Duo Processor at 2.16 GHz is used to serve up to three collimators.

Two DAQ cards NI PXI 6143 with 8 simultaneous differential analog inputs sampled at 16 bit and 250 kS/s is used to acquire the 14 signals coming from the secondary coils of the 7 LVDTs installed on each collimator. Data acquired are transferred to the CPU via DMA using circular buffers. In this way the LVDT signal acquisition is a background process and at each cycle the CPU reads the samples array needed for the demodulation from the circular buffer.

An FPGA card NI PXI 7831R having 8 analog outputs is used to generate the feeding signals for the 21 LVDTs. The frequencies of the 7 sine waves generated are orthogonal with respect

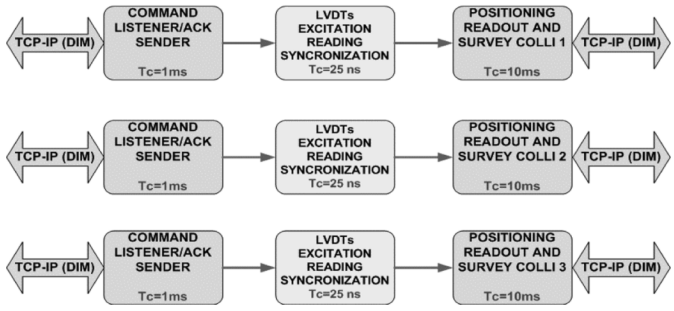

Fig. 9. The architecture of the real-time tasks of the PRS.

to the frequency response sine-fit algorithm used for sine wave demodulation in order to reduce the crosstalk between LVDTs of the same collimator [12].

All the signals sent to or coming from the LVDTs are interfaced through custom Europa cards hosting power buffers to properly drive each sensor over the long cables. This solution guarantees both reliability and modularity. In fact, even if the 7 analog output signals from the NI PXI 7831 are used to excite 21 LVDTs on three different collimators, the power stage is individual for each LVDT. For a system based on three collimators, the Europa chassis will contain only one generation card and three different acquisition cards, each of the latter ones used for one collimator. In case of a fault on a power stage of one of the collimators, the two others will not be affected. The same FPGA card is also responsible for the synchronization of all the DAQ cards on the precise clock at 10 MHz distributed to all the PXI chassis as well as the synchronization of the motion survey process for each collimator on the trigger signal sent by the CSS. A 250 kHz clock for all the DAQ cards is derived by the FPGA from the CLK/10 on the RTSI bus as reference clock. The start of the readout and survey loops for each collimator is triggered respectively by the signals TRG0, TRG1 and TRG2 on the RTSI bus generated by the FPGA upon reception of a trigger from the CSS.

C. RT Tasks Architecture

The architecture of the RT tasks executed on the PRS for the readout and survey of three collimators is shown in Fig. 9, where grey tasks are executed on the CPU and light-colored tasks on the FPGA.

The CPU ***command listener*** waits for commands or settings sent by the CSS on the related DIM services. As for the MDC the event recognition loop has 1 ms cycle time and the highest priority to ensure negligible reaction time. The main command concerns the request for motion profiles survey. As soon as this command is recognized the upper and lower limit arrays are prepared for each axis according to the boundary function sent by the CSS. A linear interpolation is applied to update the threshold value each 10 ms. An acknowledgement is sent to the CSS to inform it that the PRS is ready for the profile survey and is waiting for the hardware trigger.

The synchronization of the reading and survey process on the CPU is performed by the FPGA card (task ***LVDTs excitation reading and synchronization***). As soon as the trigger is received from the CSS a 100 Hz clock is generated on the RTSI

trigger line related to the collimator to be surveyed. This accurate timing signal synchronizes the CPU task that performs the LVDT readings and the comparison with the associated thresholds. This approach guarantees that the jitter on the LVDT readings will not exceed 100 μs. The error on position reading will therefore remain well within specifications.

The ***position readout and survey task*** on the CPU performs the reading of the 7 LVDTs installed on the collimator using a ratiometric conditioning [13]. The amplitude of the LVDT output signals is estimated with a sine fit algorithm [14] optimized for an RT implementation. The high noise immunity is guaranteed by the frequency response of the algorithm [12].

Accuracies of a few micrometers up to 800 m of cable between the 7 LVDTs and the electronic have been verified experimentally. The time needed to execute a full reading cycle for all the 21 LVDTs of one collimator using 2000 samples for each signal has been measured to be below 1.5 ms with a jitter below 50 μs [12].

VI. Conclusion

The collimation Low Level Control System integrates several functions that require real-time characteristics. The National Instruments PXI platform running Labview RT has been used to implement all real-time tasks. Using, DIM the low-level tasks have been reliably connected to the CERN middleware CMW. Two different PXI chassis are used to implement the functions of stepping motor control and position survey to ensure a good level of reliability. Motors can be synchronized to within 100 μs, resolvers can be read at 400 Hz and LVDTs at 100 Hz. The systems have been optimized so that a sufficient margin is left to implement digital filters to reduce EMI if necessary. Tests performed in laboratory and in a real accelerator show that a good immunity to noise has already been achieved.

Acknowledgment

The authors would like to thank R. W. Assmann, M. Jonker, M. Sobczak, and S. Redaelli for the long discussions on the architecture of the collimation control system. They are indebted to their colleagues of the AB/ATB/LPE Section, in particular A. Brielmann, G. Conte, M. Donze, P. Gander, J. Lendaro, and M. Martino for the fruitful collaboration on the implementation of the system. They are also indebted to a number of people in National Instruments, in particular B. Runnels, C. Loew, D. Shepard, G. Cirigioni, S. Concezzi, and C. Moser. Without their help and support that went beyond standard commercial relationships, the implementation of this project would never have been possible. Finally, the authors wish to thank C. Gaspar for her precious support and patience in assisting them with compiling DIM under Pharlap, a key to the success of the project.

References

[1] O. Brüning, P. Collier, P. Lebrun, S. Myers, R. Ostojic, J. Poole, and P. Proudlock, "The LHC main ring, LHC design report," CERN-2004 003, 2004.

[2] R. Assmann *et al.*, "The final collimation system for the LHC," presented at the Euro. Particle Accelerator Conf. (EPAC), Edinburgh, Scotland, U.K., 2006.

[3] O. Aberle *et al.*, "The controls architecture for the LHC collimation system," presented at the 10th ICALEPCS Int. Conf. Accelerator Large Expt. Phys. Control Syst., Geneva, Switzerland, 2005.

[4] A. Guerrero *et al.*, "CERN front-end software architecture for accelerator controls," presented at the 9th ICALEPCS Int. Conf. Accelerator Large Expt. Phys. Control Syst., Gyeongju, Korea, 2003.

[5] J. Serrano, P. Alvarez, D. Dominguez, and J. Lewis, "Nanosecond level UTC timing generation and stamping in the CERN's LHC," presented at the 9th ICALEPCS Int. Conf. Accelerator Large Expt. Phys. Control Syst., Gyeongju, Korea, 2003.

[6] D. Still, J. Annala, M. Church, B. Hendricks, B. Kramper, and A. Legan, "The Tevatron Collider Run II Halo Removal System," in *Proc. AIP Conf.*, 2003, pp. 176–179.

[7] R. Assmann, M. Jonker, M. Lamont, and S. Redaelli, "Application software for the LHC collimators and movable elements," CERN LHC-TCT-ES-0001-00-10, CERN EDMS 826861, 2007.

[8] L. Mestre *et al.*, "A pragmatic and versatile architecture for LHC controls software," presented at the 10th ICALEPCS Int. Conf. Accelerator Large Expt. Phys. Control Syst. , Geneva, Switzerland, 2005.

[9] K. Kostro, J. Andersson, F. di Maio, S. Jensen, and N. Trofimov, "The controls middleware (CMW) at CERN," presented at the 9th ICALEPCS Int. Conf. Accelerator Large Expt. Phys. Control Syst., Gyeongju, Korea, 2003.

[10] C. Gaspar and M. Donszelmann, "DIM—A distributed information management system for the DELPHI experiment at CERN," presented at the IEEE 8th Conf. Real Time Comput. Appl. Nucl., Particle Plasma Phys., VC, Canada, 1993.

[11] CEI 60297-3-101, CEI 60297-3-102, CEI 60297-3-103/IEEE 1101.1, IEEE 1101.10/11.

[12] A. Masi, A. Brielmann, R. Losito, and M. Martino, "LVDT conditioning on the LHC collimators," *IEEE Trans. Nucl. Sci.*, vol. 55, no. 1, pp. 67–75, Feb. 2008.

[13] J. Szczyrbak and E. D. D. Schmidt, *LVDT Signal Conditioning Techniques*. London, U.K.: Lucas Varity, 2005.

[14] *IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters*, IEEE Standard 1241, 2000.