UNIVERSIDADE DE SANTIAGO DE COMPOSTELA

DEPARTAMENTO DE ELECTRÓNICA E COMPUTACIÓN

TESE DOUTORAL

"Design and Development of Electronics and the Control Software for the Silicon Tracker of LHCb"

Presentada por:
Daniel Esperante Pereira

Dirixida por:
Bernardo Adeva Andany
Abraham Antonio Gallas Torreira
Pablo Vázquez Regueiro

Titor:
Diego Cabello Ferrer

Santiago de Compostela, Xuño de 2010

Dr. **Bernardo Adeva Andany,**

Catedrático de Física Atómica, Molecular e Nuclear da Universidade de Santiago de Compostela

Dr. **Abraham A. Gallas Torreira,**

Investigador de Física Atómica, Molecular e Nuclear da Universidade de Santiago de Compostela

Dr. **Pablo Vázquez Regueiro,**

Investigador de Física Atómica, Molecular e Nuclear da Universidade de Santiago de Compostela

Dr. **Diego Cabello Ferrer,**

Catedrático de Electrónica e Computación da Universidade de Santiago de Compostela

FAN CONSTAR:

Que a memoria titulada **"Design and Development of Electronics and the Control Software for the Silicon Tracker of LHCb"** foi realizada por D. **Daniel Esperante Pereira** baixo a nosa dirección e tutela no departamento de electrónica e computación da Universidade de Santiago de Compostela, e constitúe a Tese que presenta para optar ao Grado de Doutor en Ciencias Físicas.

Santiago de Compostela, a 28 de Xuño de 2010.

Asdo: **Bernardo Adeva Andany**

Codirector da tese de Licenciatura

Asdo: **Abraham A. Gallas Torreira**

Codirector da tese de Licenciatura

Asdo: **Pablo Vázquez Regueiro**

Codirector da tese de Licenciatura

Asdo: **Diego Cabello Ferrer**

Titor da tese de Licenciatura

Asdo: **Daniel Esperante Pereira**

Autor da Tese

AGRADECIMIENTOS

No resulta fácil escribir este apartado de agradecimientos sin olvidarse de alguien. Por eso, en general quiero agradecer a todos aquellos que durante mi vida han sido compresivos conmigo y que han dado lo mejor de sí mismos sin esperar nada a cambio.

Siento una gran gratitud hacia todos aquellos con los que he compartido horas de trabajo a lo largo de estos años y que han entendido en todo momento mi forma ser. No son pocas las personas con las que he compartido tareas y dedicación con el afán de conseguir que el trabajo fruto de esta tesis saliese adelante. Quiero agradecer por ello a la gente de la colaboración tanto de Santiago, como de Laussanne, Zurich y Heidelberg y a todos aquellos que he conocido en el CERN. En especial me gustaría nombrar a aquellos con los que he tenido más contacto, Eliseo, Carlos, Cristina, Alba, Antonio, Martín, Rodolfo, Pablo, Manolo, Abraham, el otro Pablo, Diego, Xabier, Louis, Angela, Marco, Helge, Jeroen, Clara, etc. Y que no se sientan excluidos de la lista todos aquellos que no aparecen y a los que he tenido el placer de "molestar" cuando las circunstancias lo han requerido.

Agradezco también a todos los jefes de la colaboración del Silicon Tracker haber depositado su confianza en mí para llevar a cabo el trabajo aquí presentado. Espero que en ningún momento supusiese para ninguno de ellos más que confianza un acto de fe.

Pero por encima de todas las cosas, quiero agradecer a todos aquellos con los que he compartido en mayor o menor medida mi vida personal. Quiero agradecérselo a mis amigos, a mis padres, hermanos, familiares y a mi novia. Es con todos ellos con quien he tenido el placer de pasar los momentos dulces y amargos que forman a una persona. Es gracias a ellos que soy como soy y que he conseguido que llegue a su fin el mayor logro académico de mi vida. Es por ellos que cada día que ha pasado he intentado hacerlo mejor que el anterior. Y es para ellos el honor de ver que todo se puede conseguir con trabajo, entrega e ilusión.

Les agradezco a mis padres haber luchado tanto en su vida para darme la oportunidad de dedicarle estas palabras. Gracias por darnos a mis hermanos y a mí las oportunidades de las que vosotros no pudisteis disponer en vuestro momento. Gracias a mis hermanos por todos los momentos que hemos vivido desde la infancia hasta ahora. Gracias a toda mi familia por haberme criado en la felicidad y la humildad y haber conseguido que os lleve siempre en mi corazón. Es gracias a todos vosotros que desde pequeño ha habido una inquietud dentro de mí que siempre me ha llevado a exigirme la máxima responsabilidad. Estoy seguro de que están más orgullosos que yo de ver que he alcanzado esta meta y se lo merecen más que nadie.

No podrían faltar las últimas palabras dedicadas a Paula. Gracias por haberme acompañado en estos años de mi vida y haber sacrificado parte de la tuya viniendo a Suiza. No hay pago que pueda compensar todo lo que has hecho por mí, ya que la felicidad no tiene precio. Han sido, y seguro que serán, los mejores 5 años de mi vida.

Se lo dedico a todos mis amigos, de antes y de ahora, a toda mi familia y a todos aquellos con los que he tenido el honor de disfrutar de los avatares de la vida.

Take it easy!

A mi abuelo Saúl,
reflejo de los valores de bondad, honestidad,
respetuosidad, sacrificio, educación y alegría.

Si todos fuésemos como él,
la felicidad en el mundo sería una rutina.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

BE: Back-End

CAN: Controller Area Network

CB: Control Board

CCPC: Credit Card PC

CERN: Conseil Européenne pour la Recherche Nucléaire

CMRR : Common Mode Rejection Ratio

CMS : Compact Muon Solenoid

COTS: Commercial Of The Shelf

CU: Control Unit

DAQ: Data Acquisition

Db: Database

DB: Digitizer Board

DCS: Detector Control System

DEN: Device Editor Navigator

DHFSM: Distributed Hierarchical Finite States Machines

DIM: Distributed Information Management

DIP: Data Interchange Protocol

DP: Data Point

DPE: Data Point Element

DPT: Data Point Type

DU: Device Unit

DSS: Detector Safety System

ECS: Experiment Control System

ELMB: Embedded Local Monitoring Board

FE: Front-End

FSM: Finite State Machine

FW: FrameWork

HEP: High Energy Physics

HIR: High Input Range

HLT: High Level Trigger

HV: High Voltage

HW: Hardware

I^2C: Inter Integrated Circuit bus

IT: Inner Tracker

JCOP: Joint COntrols Project

JTAG: Joint Test Action Group

LHC: Large Hadron Collider

LHCb: Large Hadron Collider beauty experiment

LIR: Low Input Range

LU: Logical Unit

LV: Low Voltage

LVDS: Low Voltage Differential Signaling

MARATON: Magnet field and Radiation Tolerant New Power Supply System

OPC: OLE for Process Control

PBUS: Parallel Bus

PLC: Programable Logic Controller

PLL: Phase-Locked Loop

PP: Patch-Panel

PS: Power Supply

PVSS: Prozessvisualisierungs- und Steuerungs-System

QPLL: Quartz Phase-Locked Loop

RH: Relative Humidity

RS: Read-out Supervisor

SB: Service Box

SCADA: Supervisory Control And Data Acquisition

SM: State Manager

SMC: Surface Mount Connector

SMD: Surface Mount Device

SMI: State Management Interface

SML: State Manager Language

SPECS: Serial Protocol for the Experiment Control System

ST: Silicon Tracker

SW: Software

TELL1: Trigger Electronics and L1 board

TFC: Time and Fast Control

TT: Tracker Turicensis

TTC: Timing and Trigger Control

INTRODUCIÓN

O Modelo Estándar da física de partículas é a combinación das teorías das interaccións forte e electro-débil. Describe as propiedades cinemáticas e as interaccións dos bloques elementais que constitúen a materia: quarks e leptóns. Aglutina o coñecemento actual da física de partículas. A súa validez susténtase por gran cantidade de evidencias experimentais e incluso foi capaz de predicir a existencia de partículas descoñecidas coma o W^+, W^-, e os bosóns Z^0; os mediadores da interacción débil. A teoría identifica aos gluóns coma os mediadores da interacción forte, os cales son bosóns sen masa e electricamente neutros que teñen carga de color e manteñen aos quarks ligados en nucleóns. Sen embargo, deixa preguntas sen resposta que o LHC (Large Hadron Collider) tentará contestar.

O Modelo Estándar non explica a orixe da masa das partículas, nin por qué algunhas partículas son moi pesadas mentres que outras non teñen masa. A resposta podería ser o coñecido mecanismo de Higgs. Nesta teoría, todo o espazo está ocupado por un 'Campo de Higgs', i é a través da interacción con este campo que as partículas adquiren a súa masa. Este campo de Higgs ten polo menos unha partícula asociada a el, o bosón de Higgs. Se esa partícula existe, os experimentos do LHC deberían ser capaces de detectala.

O Modelo Estándar non da unha descrición unificada de todas as forzas fundamentais, xa que é complicado construír unha teoría da gravidade semellante á das outras forzas. A teoría da supersimetría podería facilitar a unificación das forzas fundamentais. Se a supersimetrías existe, entón as partículas supersimétricas mais lixeiras debería encontrarse no LHC.

Observación astrofísicas e cosmolóxicas demostran que a materia visible supón só o 4% do Universo. Estase á busqueda de partículas ou fenómenos que expliquen a materia escura (23%) e a enerxía escura (73%). Unha hipótese é que a materia escura está composta por partículas supersimétricas neutras sen descubrir.

Segundo as teorías actuais, o Universo, nacido do Big Bang, pasou a través dunha fase na que a materia existente consistía nunha especie de plasma quente e denso composto por elementos fundamentais, o coñecido como quark-gluon plasma (QGP). A medida que o Universo se expandiu e arrefriou, os quarks pasaron a formar partículas compostas coma os protóns e neutróns. Este fenómeno chámase o confinamento dos quarks. O LHC será capaz de recrear o QGP acelerando e colidindo feixes de ións pesados. Nestas condicións de alta densidade de enerxía e temperatura, os quarks son libres de novo, e os detectores observarán e estudarán este plasma para esclarecer os principios básicos das partículas e de cómo se xuntan para constituír a materia.

O LHC tamén axudará a investigar o misterio da antimateria. Materia e antimateria producíronse na mesma cantidade no momento do Big Bang fai 13.700 millóns de anos. A enerxía transformouse en materia e antimateria. A medida que os pares materia-antimateria coliden e se aniquilan vólvense a transformar en enerxía. Por un anaco houbo un balance perfecto entre materia e antimateria. Con todo, a medida que o Universo se expandía e arrefriaba sufriu unha serie de cambios na súa composición. As partículas adquiriron as súas masas características e un fenómeno que diferenciou a materia da antimateria aconteceu, causando unha asimetría entre elas. I.e. a antimateria non é un "reflexo" perfecto da materia. Se substituísemos materia con antimateria e mirásemos ao resultado nun espello veriamos que o reflexo é imperfecto, e isto pode ser o motivo do desequilibrio no noso Universo, que está composto de materia. Noutras palabras, tras o Big Bang as leis da física deberon afectar de xeito distinto á materia e á antimateria, i.e. violando a simetría CP.

Para investigar este máis outros fenómenos estraños do Modelo Estándar, así como para buscar nova física, un novo colidor de hadróns foi construído no Laboratorio Europeo de Física de Partículas (CERN), o LHC. O acelerador comezou a operar en Novembro de 2009. Un dos experimentos do LHC está dedicado ao estudo de física de sabor: o experimento LHCb. O obxectivo principal é buscar evidencias indirectas de nova física na violación CP e en desintegracións raras dos hadróns "beauty" e "charm". Ate o de agora, as factorías de Bs e os programas experimentais do Tevatron mostráronnos que a física de sabor é totalmente consistente co mecanismo CKM. Por outro lado, o nivel de violación CP nas interaccións débiles do Modelo Estándar non explican a cantidade de materia no universo. Polo tanto, necesítase unha nova fonte de violación CP mais alá do Modelo Estándar para resolver o puzzle. O efecto desa nova fonte debería verse na física de sabor grazas á

mellora da precisión no programa experimental no futuro. Hai diversos modelos de nova física que explicarían contribucións maiores das esperadas na violación CP, na produción de desintegracións raras e incluso poderían xerar modos de desintegración non permitidos no Modelo Estándar.

Para avaliar as posibilidades mencionadas, o experimento LHCb estudará as asimetrías CP e as desintegracións dos mesóns B_d, B_s e D cunha gran estatística e usando moitas desintegracións distintas. Espérase unha sección eficaz na produción $b\bar{b}$ duns 500 μb a unha enerxía de 14 TeV, o cal fará ao LHC a maior fonte de mesóns B do mundo, cunha tasa de produción de quarks b en LHCb de 100 kHz. O experimento LHCb abrangue uns 700 científicos dunhas 50 universidades e laboratorios diferentes en 15 países. O detector LHCb está divido en sub-detectores, cada cal crucial para o funcionamento do experimento.

Esta tese céntrase no desenvolvemento da electrónica e software de control para o detector Silicon Tracker do experimento LHCb, así como na construción e na súa posta en marcha. Esta tese está pensada coma un documento auto-contido que permita entender como é o detector e como se opera. A construción do Silicon Tracker é froitoda colaboración de grupos de España (Santiago de Compostela), Suiza (Zurich e Laussanne) e Alemania (Heidelberg). O grupo de Santiago de Compostela, ao cal pertenzo, foi o responsable de construir e por en marcha o sistema de control, os sistemas de baixa e alta voltaxe do Silicon Tracker así como colaborar na construcción dos módulos de lectura do Inner Tracker.

Os sistemas de control dos experimentos do LHC presentan retos desafiante empezando polo tamaño e complexidade dos experimentos e seguindo polo nivel de fiabilidade requirido nun entorno tan duro como o das zonas experimentais. En particular, a inaccesibilidade á zona experimental durante o tempo de funcionamento xunto con entorno radioactivo supoñen un rigoroso requisito sobre o deseño de sistemas de control á hora de minimizar o risco de fallos e errores graves, incluíndo os humanos. A cantidade de datos manexados polos sistemas de control é comparable coas tasas de datos de experimentos anteriores.

Interconexións complexas entre distintos subsistemas (DCS, *Data Acquisition*, *Trigger* and *Accelerator systems*) implican construír sistemas de control que empregan lóxica complexa e diversas capas de abstracción e integración.

A idea de utilizar Maquinas de Estados Finitos (Finite State Machines ou FSM en inglés) no deseño de sistemas de control intrinsecamente evita potenciais problemas e representa unha innovación. A industria emprégaas dende fai tempo nos procesos de control. Sacando proveito de este "saber-facer" e da experiencia en anteriores experimentos de física de altas enerxías, tentamos aplicar a tecnoloxía a sistemas de maior envergadura. De feito, a fiabilidade dos sistema industriais indica en certo modo que o camiño elixido é o correcto. Do mesmo xeito, experiencias previas no CERN co experimento DELPHI proporcionan pautas a seguir.

A tese está estruturada de xeito que os elementos que constitúen o detector preséntanse dende o punto de vista en que está dividido o sistema de control, i.e. nun modo funcional. Nalgúns casos preséntase traballo feito por outra xente da colaboración xa que é esencial para comprender como funciona o sistema.

O capítulo 1 é unha introdución ao LHC e o experimento LHCb. O capítulo 2 dá unha visión xenérica da electrónica do Silicon Tracker de xeito que os capítulos seguintes, onde a electrónica se explica por dominios funcionais, son mais doados de entender. O capítulo 3 describe a Control Board, que é o hardware deseñado en construído por min mesmo e que é un dos elementos fundamentais desta tese. O capítulo 4 explica a arquitectura do sistema de LV mentres que o sistema de HV se presenta no capítulo 5. O capítulo 6 describe os sistemas de supervisión de temperatura e humidade. O capítulo 7 mostra a descrición do sistema de DAQ do Silicon Tracker. Os capítulos 8, 9, 10 e 11 describe o software do sistema de control. O capítulo 12 detalla o deseño e implementación da lóxica paralela á de control encargada de manter o detector nun estado seguro. O capítulo 13 mostra un resumo dos resultado obtidos ao largo do tempo de traballo na tese.

INTRODUCTION

The Standard Model of particle physics is a combination of the theories of the electroweak and strong interactions. It describes the kinematical properties and interactions of the elementary building blocks of matter: quarks and leptons. It summarizes our current knowledge of the particle physics. It has been supported by a great deal of experimental evidence and it has proven particularly successful in anticipating the existence of previously undiscovered particle like the W^+, W^-, and Z^0 bosons; the force mediators of the weak interaction. The theory identifies the force mediators of the strong interaction as gluons, massless electrically neutral bosons that carry color charge and bind the quarks together in nucleons. However, it leaves many unsolved questions, which the LHC (Large Hadron Collider) will try to answer.

The Standard Model does not explain the origin of the mass of the particles, nor why some particles are very heavy while others are massless. The answer may be the so-called Higgs mechanism. In this theory, the whole space is filled with a 'Higgs field', and by interacting with this field, particles acquire their mass. This Higgs field has at least one new particle associated with it, the Higgs boson. If such a particle exists, experiments at the LHC may be able to detect it.

The Standard Model does not offer a unified description of all the fundamental forces, as it remains difficult to construct a theory of gravity similar to those for the other forces. The supersymmetry theory could facilitate the unification of fundamental forces. If supersymmetry is right, then the lightest supersymmetric particles should be found at the LHC.

Astrophysical and cosmological observations have shown that all the visible matter accounts for only 4% of the Universe. The search is open for particles or phenomena responsible for dark matter (23%) and dark energy (73%). A possible hypothesis is that dark matter is made of neutral (but still undiscovered) supersymmetric particles.

According to the current theories, the Universe, born from the Big Bang, passed through a stage during which matter existed as sort of extremely hot and dense plasma composed of the elementary blocks of matter, the so-called quark-gluon plasma (QGP). As the Universe expanded and cooled down, the quarks became trapped into composite particles such as protons and neutrons. This phenomenon is called the confinement of quarks. The LHC will be able to recreate the QGP by accelerating and colliding beams of heavy ions. In these conditions of extreme energy density and temperature, the quarks are free again, and the detectors will observe and study this plasma, thus probing the basic properties of the particles and how they aggregate to form matter.

The LHC will also help us to investigate the mystery of antimatter. Matter and antimatter must have been produced in the same amounts at the time of the Big Bang 13.7 billion years ago. Energy was converted into particles of matter and antimatter. Pairs of matter and antimatter particles collided and annihilated each other, turning back into energy. For a short time there was a perfect balance or symmetry between matter and antimatter. However as the Universe expanded and cooled it went through a series of changes in its composition. Particles acquired their characteristic masses and a phenomenon occurred that differentiated matter and antimatter, causing asymmetry between the two. I.e. antimatter is not a perfect "reflection" of matter. If we replace matter with antimatter and looked at the result as if in a mirror we see that the reflection is imperfect, and this could have led to the matter-antimatter imbalance in our Universe, that is made of matter. In other words, after the Big Bang, physical laws must have acted differently for matter and antimatter, i.e. violating CP symmetry.

To investigate these and other outstanding issues in the Standard Model, as well as to directly search directly for new physics, a new hadron collider has been built at the European Laboratory for Particle Physics, CERN: The Large Hadron Collider. The accelerator started its operation in November 2009. One of the experiments at the LHC is dedicated to the study of heavy flavour physics: the LHCb experiment. Its primary goal is to look for indirect evidence of new physics in CP violation and rare decays of beauty and charm hadrons. So far, B-factories and the Tevatron's experimental programs have taught us that heavy flavor physics is fully consistent with the CKM mechanism. On the other hand, the level of CP violation in the

Standard Model weak interactions cannot explain the amount of matter in the Universe. Hence, we need a new source of CP violation beyond the Standard Model to solve this puzzle. The effect of such a new source might be seen in heavy flavour physics, given the fact of the much improved precision in the experimental program expected in the future. There are many models of new physics which produce contributions that change the expectations of for instance CP violation phases, rare decay branching fractions, and may even generate decay modes forbidden in the Standard Model.

To examine the aforementioned possibilities, the LHCb experiment will study CP asymmetries and rare decays of B_d, B_s and D mesons with high statistics and using many different decay modes. A large $b\bar{b}$ production cross section of about 500 μb is expected at an energy of 14 TeV, and will make the LHC the most copious source of B mesons in the world with a rate of *b*-quarks produced in the LHCb detector of 100 kHz. The LHCb experiment is the result of a collaboration of 700 scientists from about 50 different universities and laboratories in 15 countries. The LHCb detector consists of a number of subdetectors, each of which is crucial for the operation of the experiment.

This thesis is devoted to the development of the control electronics and software for the Silicon Tracker detector of the LHCb experiment as well as to the construction and commission of the so-mentioned detector. This thesis was thought as a self-contained document that permits understanding how the Silicon Tracker detector is like and how it can be operated. The construction of the Silicon Tracker detector is fruit of a collaboration of groups from Spain (Santiago de Compostela), Switzerland (Zurich and Laussanne) and Germany (Heidelberg). The group of Santiago de Compostela, to which I belong to, was the responsible to construct and commission the control system, the low voltage and high voltage systems of the Silicon Tracker as well as to collaborate in the construction the Inner Tracker sensor modules.

The Detector Control Systems of the LHC experiments face enormous challenges arising from the size and complexity of the experiments, as well as from the unprecedented level of reliability required in the harsh experimental environment. Especially the inaccessibility of the experimental site during the run combined with radiation environment imposes stringent requirements on the design of the control systems to minimize the risk of serious errors and faults, including the human ones. The amount of data handled by these control systems is now comparable with the physics data rates from previous generation of experiments.

Complex interconnections among various subsystems (DCS, Data Acquisition, Trigger and Accelerator systems) require the use of a sophisticated logic and several layers of abstraction and integration built into the control systems.

The approach of Finite State Machines (FSM) to the design of control systems avoids intrinsically several possible problems and represents an innovation. The industry has since long time adopted the same approach in its process control. Profiting of this know-how and past experience in previous high energy physics experiments, now we intend to extend the technology to much larger systems. In fact, the stringent standard of reliability of the industrial automation makes us confident that the chosen approach is effective. In addition the experience already done at CERN with the DELPHI experiment at LEP provides with useful guidelines.

This thesis is structured such the constituent elements of the detector are presented in the way the control system layout is arranged. In some cases, work developed by other people of the collaboration is presented, since it is essential to understand how the system functions.

Chapter 1 gives an outline of the LHC accelerator complex, and describes the LHCb detector. Chapter 2 gives an overview of the whole Silicon Tracker electronics, so the subsequent chapters, where the electronics is explained in a functional domain basis, are easier to follow. Chapter 3 describes the Control Board, the custom hardware designed and constructed by myself, which is one of the basic elements of this thesis. Chapter 4 explains the architecture of the LV system hardware whilst the HV system is reported in Chapter 5. The Chapter 6 describes the temperature and humidity monitoring system infrastructure. The Chapter 7 gives a description of the Silicon Tracker DAQ system and basic building blocks. The chapters 8, 9, 10 and 11 describe the control system software. The Chapter 12 shows a summary of results obtained during the development of the thesis.

CHAPTER 1

THE LHCB DETECTOR AT THE LHC

ABSTRACT

This chapter provides an overall description of the LHCb detector.

"Quien volviendo a hacer el camino viejo aprende el nuevo, puede considerarse un maestro"

Confucio

1.1 THE LARGE HADRON COLLIDER

The Large Hadron Collider [1] (LHC) at CERN, the European Laboratory for Particle Physics, is a two-ring superconducting hadron accelerator and collider located near Geneva, Switzerland. It is the most powerful accelerator in the world, designed to collide protons at a total central-of-mass energy of $\sqrt{s}$ = 14 TeV at a peak luminosity of $\mathscr{L} = 10^{34}$ cm^{-2}s^{-1}. In addition, the LHC will accelerate and collision heavy ions (e.g., Pb) at an energy of 2.76 TeV per nucleon, corresponding to a center-of-mass energy of 1150 TeV and a nominal luminosity of $\mathscr{L} = 10^{27}$ cm^{-2}s^{-1}. The LHC is installed in the 26.7 km tunnel that was constructed for the electron-positron collider LEP between 1984 and 1989. The tunnel was built at an average depth of 100 m, due to geological considerations (translating into cost) and at a slight slope of 1.4%. Its depth varies between 175 m (under the Jura) and 50 m (towards Lake Geneva). The approval of the LHC project was given by the CERN Council in December 1994, however the installation work in the tunnel could only start after the LEP accelerator had been dismantled in 2001. On September 10, 2008, protons circulated in the LHC ring for the first time.
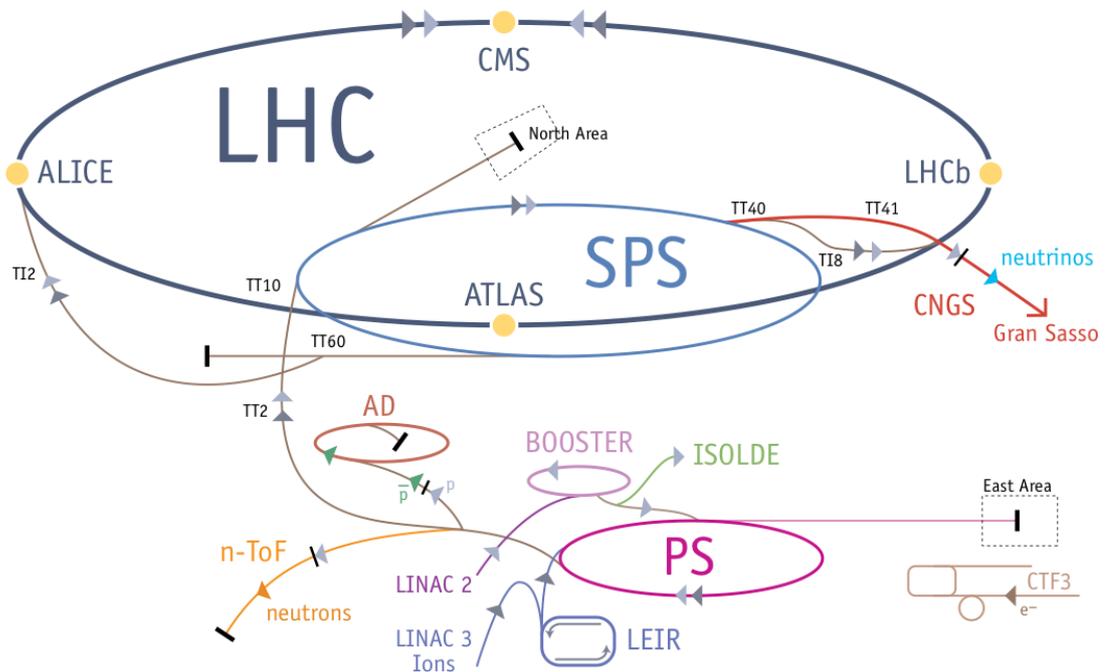
Figure 1.1: The CERN accelerator complex.

The whole CERN accelerator complex is shown in Figure 1.1. The protons which are obtained by removing electrons from hydrogen atoms are passing through a linear accelerator called LINAC 2 and are injected into the booster with an energy of 50 MeV. In this first circular accelerator the protons are further accelerated up to 1.4 GeV before they are directed to the Proton Synchroton (PS) that they leave with 26 GeV for the Super Proton Synchrotron (SPS). Via two injection lines (TI 2 and TI 8) the two proton beams are injected with 450 GeV energy into the LHC accelerator, where they finally reach the maximal energy of 7 TeV. To keep the two circulating proton beams in their orbits a total of 1232 superconducting dipole magnets are needed. To reach the required field strength of 8.33 T the magnets are cooled down to 1.9 K using super-fluid helium. At design luminosity the protons are accelerated in bunches of 1.15 x 10^{11} protons each, with 24.95 ns bunch spacing. Only 2808 bunches are filled out of the 3564 possible bunches in each beam direction. This corresponds to a total beam current of 0.582 A, equivalent to a stored energy of 362 MJ per beam. In addition, the electromagnetic energy stored as electrical current in the magnets is 600 MJ. Dedicated systems can safely absorb the total of about 1 GJ in case of malfunction, emergency, or simply at the end of each run.

The four largest experiments, namely ATLAS ("A Toroidal LHC Apparatus"), CMS ("Compact Muon Solenoid"), LHCb ("Large Hadron Collider beauty") and ALICE ("A Large Ion Collider Experiment"), are located at four interaction points around the LHC where the two proton beams are brought to collision. ATLAS [2] and CMS [3] are general-purpose experiments, designed to cover the widest possible range of

physics at the LHC, from the search for the Higgs boson to supersymmetry (SUSY) and extra dimensions. The ALICE [4] experiment focuses on studying strongly interacting matter at the extreme energy densities in heavy-ion collisions and performing measurements of the phase transition between hadronic matter and the quark-gluon plasma. A state of matter where quarks and gluons, under conditions of very high temperatures and densities, are not longer confined inside the hadrons. Such state of matter probably existed just after the Big Bang, before particles such as protons or neutrons were formed. LHCb [5] is the LHC experiment dedicated to the physics of the beauty quark, and will study CP–violating processes and rare decays. It was designed as a single-arm forward spectrometer and is described in the next section.

The view of the location of the LHC and of these four experiments is shown in Figure 1.2.
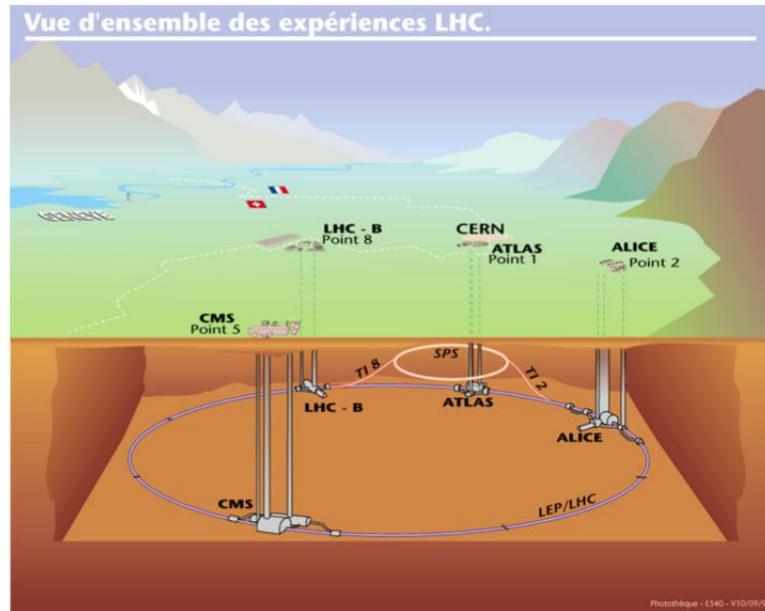
Figure 1.2: View and location of the LHC and the four experiments ATLAS, CMS, LHCb and ALICE.

Apart from the above-mentioned large experiments installed in four huge underground caverns built around the collision points, there are two smaller experiments at the LHC, TOTEM ("TOTal Elastic and diffractive cross section Measurement experiment") and LHCf ("Large Hadron Collider forward ") experiments. The TOTEM [6] experiment aims at the simultaneous measurements of the total pp cross-section and the luminosity. It covers the very forward region in the pseudo-rapidity range $3.1 \geq |\eta| \geq 6.5$. It consists of Roman pots placed several hundred meters on either side of the CMS interaction point, as well as detectors integrated in the CMS apparatus at about ten meters from the interaction point. LHCf [7] is another very forward experiment, located 140 meters on either side of the ATLAS experiment interaction point. The motivation is to test hadronic models at very high energy to be used in the understanding of ultra-high energy cosmic rays.

1.2   THE LHCB DETECTOR

The LHCb detector [5] is a single-arm spectrometer with an angular acceptance of 10 mrad to 300 mrad (250 mrad) in the bending (non bending) plane. This corresponds to a pseudorapidity coverage of $2.1 < \eta < 5.3$ and $1.7 < \eta < 5.3$ respectively. The layout is justified by the fact that B mesons are produced predominantly in the same forward and backward region at polar angles around zero and π respectively (Figure 1.3). Figure 1.4 shows a side view of the detector. The LHCb experiment is located at the LHC Intersection Point 8 in the cavern where the DELPHI experiment was installed during LEP operation. To accommodate the full length of the detector in the cavern, the interaction point has been shifted 11.25 m from the centre of the cavern in the direction of IP7.

The experiment uses a right-handed coordinate system, with the z-axis pointing downstream along the beam direction and the y-axis in the upwards direction. The origin is located at the centre of the interaction region.

A warm dipole magnet is used to deflect charged particles, thus providing an observable for a particle momentum measurement. The main component of the magnetic field is in the y-direction and the integrated

magnetic field is 4.2 Tm. At the nominal luminosity of 2×10^{32} cm^{-2}s^{-1} an estimated number of 10^{12} b$\bar{\text{b}}$ pairs will be produced and in the LHCb acceptance each year, corresponding to a rate of 100 kHz.
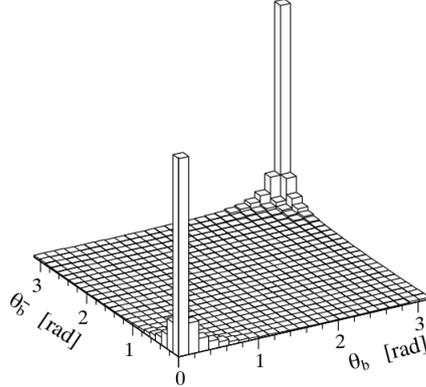
Figure 1.3: The polar angle correlation between two b-quarks produced in a pp collision at LHC predicted by a Monte Carlo Simulation.
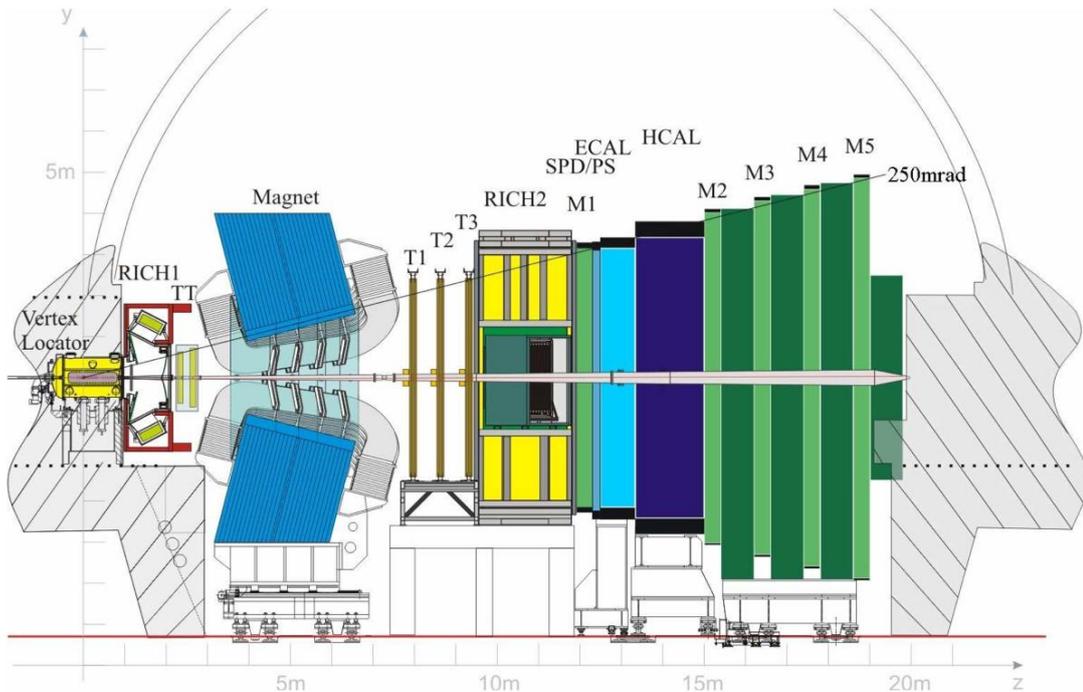
Figure 1.4: A cross-section of the LHCb detector in the non-bending yz-plane. The interaction point is located inside the Vertex Locator.

At LHC design luminosity of 10^{34} cm^{-2}s^{-1}, 27 inelastic interactions per collision occur on average. However, LHCb is not utilizing the full luminosity potential of the LHC. The mean luminosity is adjusted to 2×10^{32} cm^{-2}s^{-1} by a softer focus of the beams. By operating at limited luminosity, the average number of inelastic interactions per bunch crossing at IP8 is 0.54. One inelastic interaction occurs in about 30% of the collisions andthe probability of more than one interaction is less than 10%. By limiting the number of multiple interactions, it is easier to unambiguously identify the displaced vertices that are characteristic for B-meson decays and to link these vertices to the correct primary vertex. To further avoid pollution from multiple interactions in one event ("pile-up events"), a dedicated detector upstream of the interaction point vetoes bunch-crossings with more than one interaction. The limited luminosity has further experimental advantages compared to LHC design luminosity operation: the detector has to withstand a lower radiation dose and the individual subdetectors get lower hit multiplicities, which is beneficial for event reconstruction. Nevertheless, the rapidity region explored by LHCb has a high particle density. It is important to minimize the number of secondary particles created from interactions of protons incident on the beam pipe. The beam pipe through

LHCb is therefore of a conical design and its first 12 m (out of 19 m) are made out of 1 mm thick beryllium, which has a large radiation length and a good elasticity. This makes it transparent to primary particles while withstanding the external atmospheric pressure. The design pressure in the beam pipe is 10−9 mbar. The most downstream section of the beam pipe, which is out of the critical zone of transparency, is made of stainless steel.

The LHCb detector is composed of several subdetectors that can be divided in two main types depending on their tasks.

The tracking system consists of the Vertex Locator (VELO), the Tracker Turicensis (TT) and the tracking stations T1-T3. The tracking stations T1-T3 are actually two subdetectors: the Inner Tracker (IT) and the Outer Tracker (OT). The tracking system reconstructs particle trajectories across the spectrometer. The momenta of the particles are measured using the curvature of the tracks in conjunction with the very well known magnetic filed produced by the dipole magnet. The tracking system determines as well the position of the primary and secondary vertexes produced in the collisions.

The particle Identification (PID) system uses three detection technologies: The Ring Imaging Cherenkov detectors (RICH1 and RICH2) provide K/π separation. The electromagnetic and hadronic calorimeters (SPD, PS, ECAL, and HCAL) provide e/γ separation and hadron identification, and finally the muon chambers (M1-M5) perform muon identification.

1.2.1 TRACKING SYSTEM

A. VERTEX LOCATOR

The Vertex Locator system (VELO) [8] is the first tracking detector of the LHCb experiment, positioned before the magnet. The purpose of the this silicon micro-strip detector is to measure very precisely the tracks of charged particles resulting from *b* hadron decays very close to the beam pipe. The sensors, arranged in 23 consecutives planes perpendicular to the beam axis, have circular and radial strips in order to measure the radial (r) and azimuthal (ϕ) component of the tracks. Each station in these 23 planes is composed of two half-moon shaped planes, one on each side of the proton beam. During data taking the sensors are positioned 8.2 mm from the beam axis, while during beam injection and phases of instable beam the sensors are moved away from the beam axis mechanically to prevent any damage. The VELO is placed inside the LHC vacuum system in order to minimize the distance to the interaction point and the amount of material in between as small as possible. Two of the stations make up the Pile-up Veto detector. They are used in the first level trigger (L0)

B. TRACKER TURICENSIS

The Tracker Turicensis (TT) [9], is located in the fringe field upstream of the LHCb 4 Tm dipole magnet (Figure 1.4). This ~150 cm wide and ~130 cm high planar tracking station covers the full detector acceptance. It has been constructed using p-on-n type sensors with a pitch of 183 μm and a thickness of 500 μm with the same design as in the CMS barrel tracker. It is composed of four layers arranged into two half stations separated 30 cm along the beam axis, and in orientation 0°, 5°, -5°, 0°. Depending on the position in the layer up to four sensors are bonded together giving strip lengths up to 37 cm. Its four planer detection layers are depicted in Figure 1.5.

C. INNER TRACKER

The Inner Tracker (IT)[9]. It covers the region of highest particle density closest to the LHC beam pipe in the three T stations located downstream of the magnet. An IT station consists of four independent boxes arranged around the beam pipe in a 125 cm wide and 40 cm high cross shape (Figure 1.5). Each box contains four layers of silicon micro-strips in orientation 0°, 5°, -5°, 0°. The ladders placed left and right of the beam pipe are 22 cm long with a thickness of 410 μm, the ladders above and below the beam pipe are 11 cm in length and with a thickness of 320 μm, in both cases the read-out pitch is 198 μm.

D. OUTER TRACKER

The Outer Tracker (OT) [10] is a straw-tube detector surrounding the IT in the T1-T3 tracking stations (Figure 1.4). This detector determines the positions of the charged particles by measuring the drift times in straw tubes. The maximum drift time is kept below 50 ns and the drift coordinate resolution is about 200 μm. The OT modules are arranged in three stations like the IT. Each station consists of four layers in the same orientations as the IT and TT. The modules that make up an OT layer consist of two staggered "monolayers" with 5 mm diameter straw-tubes, as it is shown in Figure 1.6.
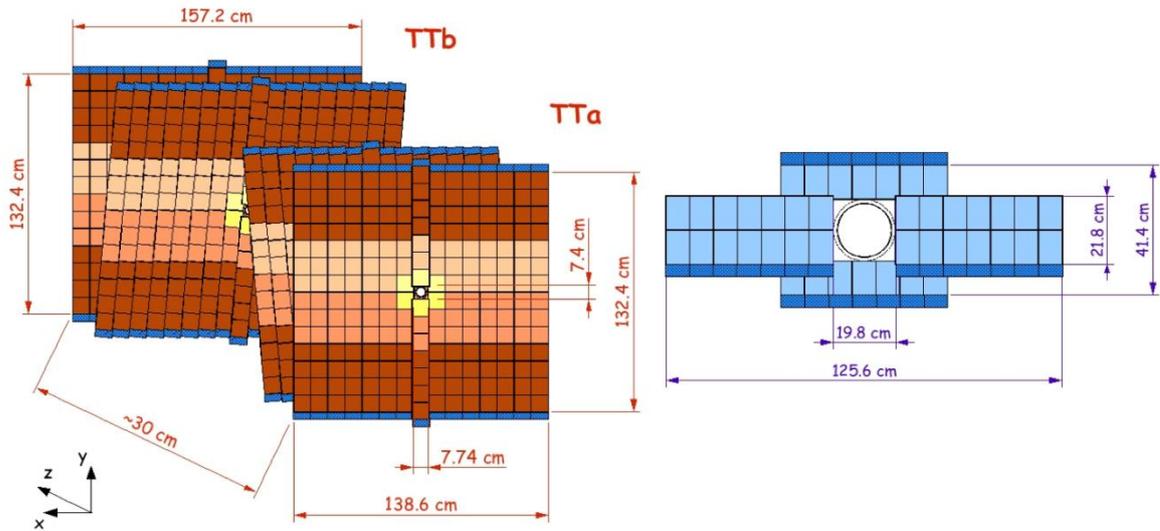
Figure 1.5: Illustration of the TT (left). Sketch of the first IT station (right).
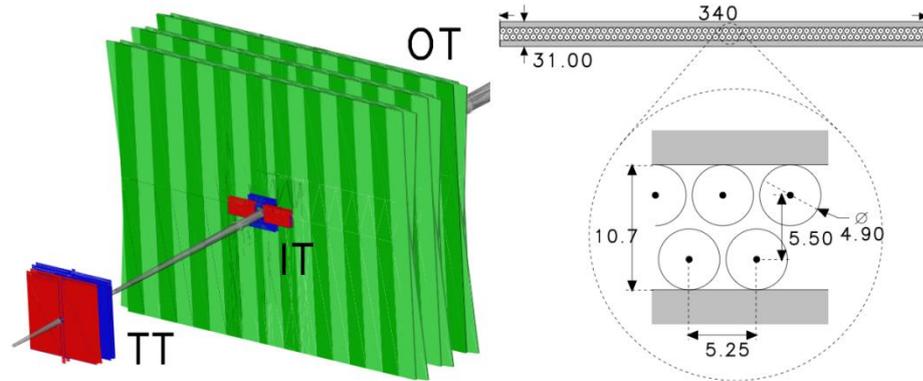
Figure 1.6: A perspective view of the TT, IT and OT detection layers (left).Cross-section of an OT module, showing the staggered mono-layers and the layout of the drift-tubes. Sizes are in millimeters (right).

1.2.2   PARTICLE IDENTIFICATION SYSTEM

A.   RICH

Reliable particle identification is one of the main requirements of the physics program ofthe LHCb experiment. In particular, we need to separate pions from kaons in B meson decays to reduce the background from $B_d \rightarrow K^{\pm}\pi^{\pm}$, $B_s \rightarrow K^{\pm}\pi^{\pm}$ and $B_s \rightarrow K^{\pm}K^{\mp}$. For this purpose, two Ring Imaging Cherenkov (RICH) detectors [11] are used. They provide particleidentification in a momentum range from 1 to 100 GeV/c. This is achieved by using three different radiators: aerogel and C_4F_{10} in RICH1 (1-60 GeV/c) and CF_4 in RICH2 (15-100 GeV/c).

RICH1 is placed upstream the dipole magnet, between the TT and the VELO, while RICH2 is situated between tracking station T3 and the first muon station (Figure 1.4). The design of the two detectors is depicted in Figure 1.7.

B.   CALORIMETERS

The calorimeter system [12] is placed between the muon stations M1 and M2 (Figure 1.4). It has three main purposes: it will trigger on electrons, photons and hadrons, it will measure energies and positions of traversing particles and it will identify photons and neutral pions to study specific B meson decays. The whole system consists of several layers: the Scintillating Pad Detector (SPD), the Pre-Shower (PS), the Electromagnetic Calorimeter (ECAL) and the Hadron Calorimeter (HCAL) (Figure 1.4). The SPD and the PS are placed before the ECAL and the HCAL. They determine the charge (charged or neutral) and the electromagnetic character of the traversing particles. Both the ECAL and the HCAL work with a basic

similar principle: incident particles interact with the absorber material (lead in the ECAL and iron in the HCAL) of the calorimeter, creating showers of secondary particles that will produce light as they pass through the scintillators inside the calorimeter. This light is collected by photomultiplier tubes and its amount can be related to the energy of the original incident particle. The ECAL has been designed to fully contain the shower of electrons and photons, while the HCAL fully absorbs the energy of the hadrons. The hit density depends on the distance to the beam axis and varies over the calorimeter acceptance by two orders of magnitude.
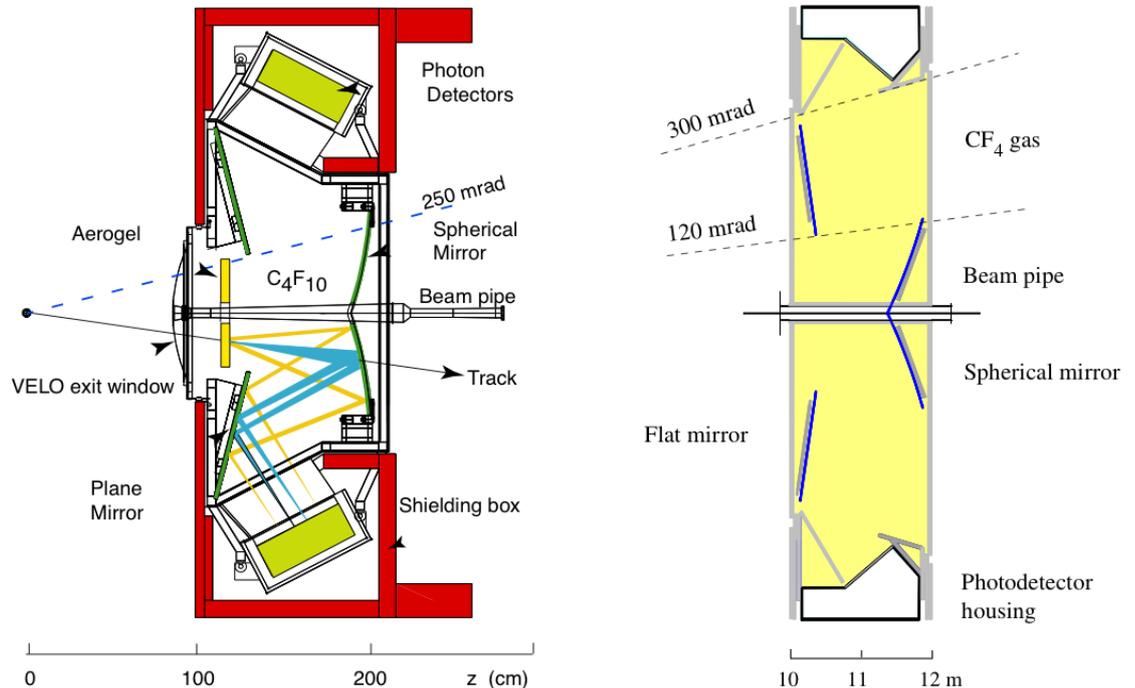
Figure 1.7: RICH1 detector side-view (left). RICH2 detector side-view (right). Note that the scale differs between the figures.

C. MUON SYSTEM

Since muons are present in the final states of many CP-sensitive B decays, muon triggering and muon offline identification are fundamental requirements of the LHCb experiment. The muon system [13]-[15] consists of five stations (M1-M5) situated downstream of the RICH2 detector (Figure 1.4). These stations, of rectangular shape, are placed around the beam pipe. The calorimeter system is positioned between M1 and M2. The detector technology used in the muon system is the radiation hard multi-wire proportional chamber (MWPC). This technology is used in all the stations but in the inner region of M1 where Triple-GEM detectors [15] are installed. Iron absorbers (80 cm) are interleaved between stations M2-M5 and behind M5. Together with the calorimeter system, they remove the hadronic background and shield the muon detector from particles that have been strayed from the LHC beam. The minimum momentum a muon must have to reach M5 is approximately 6 GeV/c. The muon system is used in the L0 trigger to select events with muons of high transverse momentum (p_T) and in the high-level trigger (HLT) and in the offline analysis for muon identification.

1.2.3 THE TRIGGER SYSTEM

The LHCb experiment will collect data at a luminosity in the interaction region of 2 x 10^{32} cm^{-2} s^{-1}, a factor 50 lower than the LHC design luminosity. The reasons for this were discussed in the section 1.1. This has two advantages. First of all, the event rate is mainly dominated by single interactions (Figure 1.8), which are easier to analyze, and secondly the radiation damage is lower.
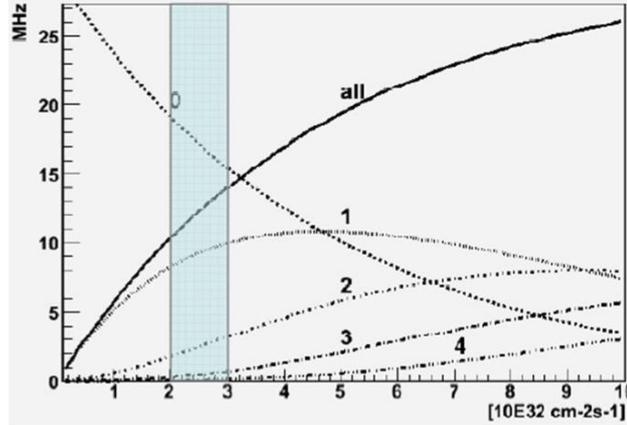
Figure 1.8: Multiple interaction rate per bunch crossing at the LHCb interaction point as a function of the Luminosity. The luminosity at LHCb is limited to 2 x 10^{32} cm^{-2} s^{-1} in order to have mainly single interactions.

The purpose of the trigger system of the LHCb experiment is to reduce the 40 MHZ bunch crossing rate to about 2kHz which is the rate at which the events can be stored for offline analysis. From Figure 1.8 we can see that the rate of events with proton-proton collisions at the nominal luminosity at LHCb is only about 10 MHz. Bearing in mind the fact that we need to reduce this rate to 2kHz because data storage limitations and the fact that the B decays interesting for new physics are very rare, we can appreciate the utmost importance of a reliable and very efficient trigger system for the performance of the LHCb experiment.

The LHCb trigger system is implemented in two levels: Level Zero (L0) and the High Level Trigger (HLT). The trigger scheme is illustrated in Figure 1.9.

A.   LEVEL 0

The L0 trigger operates at 40 MHz synchronous with the LHC clock. It uses custom made electronics to make a trigger decision using information from the calorimeters, the muon system and the pile-up veto detector. Due to the   large mass of the B-mesons, the decay products often have large p_T and E_T . The L0 decision is based on the observed values of the two highest p_T muons and the highest E_T hadron, electron and/or photon clusters. In addition the pile-up detector information is used to reject multiple interaction crossings. All this information is collected by the L0 Decision Unit that evaluates the data and checks whether one or more of the required criteria in the table in Figure 1.9 is matched. If that is the case, a signal is sent to the front-end electronics of all the sub-detector system, triggering them to be read out. The L0 latency, the total time between bunch crossings and the arrival of the L0 trigger signal to all the sub-detector front-end electronics, is fixed to 4 µs. The allowed output bandwidth of the L0 is 1.1 MHz.
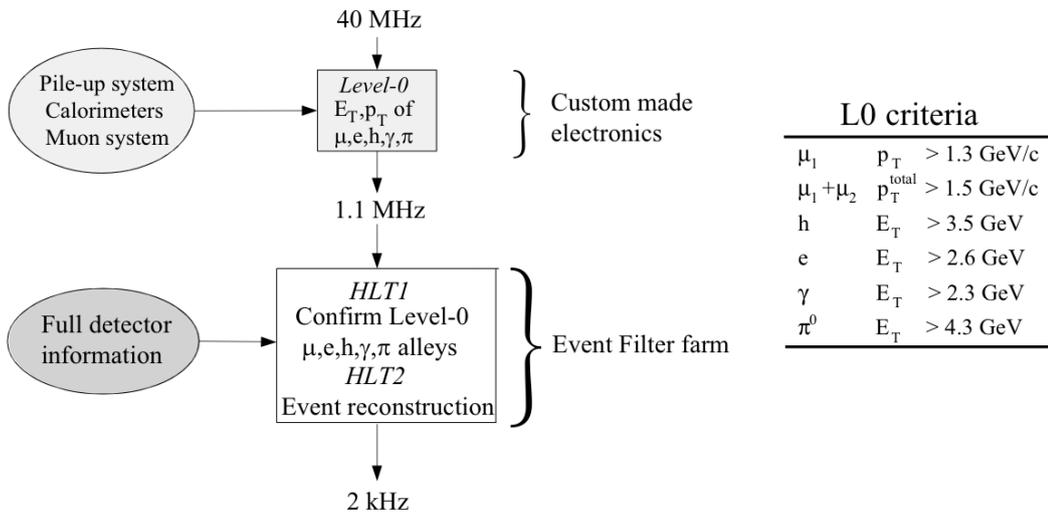
Figure 1.9: Left: Flow chart of the Trigger system reducing the 40 MHz bunch crossing rate to the 2kHz of the storage rate. Right: L0 trigger criteria for nominal luminosity.

B.   *HIGH LEVEL TRIGGER*

The High Level Trigger (HLT) is a computer program that runs on a farm of ~2000 CPUs. It has access to the full event information from all the sub-detectors. Its goal is to reduce the 1.1 MHz input rate from L0 to a 2kHz output rate at which events are stored for further offline analysis. The HLT has to stages in order to meet the time limitations imposed by the previous requirement. In the first stage (HLT1) the events are sorted into different "alleys". Each of them addresses one of the L0 trigger types. Depending on the L0 trigger type, events may enter multiple alleys. The HLT1 confirms the selection done by the L0 trigger by partially reconstructing the event further information from the VELO and the tracking system. The majority of the uninteresting events is rejected at this stage. The primary vertex position is reconstructed using 2D Rz-tracks in the VELO. The impact parameter of a track with respect to the primary vertex is a powerful discriminator for B events and used in many of the alleys. The HLT1 reduces the rate to approximately 30 kHz. The HLT2 then reduces the event rate to the demanded 2 kHz by performing a full event reconstruction with information from all the sub-detectors and makes the final trigger selection.

1.2.4 THE ONLINE SYSTEM

The main purpose of the LHCb online system is to monitor and control de data transmission from the front-end electronics to the final storage. The online system ensures that this data transfer is performed correctly and under controlled and known conditions. It supervises the timing of the whole detector in the sense of synchronizing all the sub-detectors with the LHC clock. The online system can be divided into three subsystems that run in parallel.

- The Data Acquisition (DAQ) system guarantees a safe data transfer from the sub-detector electronics to the final storage. L0 triggered data is sent from the front-end electronics to the LHCb back-end readout boards. After further processing the data is transferred to a CPU farm where the HLT selects the events that are finally stored in the CERN CASTOR facility. From there the raw data are distributed in quasi real time to six Tier-1 centers across Europe.
- The Timing and Fast Control (TFC) system plays an important in the data readout between the front-end electronics and the online processing farm since synchronizes trigger decisions and beam-synchronous commands to the LHC clock and orbit signal provided by the LHC.
- The Experiment Control System (ECS) monitors the systems described above and all the sub-detectors of the LHCb experiment. The ECS controls and monitors the DAQ system, the TFC system, the trigger system and as well the operational and environmental parameters of all the sub-detectors like HV and LV, pressures, gas flows, temperatures, radiation doses etc…

The DAQ and ECS parts are subdetector specific, so each one has to do a specific implementation. The way this has been done for the ST is summarized in [16].

CHAPTER 2

THE SILICON TRACKER ELECTRONICS OVERVIEW

ABSTRACT

This chapter provides an overall description of the ST electronics. It depicts the general scheme of the readout, control and other electronics equipment used in the ST. Some parts of the electronics that are not part of the scope of this thesis will be shown with a bit of detail herein, as it is the case of the TELL1. In addition the grounding, shielding and power distribution strategy is described. This information is essential to understand the subsequent chapters of the thesis where the electronics is described in a domain basis standpoint. The topological distribution of the system described next helps to understand how the entire system is like.

"Si haces lo que siempre has hecho, nunca llegarás más allá de donde siempre has llegado"

Anónimo

2.1 INTRODUCTION

Although the mechanics of the the IT and TT sub-detectors are completely different their electrical layout is largely the same. Unless stated explicitly, the system described in this chapter is identical for both IT and TT.

The Silicon Tracker of the LHCb experiment consists of silicon strip detectors with a pitch of around 200 µm. For the TT station a 500 µm thick sensor with 512 strips is used, while the three IT stations feature 320 and 410 µm thick sensors of 384 strips each. The signals are amplified and processed by the Beetle [17] readout chip, which transmits its data via differential analogue lines to the Service Boxes (SB). Here the signals are digitized and converted into optical to be transmitted via fiber to the pre-processing boards in the counting house, located up to 120 m away. From now on, we will call detector unit to the ensemble compound by one hybrid and the corresponding silicon sensors.

From a physical standpoint the detector equipment is placed both in the experimental area (cavern) and in the barracks (the so-called counting room behind the concrete shielding wall) as sketched in Fig. 2.1. The electronic elements located in the cavern are the service boxes and the detector, in addition to the MARATON low voltage power bins. All the devices in the cavern need to withstand both the radiation and magnetic fields levels present there. Part of the employed electronic devices were explicitly developed for that purpose while others are commercial chips which have been rated for radiation tolerance in several irradiation campaigns [18]-[21]. In the barrack are placed all the detector elements that are not rated for radiation tolerance as well as the ones that may need human accessibility when the experiment is running.
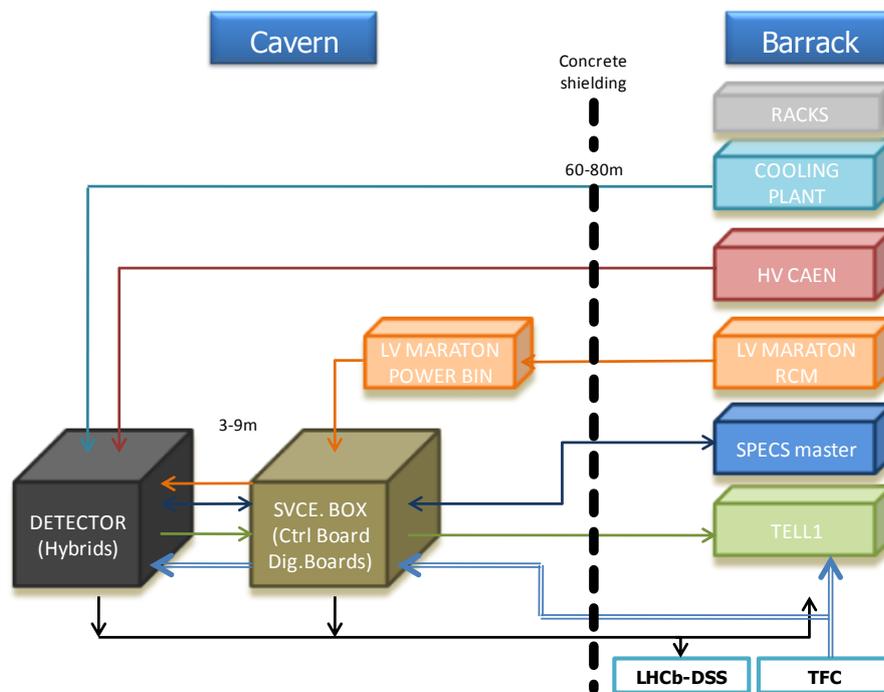
Figure 2.1: Silicon Tracker equipment distribution.

The data acquisition electronics placed in the cavern is what we call 'L0 Electronics' (level 0), while the pre-processing boards located in the counting room are known as the 'L1 Electronics' (level 1). The 'L1 Electronics' name comes from the fact that in the past the L1 trigger was implemented in these boards. This is not the case anymore since there is not event selection in the 'L1 electronics'. The data pass through to the DAQ network after some processing in the L1 boards. These boards receive the data from the L0 FE electronics and perform a first basic verification of the collected data. The accepted event data are zero-suppressed and formatted to be sent to the DAQ system. A sketch of the readout link chain is shown in the Figure 2.2 and described in detail in [22]. In the picture we can see on one end the Beetles which through the DB electronics placed in the SB send the data to the TELL1 boards [23] ('L1 electronics') in the counting room.

Besides L0 and L1 electronics, some electronic infrastructure such as the Low Voltage (LV) and High Voltage (HV) power supplies are essential to power the systems. Although it could look a trivial task, the proper selection, distribution and installation of the power systems is a crucial for the proper working of the

complete detector. This issue will be treated later on in this chapter in the power distribution, grounding and shielding section.
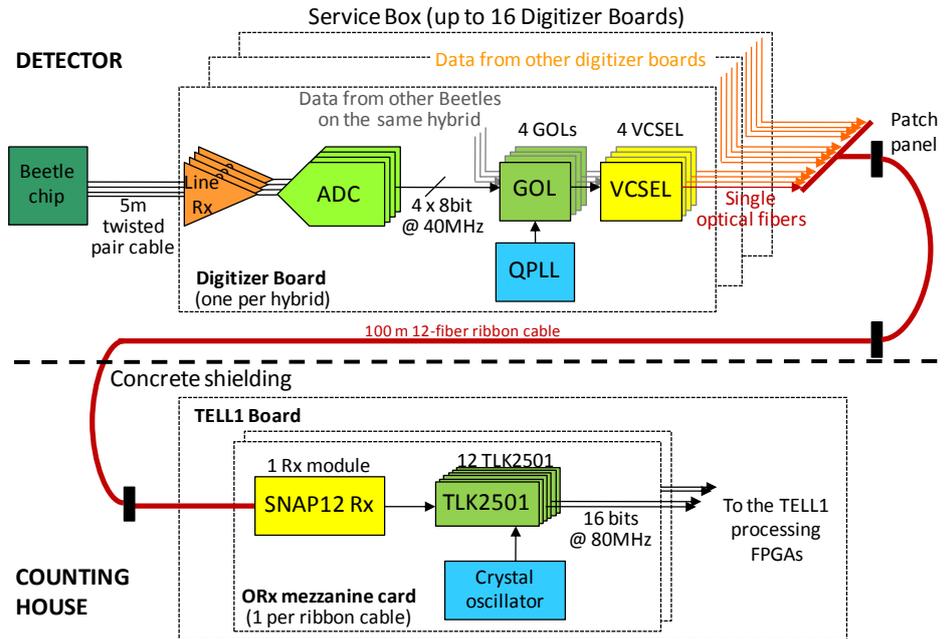
Figure 2.2: Overview of the readout link.

The IT detector consists of three identical stations (T1-T3). A front view of one entire station is shown in Fig. 2.3. It consists of two retractable support frames (A-side and C-side), each of which carries two detector boxes. Each detector box contains 28 detector units arranged in four layers as *x-u-v-x*, i.e., the sensors of two *x* layers are vertically orientated whereas the *u* and *v* are tilted under a stereo angle of ±5°. Four Service Boxes, two for each detector box, are fixed to the lower end of each support frame. They move with the frame when it is retracted. Cables between the detector boxes and the SBs are routed along the detector frame. A flexible cable chain guides cables and fibers from the SBs to the LHCb bunker.

A sketch of the TT is shown in Fig. 2.4. It consists of 280 detector units that are contained in one large detector volume and arranged in four layers (*x-u-v-x*). This detector volume is made out of two half-boxes (A-side and C-side) that can be retracted individually from the beam pipe. Detectors are read out from the top and from the bottom of each half-box, i.e. the readout hybrids are placed out of the experiment acceptance. This defines four quadrants (A-top, A-bottom, C-top, C-bottom) over which the detectors are equally distributed. For each quadrant there is a stack of six SBs that are mounted against the front-face of the LHCb dipole magnet. Flexible cable chains guide the cables from the four corners of the detector box to the corresponding stack of SBs.

The detectors are cooled using liquid C_6F_{14}, which is electrically not conductive. The major source of heat in the SBs are the linear power regulators on the backplane. A brass cooling plate, which is connected to the general LHCb mixed water cooling system is mounted directly on top of these regulators. A separate linear power regulator on the Control Board is cooled using a dedicated brass heat-sink connected to the same mixed-water circuit.

The Detector Safety System (DSS) [24] is a separate reliable and fast real-time response system which protects the equipment and prevents situations leading to severe alarms. The DSS is a parallel hardware oriented system that will take the appropriate action in real-time to protect the hardware against potential causes of damage, usually by powering off the equipment. The DSS checks a few critical parameters with dedicated sensors: temperature, water leaks, smoke sniffers, cooling status, gas status and cooling water flow switches. The logic output of all these sensors is routed to a Programmable Logic Controller, which runs a closed-loop control cycle. For example, the heat-sink on the SB backplane is protected by a thermo-switch which is connected to the LHCb Detector Safety System (DSS) and opens at 60 ºC.
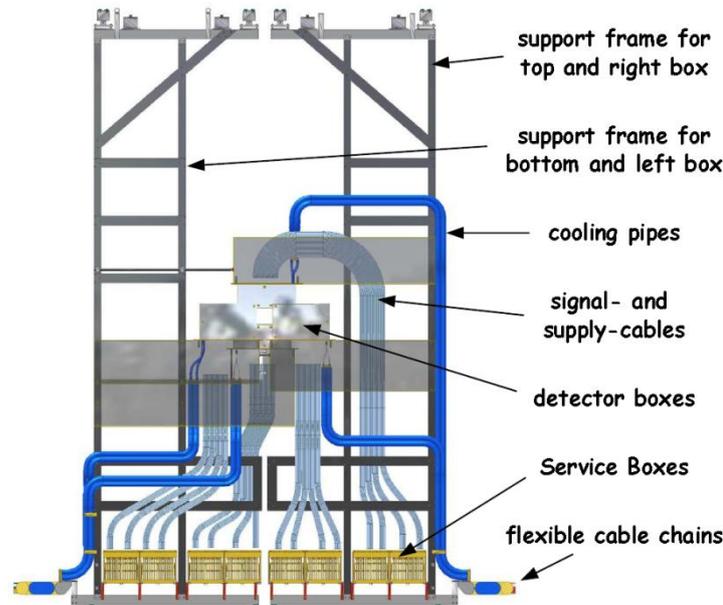
Figure 2.3: Sketch of one IT station.
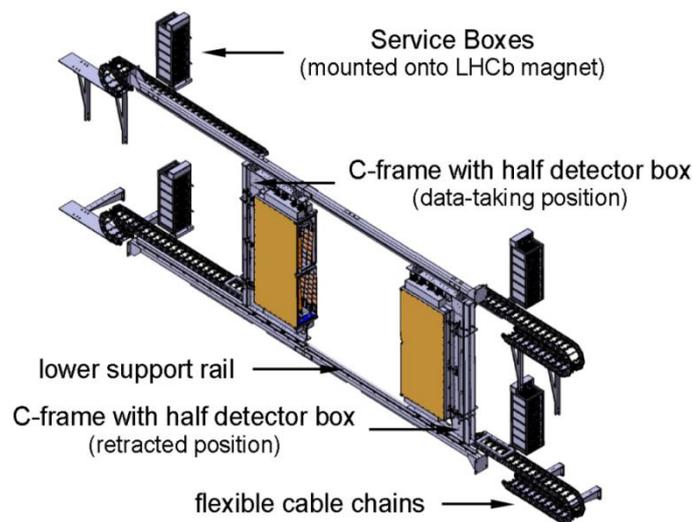
Figure 2.4: View of the TT station. The C-side is shown retracted from the beam pipe.

2.2 THE L0 ELECTRONICS

The L0 Front-End (FE) electronics must correctly capture and store detector signals, from particles generated by the bunch collisions of the LHC machine on a large number channels, ~272 000 in the ST.

The detectors boxes house the readout units, in other words, the silicon micro-strip sensors and a FE readout hybrid which accommodates either three (IT) or four (TT) Beetle chips. The Beetle chips are mounted together with passive electronic components on flexible printed circuit boards.

The detector signals from the silicon sensor are captured with sufficient time resolution by the Beetle chips. Captured data are stored in the Beetle pipeline buffer, until the Level 0 (L0) trigger decision accepts or rejects data from a given bunch crossing. Accepted data must be extracted from the pipeline buffer and temporarily stored in the L0 derandomizer buffer (also implemented within the Beetle chip), waiting to be transferred to the Digitizer Board (DB) and then to the TELL1.

The analogue data from the Beetle chips are sent via differential lines on up to 9 m long twisted pair cables to the SBs located outside the detector acceptance. In the SB, the data are then digitized by the DB with 8-bit resolution, multiplexed and converted to optical. The digital-optical signal is further transmitted via fibers of

up to 120 m length to the counting house to an optical receiver card, which is placed in the TELL1 board. It important to note and remember for the rest of the thesis that one hybrid matches always to a DB.

The Control Board (CB) located in the service box provides the necessary functionalities to carry out several control and monitoring tasks, namely: the slow and fast control of the ST readout and digitizing electronics; control and monitoring of the LV power regulators; and the monitoring of the detector box temperatures and humidity.

The Digitizer Boards and Control Board are interconnected together in the Service Box by means of a custom developed backplane, which also houses the LV Regulators (LVR).

2.2.1   HYBRID AND SENSORS

As stated before, a detector unit consists of a hybrid and the corresponding silicon sensors. The hybrid packages for the IT and TT detectors consist of the Beetle chips mounted together with passive electronic components on a flexible printed circuit board (PCB) and of ceramic pitch adapters to match the silicon sensor pitch to the pitch of the Beetle input pads. The complete package is attached to heat spreader substrates that provide a low thermal impedance path to further cooling plates. The sensors are mounted in carbon fiber supports and there are different sizes of detector units in counts of sensors, ranging from 1 to 4 (Figure 2.5). Ultrasonic aluminum wire bonding is performed for the Beetle to hybrid, Beetle to pitch adapter, pitch adapter to sensor and sensor to sensor bonding.
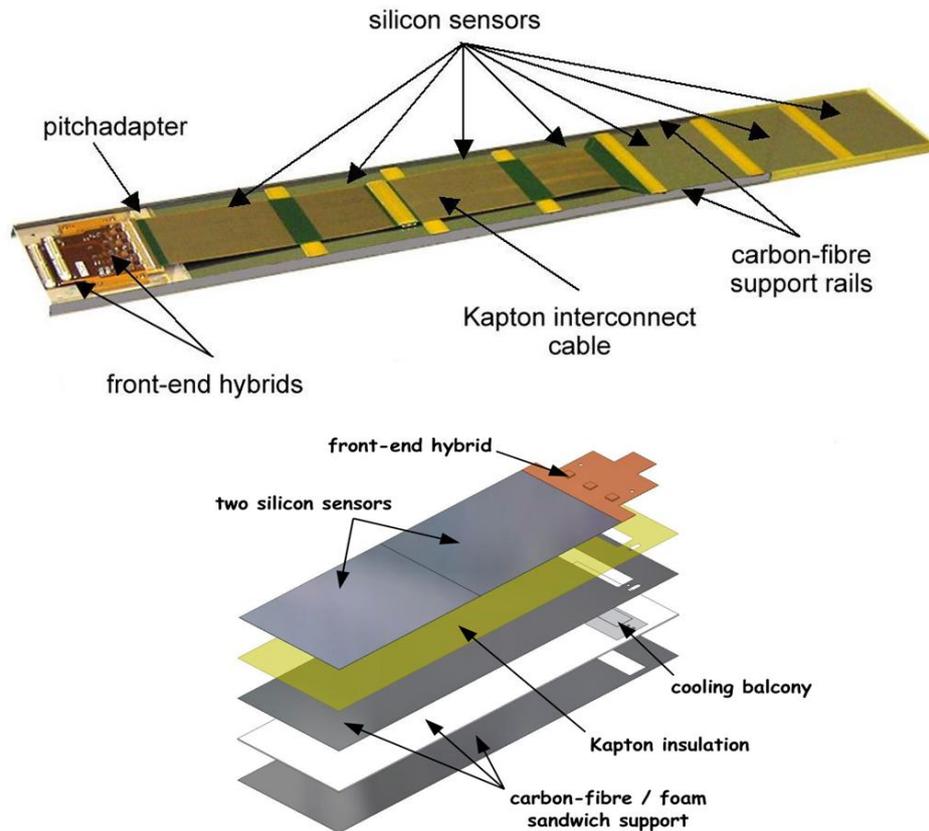
Figure 2.5: The TT (top) and IT (bottom) detector modules schematics.

The hybrid for the IT detector module (Figure 2.6) has a ceramic pitch adapter that is attached to the component loaded flex circuit. This assembly is then glued with silver epoxy to an aluminum balcony providing cooling for electronics and a low impedance connection to a common ground. Wire bonds connect then the 384 silicon strips of the detector module via the pitch adapter to the three Beetle chips.
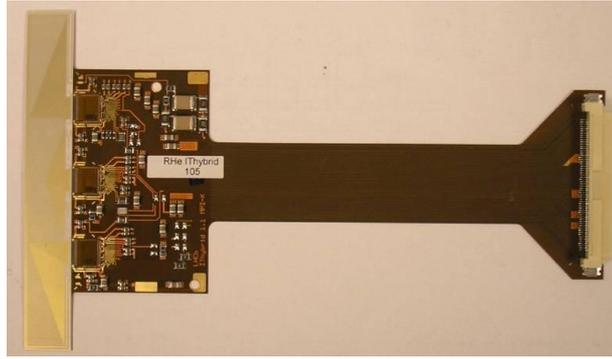
Figure 2.6: IT hybrid, flexible PCB and pitch adapter.

The TT silicon detector modules are 160 cm long silicon ladders that are longitudinally segmented in four or six independent readout units. The hybrid packages are located at both ends of a detector module. Therefore, a stack of up to three hybrids at each end is necessary for the readout. The stacked hybrid design of the TT detector requires the use of three different hybrid subtypes each having four Beetle chips. The hybrids for the TT detector differ in the overall length and in size and material of the heat spreaders (Figure 2.7).

Figure 2.7: TT hybrid, flexible PCBs and pitch adapters.

2.2.2   THE DETECTOR BOX

The detector boxes for the IT and TT provide electrical and thermal insulation. The detector units are housed inside the detector box (Figure 2.8). Low voltage and bias voltage as well as detector signals, control signals and monitoring signals are fed through the wall of the detector box using custom-designed polyimide cables (TT) or FR4 printed-circuit boards (IT). Photographs of these feed-through are shown in Figure 2.9.

Figure 2.8: View of the TT installed in LHCb (left) with one boxes open and the sensors inside. View of the first IT station (bottom right), and detail of one of the detector boxes in semi-open position (top right).

In the IT detector box there is a feed-through PCB per detector layer. This printed circuit board carries seven hybrid tail connectors inside the box, seven VHDCI connectors and one high density Dsub for the bias voltage outside of the box. In the TT there is one detector patch panel per half module. This patch-panel houses the cable adapter boards, which interconnects the polyimide feed-through cables coming from inside the box to the cables that go to the service box with VHDCI and Dsub connectors.

Custom-made shielded copper cables transport power and signals from the detector boxes to the Service Boxes that are located at a distance of a few meters from the detectors.
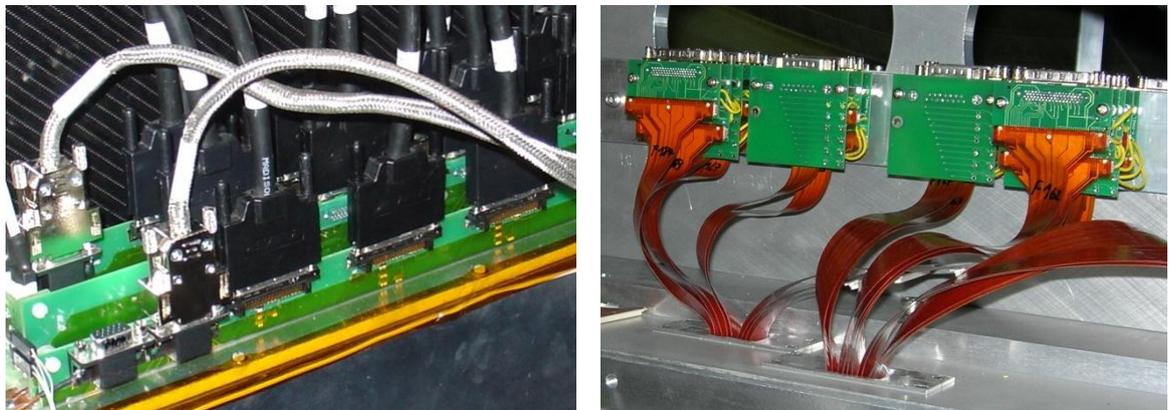
Figure 2.9: Photographs of the feed-through PCBs for the IT (left) and the polyimide feed-through cables for the TT (right).

2.2.3   THE SERVICE BOX

While the silicon sensors and the hybrids are located in the detector boxes, the digitizing and control electronics is located in the Service Boxes, out of the detector acceptance. This reduces the amount of dead material and hence multiple scattering and generation of secondary particles. In addition, the location of the SBs, features lower radiation levels and easier access for cooling and maintenance.

The SBs are custom-made crates that contain up to sixteen (IT) or twelve (TT) Digitizer Boards for the digitization and optical transmission of the detector data [22], a Control Board [25] providing interfaces to TFC [26] and ECS [27] and a custom-made backplane for the low voltage power regulation and distribution of TFC signals and other control signals. Each DB services one front-end readout hybrid. One SB therefore corresponds to up to 48 Beetle chips (up to 16×3 for IT and up to 12×4 for TT). The power for the Beetle chips (2.5 V) is filtered and regulated on the SB backplane and provided to the front-end hybrid via the corresponding DB. In addition, low-voltage levels of 2.5 V, 3.3 V and 5 V are provided to power the various electronics components on the CB and the DB themselves. From the SBs, commercial copper cables and optical fibers lead to the bunker and the LHCb counting room (in the barrack) where the power supplies,

TELL1 readout electronics, and master control electronics are located. There are some patch-panels for all these cables some meters away from the SB. An overview scheme and two finally installed SBs are shown in the figure 2.10.
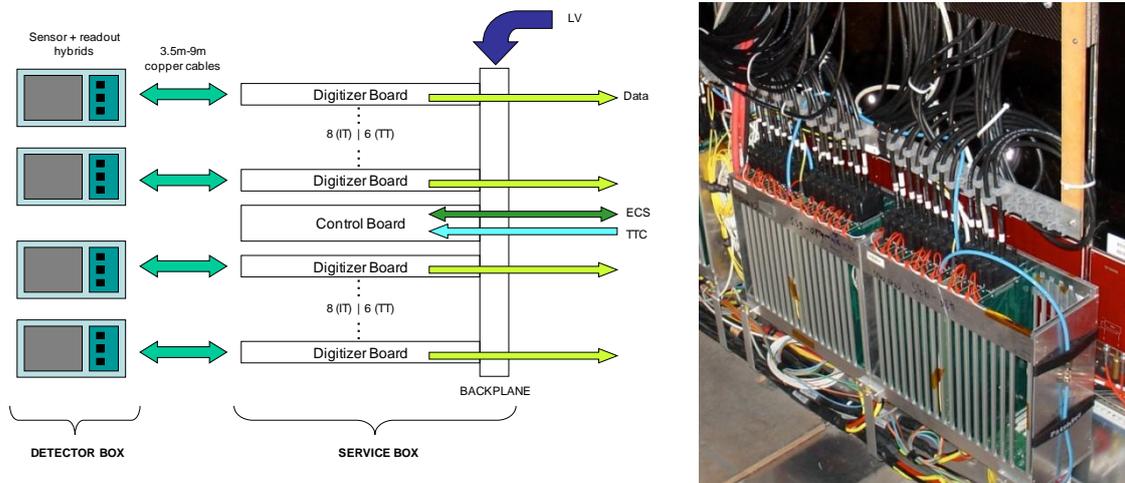
Figure 2.10: Service Box scheme (left) and two installed IT service boxes (right).

2.2.4 DIGITIZER BOARD

Any Beetle readout hybrid is connected with a 68-wire twisted-pair round cable of up to 9 m of length to a Digitizer Board.

The Figure 2.2 already showed a simplified scheme of it. First, a line receiver converts the analogue detector signal from differential to single-ended. The line receiver suppresses the common mode introduced in the cable and matches the signal amplitude to the input of the 8-bit analogue to digital converter (ADC).

Data of four ADCs, corresponding to the four analogue output ports of one Beetle are encoded by the CERN GOL ASIC [28] into the Gigabit Ethernet protocol and modulated onto a 850 nm VCSEL diode. The digital data is framed by using the *DataValid* signal of the Beetle readout hybrid to drive the *TX_EN* line of the GOL serializer. The GOL serializer provides different operational modes, some of them programmable via I^2C, others to be determined by pulling special pins to either ground or power. While some of the hardwired parameters are completely fixed, other values, for example the PLL charge pump current and the mean laser diode current, can be reprogrammed via the I^2C interface. For these parameters, the hardwired value represents the power-up default, which is set before any user intervention.

An anti-fuse FPGA from 'Actel Inc.' was used to form a 7-clockcycle triple-redundant shift register. This re-aligns the *DataValid* signal to the digitized data and allows for automatic re-synchronization of the optical link between physics triggers.

A QPLL [29] provides clock jitter filtering, which is mandatory to ensure the proper functioning of the *GOL* transmitter.

In addition, there is also and a slow control 6-channel ADC. This ADC is the CERN DCU25F chip [30] which is used for reading the hybrid temperature, the QPLL status signal and LV regulators flags. This device is also controlled via I^2C.

Two versions of Digitizer Boards have been designed: a TT version (Figure 2.11) with 16 ADCs corresponding to a 4-Beetle TT hybrid and an IT version with 12 ADCs only to match a 3-Beetle IT hybrid.
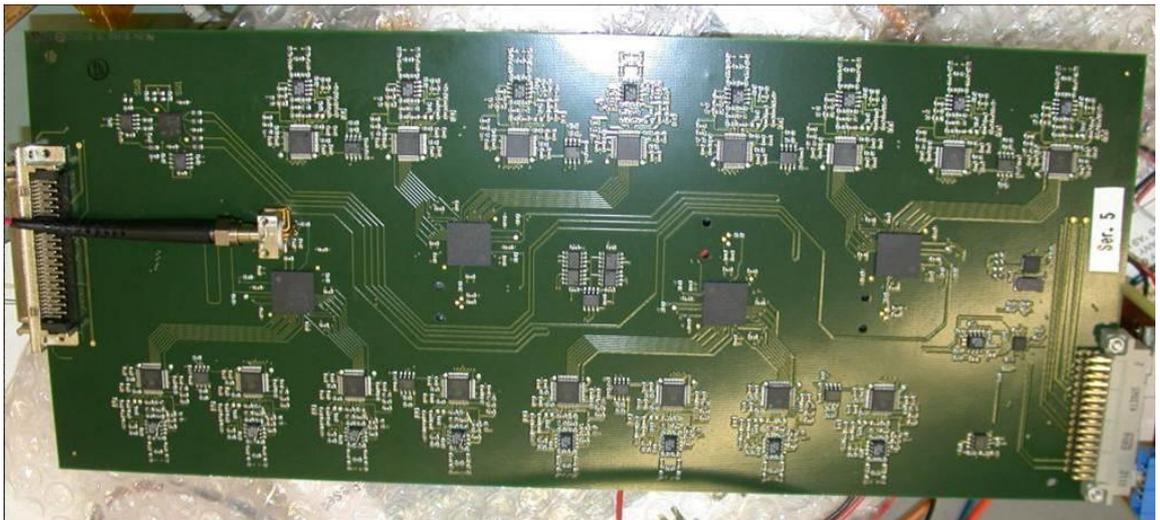
Figure 2.11: TT Digitizer Board picture.

2.2.5 CONTROL BOARD

This card provides the necessary functionalities to carry out several control and monitoring tasks, namely: the slow and fast control of the ST readout and digitizing electronics; the control and monitoring of the LV power regulators; and the monitoring of the detector box temperatures and humidity. It connects to the ECS using the SPECS field-bus system [31].

As the structure of the IT and TT detectors is very similar but with a different FE readout electronics grouping (3 Beetles per hybrid for the IT and 4 for the TT) a common control board has been designed for both detectors. A detailed description of the Control Board is given in chapter 3.
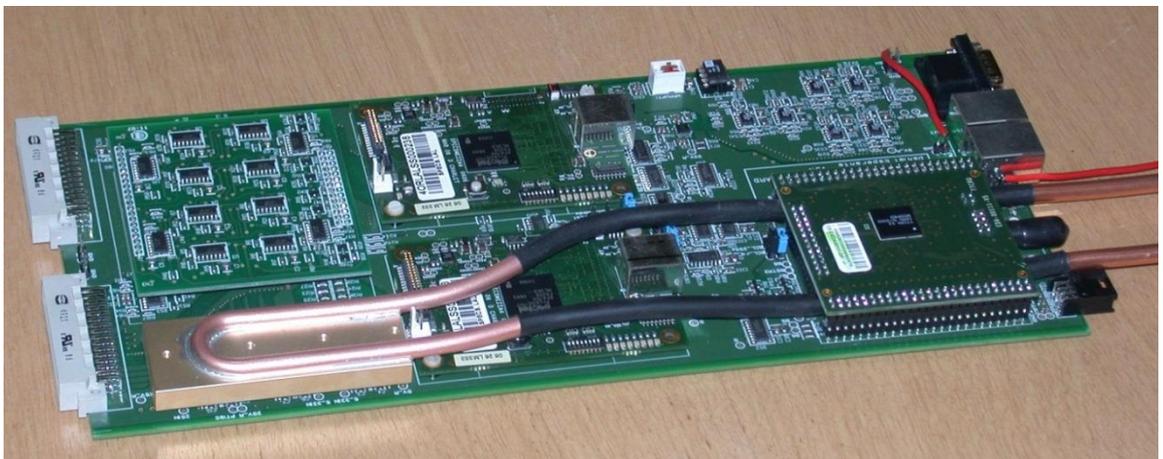
Figure 2.12: Silicon Tracker Control Board.

2.2.6 THE BACKPLANE

The Digitizer Boards are connected to the Control Board via Service Box backplane. This is the board that makes possible the use of the same CB for IT and TT despite the DBs and hybrids are different. The IT and TT versions of the backplane are different, but the signals pin-out in the connectors where the CB and DBs plug is the same. The difference is the allocation of DBs, 16 in the IT and 12 in the TT. The distribution of timing and fast trigger signals is done via impedance controlled differential traces. Equal trace lengths were maintained to keep the time skew of the fast signals among different backplane slots minimal. For each slot, three positive linear voltage regulators provide the required voltages to the Beetle front-end hybrids, the DBs and CB. As all regulators are located close to each other, a common copper water cooled heat-sink is used to dissipate the heat produced by the regulators.

The TTC (Timing and Trigger Control) signal distribution consists of the clock trees for the ADC/GOL devices and the Beetles, the *L0accept* (sometimes called trigger), the *L0reset* and the *testpulse* calibration signal. As the timing of all of these signals is critical, LVDS is used throughout the backplane. With the CB being the only signal source and the up to 16 DBs being the signal receivers, a strict tree structure was designed, with exactly one receiver per differential line. To fan out the single signal sources to multiple receivers, quad differential LVDS drivers and receivers which have been rated for radiation tolerance were used. A 1:4 fan-out for the TTC signals is shown in the Figure 2.13. A similar clock tree distributes the 40 MHz ADC/GOL clock, whose phase is independently adjusted by the TTCrx (Timing, Trigger and Control receiver) [32] in the CB. By concatenating two of these fan-out designs, each backplane is able to provide up to 16 DBs with fast TTC Signals. By placing the CB in the center slot of the backplane, trace length differences can be minimized and all the signals routed in the backplane with equal length so the signals' phase is the same on all 16 DBs. This is critical since no option is foreseen for the user to fine-tune timing relations between different DBs. The sampling point for the whole group of read-out units belonging to the same SB can be adjusted using the clock delay lines in the TTCrx.

In addition to the TTC signals, the CB also provides slow control signals to the DBs. These are the I^{2}C buses to the readout hybrids, the GOL devices and the DCU25F on each Digitizer Board, and two reset lines which can be controlled by the CB: one for the GOL and one for the QPLL.

The I^{2}C addressing scheme contains two bits, labeled 'Board-ID', which assign part of the I^{2}C addresses to the bus devices. These two bits are set by plugging a DB into a slot of the backplane, as the corresponding pins of the backplane connector are hardwired to ensure a unique address for each I^{2}C bus.
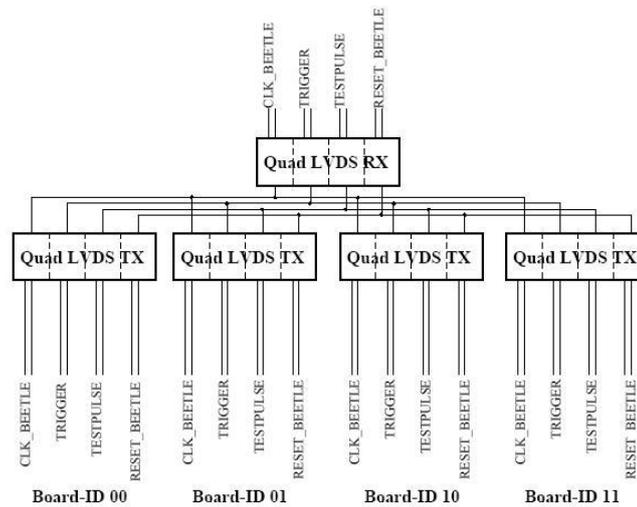
Figure 2.13: TTC signals fan-out tree for the Beetles.

The L4913 positive voltage regulator [33], developed by the CERN microelectronics group in collaboration with 'ST Microelectronics', is used for the hybrids and DBs low voltage regulation. They are designed to withstand high radiation levels. The L4913 supports the use of sense lines for voltage regulation directly at the load. While the regulator is rated for a maximum current of 3.2 A, a lower current limit can be set with an external resistor. In case the limit itself is reached, the regulator will signal this by pulling its *OCM* pins low. An *INHIBIT* pin allows shutting down the voltage regulator with a logic signal.

In the ST on-detector electronics, voltages of 2.5 V (all deep-submicron devices), 3.3 V (DB clock distribution and other DB and CB elements) and 5 V (DB clock distribution, line receivers and TTCrx photodiode) are needed. The partitioning of the loads has not only to follow a logic grouping, but has to take in account the maximum current that can be regulated by a single voltage regulator. The detailed partitioning of low voltage regulators is described in chapter 4. It must be noted that the partitioning follows the same grouping as the I^{2}C and other "services" as the HV, the DAQ readout and the clock distribution.

2.3 THE L1 ELECTRONICS

The 'L1 electronics' is located in the counting house. Its purpose is to handle L0 accepted data before transmission to the DAQ network. The data are transferred by optical lines from the service boxes to the counting house. There, the TELL1 real-time processing boards perform the deserialization, data processing

and transmission of the compressed data to the L1 and HLT (High Level Trigger) trigger CPU farm. They are placed behind a shielding wall, 60 m from the interaction point to minimize the amount of electronics exposed to radiation.

To preserve synchronization over large distances LHCb has a global clock and fast signal transmission network, the TFC system. The TFC is driven by the Readout Supervisor [34], responsible for distributing the LHC clock and scheduling the trigger decisions.

The TELL1 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 of the TELL1 are five large FPGAs which can handle ~60 Goperations·s^{-1} per board.

Figure 2.14 shows a simplified block diagram of the ST TELL1. Two optical receiver (ORx) mezzanine cards provide an input for 24 optical links from the SBs. 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 *SynchLink-FPGA* receives the fast signaling from the TFC system via TTCrx chip and distributes the synchronization signals (clock, *L0accept* 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, and to perform Ethernet and IP framing before transmission to the readout network. The readout network is interfaced by a four-port Gigabit Ethernet card (GBE) delivering 4 Gbit·s^{-1} data bandwidth. The fast control signals received via TTCrx are the LHC clock (locked at 40.079 MHz), the *L0reset*, the *L0accept* and the data IP destination address. A Credit-Card sized PC (CCPC) mezzanine board is used to provide an interface for local board control via Ethernet and hence connect with the ECS [35].
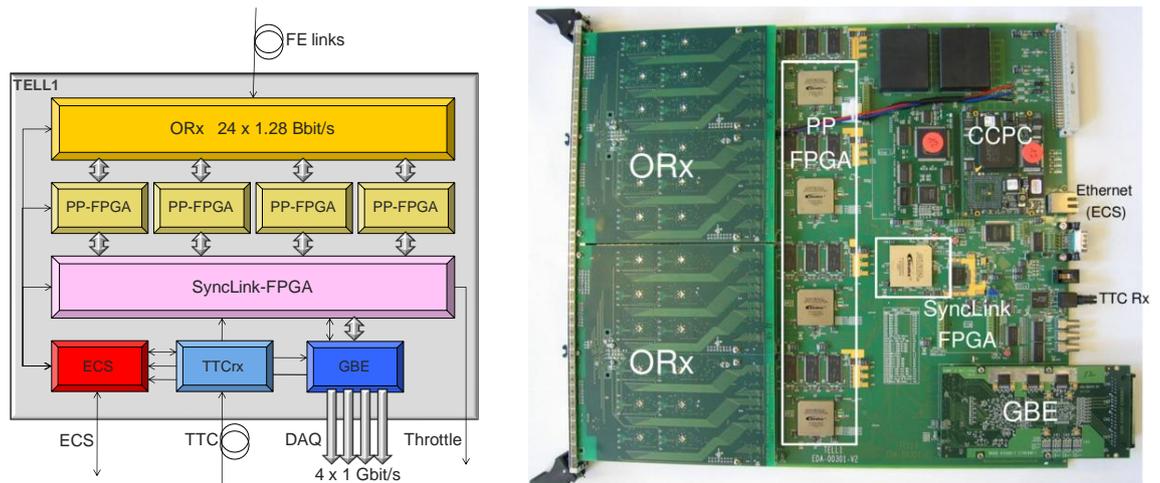
Figure 2.14: TELL1's simplified block diagram (left) and TELL1 board (right).

In addition to the zero suppressed data, the TELL1 can also send the raw data to the network. The maximum achievable data throughput with non zero suppressed data is of some kHz. The raw data is valuable to perform self-tests in the TELL1 and check regularly that the algorithms behave as expected.

2.4 GROUNDING, SHIELDING AND POWER DISTRIBUTION

The careful design of the grounding, shielding and power distribution strategy is essential in big particle detector systems [36]. These detectors have two features, which are in conflict with respect to careful electronic design: the primary detector signals to be amplified and recorded are very small, often not exceeding a few thousand electrons. At the same time the detectors cover large surfaces and are therefore vulnerable to external electromagnetic fields introducing pickup currents into the systems, which the primary detector signal circuit is most sensitive to. In addition ground currents from large scale power supply networks may sneak into the signal path and create additional noise.

The grounding strategy for LHCb has been defined in the technical note [37]. It is based on the philosophy of a low impedance mesh structure, which is certainly advantageous compared to a star coupling system. In this section we will briefly report about the Silicon Tracker grounding strategy, which is described in detail in [38].

We will first list the inventory or relevant items for grounding considerations. Then, we will shortly explain the sensitive detector input signal current circuits and finally, describe the grounding, power distribution and line filtering measures applied to each of the electrical units of the LHCb silicon tracking system.

2.4.1 RELEVANT COMPONENTS

The relevant components of the Silicon Tracker and their location are:

- The detector boxes: one big box made of two retractable independent parts in the TT and three stations consisting of four mechanical independent detector boxes each in the IT. The FE electronics is placed inside.
- The SBs, containing the data digitization, optical transmission, control electronics and the low voltage regulators located just outside of the acceptance of the experiment. For the TT station six service boxes for each detector quadrant are fixed to the TT support structure on the left and right side of the station. For the IT stations there are two service boxes for each detector box located at the lower end of the IT support structure (In total 24 service boxes for TT, 24 service boxes for IT).
- The MARATON low voltage power system, located partly in the cavern and partly in the counting house.
- TT and IT cavern HV patch panels for bias voltage distribution located both in the cavern, very close to the service boxes and in the barrack.
- Bias voltage (HV) supplies, located in the barrack.
- Both the IT and the TT detector boxes have a connector area for signal and power lines attached to them, the so called "feed-throughs".
- The data readout and the TTC signals (clock, trigger, reset, testpulse) are transmitted by optical fiber and are thus not relevant for grounding considerations. However the ECS data transmission is by "copper" (LVDS). Careful design of the CBs of the services boxes has to ensure, that no high frequency noise is picked up and no ground currents can occur on these signal lines.

2.4.2 DETECTOR SIGNAL CURRENT PATH

The first step in grounding and shielding system design for sensitive physics detectors is to identify the location of the small signal path. Most noise problems emerge from unwanted current or voltage sources within the primary signal current circuit. There are two main effects:

1. External electromagnetic fields can generate voltages or currents in the signal circuit. To minimize this so-called pickup noise, excellent shielding is required and the geometrical area surrounded by the input current circuit should be as small as possible.
2. Supply currents or other ground currents may produce variable voltage drops along conductors, which are part of the input current circuit and therefore add as noise to the signal. To avoid this, any supply and other ground currents have to be kept away from the input signal and thus make the small signal ground as low resistance and low inductance as practical. The unavoidable detector bias voltage supply should be well filtered before connecting to the input circuit.

Figure 2.15 shows the essential elements of the input circuit, the silicon sensor and the FE hybrid. The signal is generated from charged particles traversing the reverse biased silicon sensor diode. The signal is AC coupled to the readout strips, transmitted through a strip line cable (maximum length 58 cm, TT only) and amplified by the 128 channel Beetle chip. The reference input of the preamplifier is internally connected to the GND pins of the beetle. The signal current loop is closed through the bias voltage line, which in case of TT is running on a kapton strip line to the backside of the sensor. Thus, the bias voltage line has to be AC connected to the ground line of the beetle to close the input signal current loop.

The bias voltage cable will be fixed directly on the backplane of the sensors, thus the pickup area is as small as possible (point 1 from above).

The bias voltage low pass RC filter is located on the hybrids and has a characteristic frequency of $1/RC = 5$ kHz, which is rather high, the noise rejection is thus not very good. However, the capacitor is limited by its nominal voltage and geometrical size and the $1K0 \, \Omega$ resistor is limited by the expected bias currents of the sensors (up to 10 mA after irradiation). This $1K0 \, \Omega$ resistor also minimizes the amount of current that many circulate through the ground loop created in the HV return ('*DetBiasReturn*') because of the splitting of this line in the HV patch-panels in the barrack.
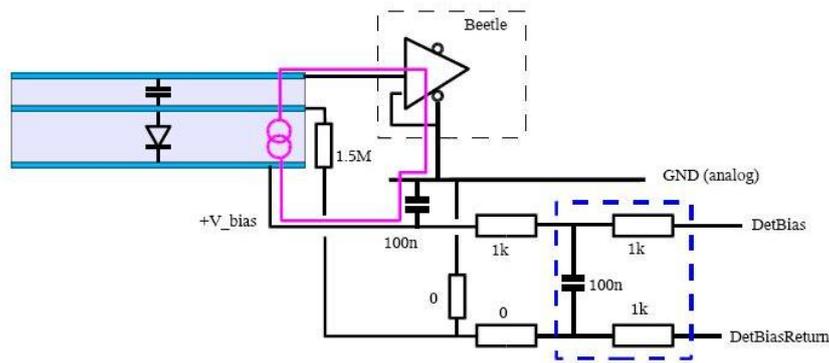
Figure 2.15: Essential elements of the silicon detector input circuit. The pink circuit shows the flow of the input signal current. The blue dashed rectangle on the right marks the bias voltage filtering, located on the front-end hybrids.

2.4.3   DETECTOR BOXES, THE FARADAY CAGES

To avoid excessive pick up noise, the detector boxes are layered out as double wall Faraday cages. Electric engineering rules tell us that optimal shielding results are achieved if the Faraday cage is as tight as possible and all electrical lines (signal, power, ground) leave the cage at one single point in space. This is done in the feed-through PCBs for the IT and on cable adapter boards for the TT (see Figure 2.9). The shielding of the cables is connected to the Faraday cage's outer shield at the same point with a low impedance connection. Furthermore all conducting pieces within the box are well connected to the inner ground shield. Both the IT and TT detector boxes are covered with 20 μm Aluminum foils. The inner shield is connected to the cooling rods (IT) and plates (TT) respectively, which are defined as the main detector ground. The carbon fiber pieces of the ladders as well as the bias voltage pads of the sensors are directly connected to the hybrid analog and power ground, which in turn connects through the cooling balconies to the cooling rod and plate, respectively (see sketch of Figure 2.16). The outer shielding of the detector box is connected via the feed-through (IT) and adapter boards (TT) to the cable shields.

2.4.4   LOW VOLTAGE AND HIGH VOLTAGE

Power supplies for sensitive analog front-end electronics is a delicate point that requires significant attention in large scale systems. Power supplies must be used in a fashion compatible to the defined grounding scheme (the configuration of power supplies is strongly linked to general grounding). Analog front-end electronics is in many cases extremely sensitive to noise in the power supply because of the single ended nature of detector signals.

The question of the location of the power supply units and the related transfer of power on cables often has significant problems. Low frequency voltage drops on cables can be compensated, to a large extent, by the use of remote sensing to compensate for cable losses. On the other hand, remote sensing can also pose stability problems as the compensation loop must be stable under large load variations and with different cable characteristics. In special cases the remote sensing circuit must be specially adapted to the final working conditions. High frequency components can only be handled with the use of local decoupling capacitors.

In most cases a difficult choice must be made between using normal commercial power supplies in the counting room with long cables (~80 m) to the FE electronics in the detector, or placing the power supplies in the cavern close to the experiment but then having to deal with radiation and magnetic fields. Power supplies for the use in the cavern must be specially designed to handle the radiation effects and the magnetic fields. For the main LHC experiments only few companies are being considered to be capable of supplying power supplies that can handle the environments of the experimental caverns. The Silicon Tracker group has decided to use the MARATON (Magnet field- and Radiation Tolerant New Power Supply System) power supplies from 'Wiener' which will be located in the cavern.

A special set of radiation tolerant linear regulators have been developed for the LHC experiments. The Silicon Tracker uses intensively the L4913 regulators for power regulation and distribution in the SBs.

For the sensors bias voltage (we call it high voltage although it cannot be considered as such since it goes only up to 500V), a standard (non radiation resistant) solution from 'CAEN' has been adopted. These systems are widely used in high energy physics and have proven to be very reliable.

A set of general points to be aware of in the design of the power distribution and grounding and shielding scheme are:

- Prevent ground loops via power supplies whenever possible.
- Use floating low voltage power supplies.
- Use twisted power cables to minimize noise pickup/generation.
- Use remote sensing to compensate for voltage drops on cables.
- Remote sense must be fully differential to minimize noise pick up.
- Remote sense must be capable of compensation for voltage drop in power supply return (known problem with radiation tolerant linear regulators from ST).
- Take special care of HV and its distribution.
- No single point failure must generate safety hazards (e.g. HV connected to accessible structure).
- Separate if possible power supplies for analog front-end from power supplies for digital logic.
- Sudden power consumption drops may generate serious over-voltage on load. Extensive local decoupling and over voltage protection may be needed.
- Magnetic coils/transformers and cooling fans in power supplies are sensitive to magnetic fields.
- Power supplies are known to be sensitive to single event burn-out caused by hadrons.
- Switching mode power supplies are known in some cases to generate significant common mode noise.

2.4.5   LOW VOLTAGE - MARATON

The LV power supply system is kept completely floating, only in the detector box the main grounding point will be connected to the general LHCb ground.

Both for the digital and analog power of the hybrid 2.5 V are needed. Two separate radiation hard regulators, type L4913 (developed by the CERN micro-electronics group) are located on the backplane of the service box and will provide the voltages. Sense lines for both the voltage and ground lines are used. This setup has been extensively tested [39].

Analog and digital grounds are connected together on the backplane of the service box, but they are kept floating and not connected to safety ground at this point. Although both grounds are in principle kept separate on the cable as well as on the hybrid layout, they are again connected together on the hybrid at one point to avoid problems due to different voltage drops on the two ground lines.

The power lines for the hybrids go through the same multi-wire twisted pair cable that connects to the DB where all the data and control signals go. Some of the pairs are dedicated for the powering, but not many, so the effective cross-section is not much. Hence the impedance of the power and ground lines is considerable and the voltage drops when the system is running is in the range of hundreds of mV. In addition, the ground lines are connected together in the SB backplane and detector feed-throughs so we are introducing a ground loop, whose effect has been minimized by keeping all the cables as together as possible to minimize the area of the loop.

The regulators are supplied by a MARATON system, which is a low noise and low ripple modular power supply system developed for being located in hostile (radiation and magnetic field) environments. The MARATON system is thoroughly described in chapter 4.

2.4.6   HIGH VOLTAGE – 'CAEN'

A commercial 'CAEN' power supply system consisting of 'SY-1527-LC' mainframes with 'A1511B' modules is used. Each module has twelve floating channels that deliver up to 500 V at a maximum current of 10 mA per channel. One mainframe with 13 modules is used for the TT and one mainframe with 8 for the IT. They are located in the counting room racks.

Four IT detector units and up to three TT detector units are connected to one power supply channel. However, individual supply lines for each detector unit run from the counting house to the detectors. The splitting of one power-supply output into three (TT) or four (IT) supply lines is implemented in a patch panel that is located in the counting house, in the same racks as the power supplies. Each output line is equipped with a HV jumper to permit disconnecting individual detector units. There is another set of patch-panels close to the detector that provides a straightforward 1:1 connection to shorter cables of the same type that lead further on to the detectors.

On the front-end hybrids, the bias voltage is filtered using a 2nd order low pass consisting of two 1K0 Ω resistors and two 100 nF capacitors.

A detailed description of the power supplies system is given in chapter 5.

2.4.7 GROUNDING AND SHIELDING SCHEME

The grounding scheme of the Silicon Tracker is illustrated in Fig. 2.16. A common ground reference for all detector units has been defined inside each detector box. The inner coating of the detector box is connected to the detector ground and the outer coating is connected to safety ground. Both coatings are connected together at one point in the detector box. The detector ground is connected to the Service Boxes using ten (TT) resp. 18 (IT) out of the 68 AWG30 wires of the data cables that connect the hybrids to their DBs. The shielding of these cables is connected to the outer shield of the detector box, using the mounting screws of the connectors. At the SB, they are connected to the aluminum frame of the crate, using a dedicated wire. The cable shields of the bias-voltage cables are likewise connected to the outer shielding of the detector box. At the other end, they are connected to the metallic structure of the patch panel, which in turn is connected to the safety ground.

To provide a solid and permanent connection of the detector boxes to the safety ground, dedicated grounding cables were implemented. In the case of the TT, two copper braids of 50 mm^2 cross section each have been included in each of the four flexible cable chains. This adds up to a total grounding cross section of 200 mm^2 per half-box. They are screwed to the aluminum frame of the detector box at one end and to the frame holding the SBs at the other end. The SB frames are connected directly to safety ground. In the case of the IT, one copper cable of a cross section of 35 mm^2 cross section has been installed in the flexible cable chains for each half station. It is connected to the SB mounts at one end and to the metallic structure at the fixed end of the cable chain at the other end.
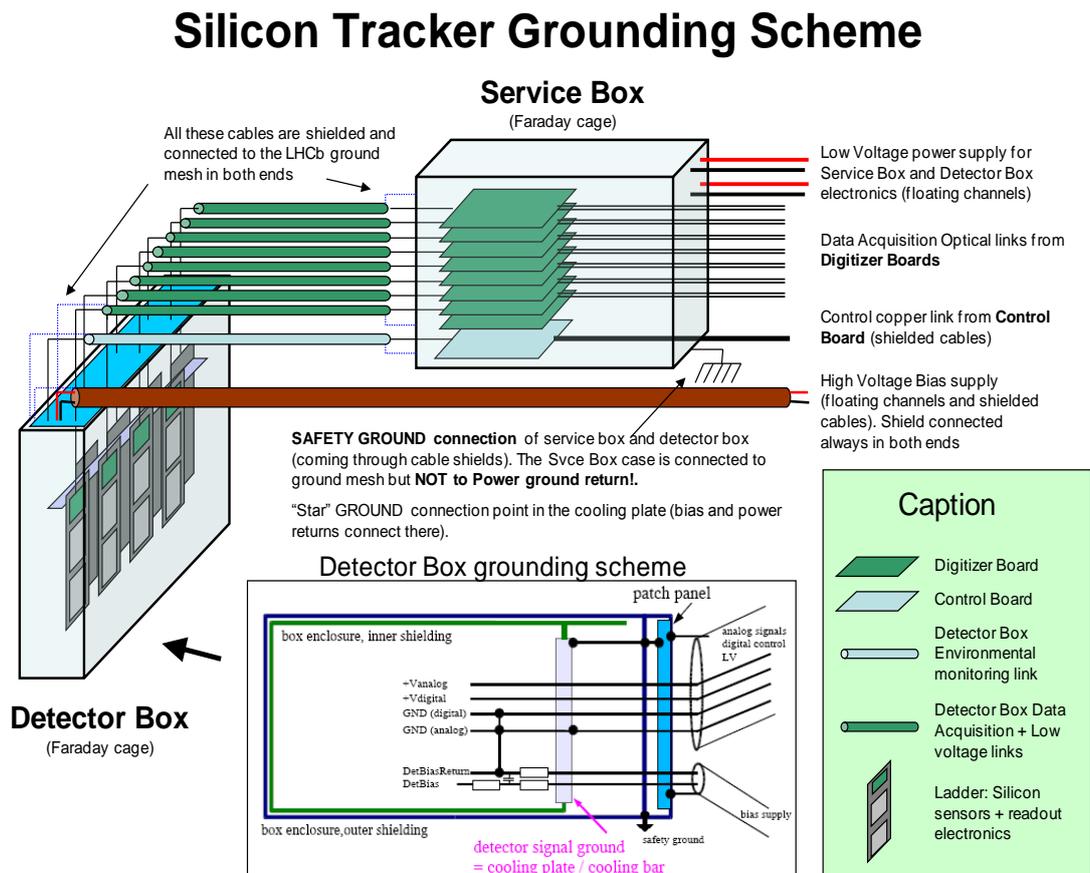
Figure 2.16: Silicon Tracker grounding scheme.

CHAPTER 3

DESIGN AND DEVELOPMENT OF THE CONTROL BOARD

ABSTRACT

This chapter is intended to give a detailed description of the Silicon Tracker Control Board (CB) design and use. This electronic card is composed of several elements and it is the hardware part of the Silicon Tracker (ST) that fulfils both the ST ECS (Experiment Control System) and TFC (Timing and Fast Control) tasks next to the detector. A description of the CB functionalities will be shown as well as a detailed description of its design and development. The contents of this chapter are fully reported in [25].

"La mejor forma de librarse de un problema es resolverlo"

Brendan Francis

3.1 INTRODUCTION

The Silicon Tracker Control Board has been designed and developed to control the ST hardware in the cavern. The hardware located in the cavern is housed in the service boxes and detector boxes. The detector boxes contain the silicon sensors and the readout electronics, whereas the service boxes hold the digitizing and control electronics. As the structure of the Inner Tracker detector and the Tracker Turicensis is very similar but with a different partitioning scheme, a common controller card has been designed for both detectors. Figure 3.1 shows the basic layout of the ST electronics in the cavern.

This card provides the necessary functionalities to carry out several control and monitoring tasks, namely: the slow and fast control of the Silicon Tracker readout and digitizing electronics; the control and monitoring of the LV power regulators; and the monitoring of the detector box temperatures and humidity.

A requirement that the Control Board has to meet is that the devices and components on it need to tolerate the radiation levels present in the cavern where the service boxes are located. For this reason some specific radiation hard devices developed at CERN are used. Several devices tested by our group have been chosen, and also other ones tested by other LHCb and CERN groups are used as well.

In the following sections a detailed description of the CB functionalities and construction is shown. First we define the several functionalities that the CB has to fulfil, then comes a detailed description of all its functional blocks together with its implementation. The layout and some other technical specifications for the construction and installation of the CBs are also shown in the final appendixes.
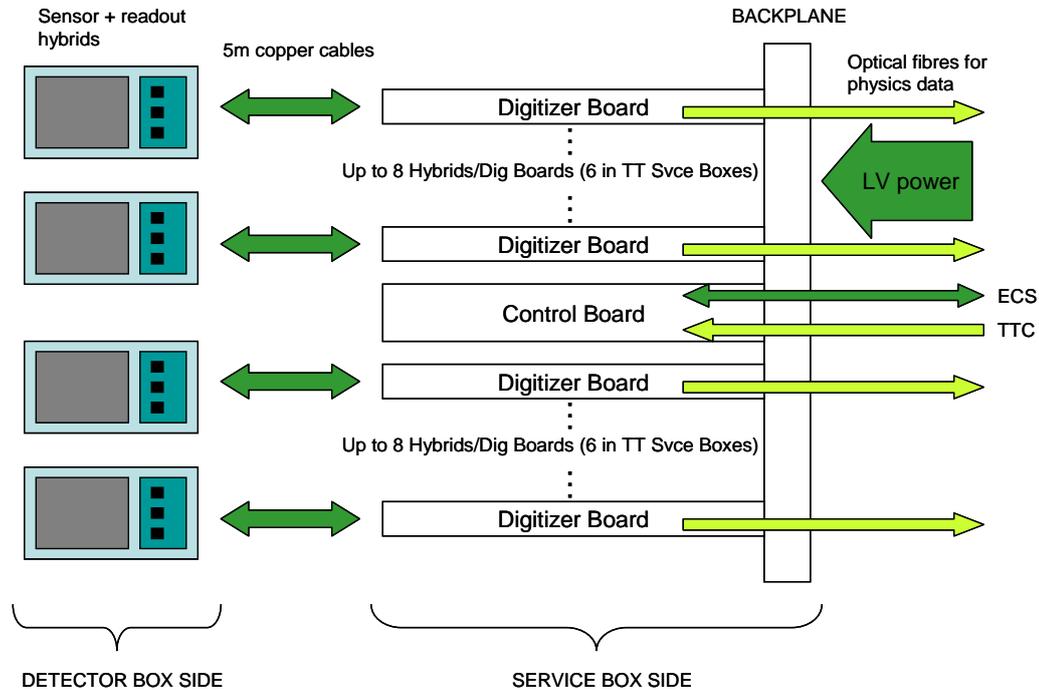
Figure 3.1: Schematic overview of the components and their connections.

3.2 CONTROL BOARD FUNCTIONAL DESCRIPTION

Figure 3.2 depicts the functional block diagram of the CB, and the Figure 3.3 shows a prototype CB and the blocks physical distribution on it:

- *Power Regulators:* Three LHC4913 [40] power regulators for supplying the 2.5 V, 3.3 V and 5 V needed by the CB devices.
- Frontal and backplane connectors:
  - *a.* The frontal connectors consist of:
    - *i.* 1 × 'SubD' high density connector for the detector box environmental monitoring.
    - *ii.* 2 × 'RJ45' connectors for the SPECS [31] daisy-chain connection.
    - *iii.* 1 × 'AMP/Tyco' 6 pin row connector for monitoring the IT half station position switches.

> *b.*   The backplane connectors are 2 half size 'DIN 41612'.

- *TTCrq:* The TTCrq [41] is mezzanine card developed by the CERN microelectronics group containing the TTCrx and other associated components such as a pin-preamplifier and a QPLL.

  - *a.*   TTCrx: An ASIC receiver developed for the LHC Timing, Trigger and Control (TTC) distribution system [42]. The ASIC implements an interface between the front–end electronics and the TTC system making the TTC coding and multiplexing schemes transparent. The receiver delivers the LHC timing reference signal, the first level trigger decisions and its associated bunch and event numbers. It can be programmed to compensate for the propagation delays associated with the detectors and their electronics. The IC supports the transmission of data and of synchronised broadcast commands.

  - *b.*   A radiation tolerant connectorized PIN photodiode with a trans-impedance amplifier for the reception of the TFC signals from the optical TTC network.

  - *c.*   QPLL: Phase-Locked Loop based on a Voltage Controlled Quartz Crystal Oscillator (VCXO). Its function is to act as a jitter-filter and clock multiplier for clock signals operating at the LHC bunch-crossing frequency (40.08 MHz). The device is implemented in a 0.25 μm CMOS technology using radiation tolerant layout practices.

- *Delay25 chip:* It is a 5 channel CMOS programmable delay line featuring 4 channels that allow phase delaying of periodic or non-periodic digital signals and a master channel that can be used to phase delay a clock signal. The master channel serves as a calibration reference. The phase of each channel can be independently programmed with a resolution of 0.5 ns through an I^2C interface. The reference clock frequency can be any of the following: 32, 40, 64 or 80 MHz.

- *SPECS slaves:* These are two mezzanine cards hosted in the CB developed for interfacing electronic devices to the ECS. The SPECS system is composed of a PCI master card (located in the control room far away from the harsh radiation environment) which can address several SPECS slave mezzanines. The heart of the mezzanines is a radiation tolerant FPGA, but it also contains a radiation hard ADC (the DCU [43]) and some radiation tolerant LVDS drivers and receivers for driving the SPECS communication to the master. Three standard interfaces are provided by the SPECS Slave to control the components on the board or at distances of up to 10 meters: I^2C, JTAG and a (simple) parallel bus, from which we only use the I^2C.

- *SPECS chaining devices:* it is a compendium of LVDS drivers and receivers implemented in the CB for daisy-chaining many SPECS slaves in the same SPECS master link. This chaining mechanism is both used for interconnecting the 2 SPECS slaves in the CB and also for interconnecting with other CBs.

- *Signal conditioning devices:* wheatstone bridges and instrumentation amplifiers used for temperature and humidity signal conditioning.

- *DCU:* a radiation hard ADC used for sampling the humidity signal and some others.

- *I^2C bus switches:* a mezzanine card whose purpose is demultiplexing the "SPECS fashion" long I^2C bus into 8 individual open collector compatible I^2C buses. These I^2C buses are necessary for controlling all the Beetles in the Detector Boxes and the GOLs and DCUs in the DBs. A detailed description of the partitioning and I^2C address assignment is shown later.

- *Level shifter:* it consists of 2 bipolar transistors working in saturation mode which allow for translating the 2.5 V I^2C generated by the SPECS slave into 3.3V necessary for TTCrq voltage compatibility.

It is also of special interest the clock and TTC signal distribution, as we'll see in section 3.2.8. The TTCrq delivers the two clocks needed for the Beetles and the sampling electronics (C*lockDes1* and C*lockDes2*'), the trigger signal (*L0accept*) and the TTC command bus from which the *L0reset* and the test-pulse calibration signal (*calib0*) are decoded. The *L0reset* and the test-pulse signal are decoded in the SPECS FPGA and the decoding of the latter can be delayed by the SPECS from 0 to 16 cycles of 25 ns. All these signals but the C*lockDes2* pass through the Delay25 chip so they can be phase delayed within a range of 32 ns in steps of 0.5 ns permitting a fine tuning of the signals timing.

3.2.1   CONTROL BOARD POWER REGULATORS

The three necessary voltages (2.5 V, 3.3 V and 5 V) for the CB devices are delivered by 3 rad-hard positive voltage regulators L4913. Two different voltages of ~4.5 V and ~7 V are supplied by the low voltage MARATON power supplies. The ~4.5 V line from the MARATON will supply the 2.5 V power regulator and the ~7 V will supply the 3.3 V and 5 V regulators. The figure below (Figure 3.4) shows the schematic for

one of the regulators. The *INH* line is left open since the CB must be always supplied, and the *OCM* line is connected to one of the SPECS I/O lines.

The three voltages are monitored by three DCU channels through a voltage divider since the maximum allowed input voltage for the DCU is 2.5 V. The Figure 3.5 shows the voltage divider.
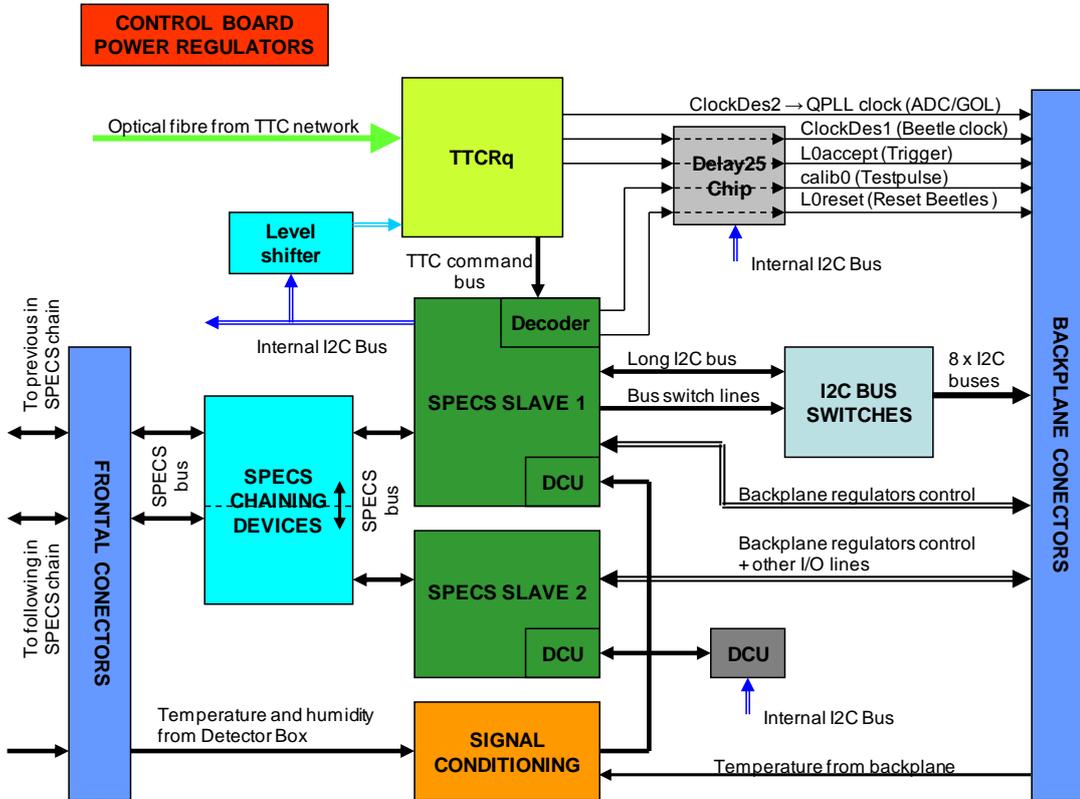
Figure 3.2: Control Board block diagram.
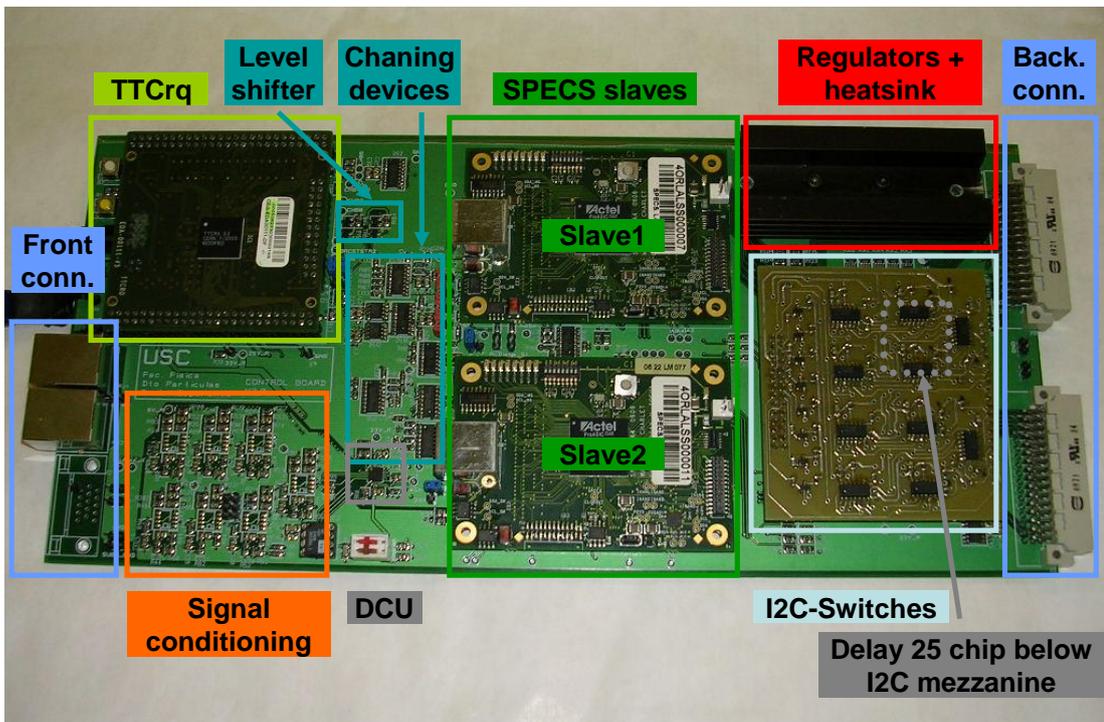
Figure 3.3 : Control Board image showing the blocks distribution.
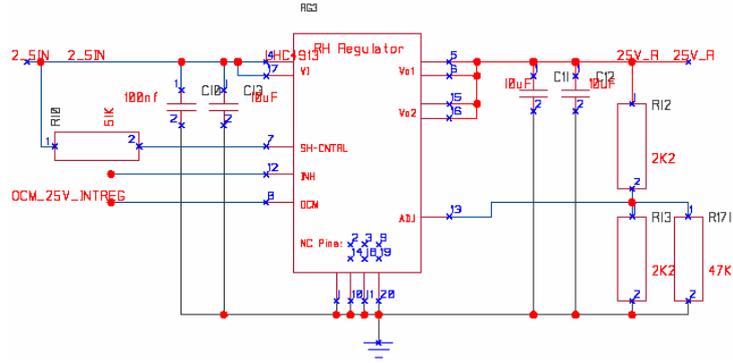
Figure 3.4: Schematic design for the 2.5 V voltage regulator.
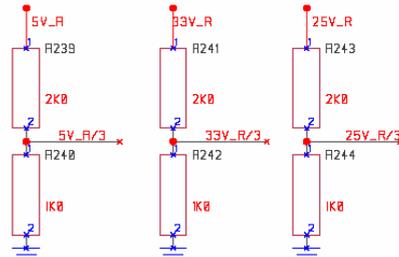
Figure 3.5: Voltage dividers for power supply monitoring.

3.2.2 HEAT DISSIPATION STUDY

An important issue when using power regulators is the thermal dissipation management. The calculations that will be shown next reveal that heat-sinks or heat dissipaters attached to the regulators are mandatory. Due to space reasons is not possible to attach a heat-sink to each regulator, but two for the group of three or building a custom cooling block.

The Table 3.1 shows the thermal resistance for various packages taken from a 'ST Microelectronics' note. [44]. Our regulators are packaged with the 'PowerSO20' slug-up type and the yellow shaded cells show the values of interest for us.

Symbol	Description	PowerDIP20	SO20	PowerSO20	Unit
$R_{th\text{-}j\text{-}pins}$	Maximum Thermal Resistance Junction-Pins	12	14	-	ºC/W
$R_{th\text{-}j\text{-}case}$	Maximum Thermal Resistance Junction-Case	-	-	2	ºC/W
$R_{th\text{-}j\text{-}amb1}$	Maximum Thermal Resistance Junction-Ambient [1]	40	51	-	ºC/W
$R_{th\text{-}j\text{-}amb1}$	Maximum Thermal Resistance Junction-Ambient [2]	-	-	35	ºC/W
$R_{th\text{-}j\text{-}amb1}$	Maximum Thermal Resistance Junction-Ambient [3]	-	-	15	ºC/W
$R_{th\text{-}j\text{-}amb2}$	Maximum Thermal Resistance Junction-Ambient [4]	56	77	62	ºC/W

Table 3.1: Thermal resistances for the PowerSO20 slug-down package.

(1)   Mounted on a multi-layer FR4 PCB with a dissipating copper surface on the bottom side of 6 cm^2 (with a thickness of 35 μm).

(2)   Mounted on a multi-layer FR4 PCB with a dissipating copper surface on the top side of 6 cm^2 (with a thickness of 35 μm).

(3)   Mounted on a multi-layer FR4 PCB with a dissipating copper surface on the top side of 6 cm^2 (with a thickness of 35 μm), 16 via holes and a ground layer.

(4)   Mounted on a multi-layer FR4 PCB without any heat sinking surface on the board.

The maximum junction temperature that the regulator should withstand is ~125 ºC, and the maximum power consumption will take place in the 3.3 V volts regulator which will be delivering 1.5 A for a TT service box. Assuming a worst case scenario with ~7 V at the input:

$\Rightarrow$ $P_j = (7 - 3.3) \times 1.5 = 5.55$ W

If we don't use any kind of heatsink and we use as approximation the $R_{th\text{-}j\text{-}amb2}$ value for the 'SO20' (since our regulator is slug-up) we obtain a temperature difference of:

$\Rightarrow$ $T_j - T_a = P_j \times R_{th\text{-}j\text{-}amb2}(SO20) = 427.35 \,ºC$

which means that the junction temperature will be 427 ºC higher than the ambient, so the regulator will clearly die at room temperature. Even applying reasonable layout techniques an average area of 4.7 cm^2 of dissipating copper connected through many vias to the ground layer has been created for each regulator. This leads us to a $R_{th\text{-}j\text{-}amb2}$ of ~52, according to the pictures in 3.6:
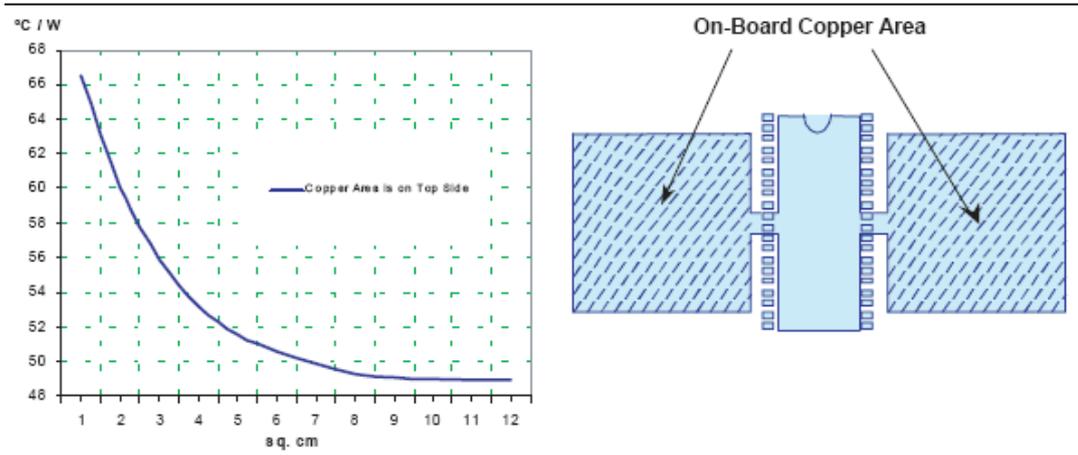
Figure 3.6: So20 mounting and Junction-Ambient thermal resistance versus on-board copper area.

This value does not still fulfill our thermal requirements:

$\Rightarrow$ $T_j - T_a = P_j \times R_{th\text{-}j\text{-}amb2}$ ('SO20' 4 cm^2) = 288.6 ºC

We've been making pessimistic assumptions using the SO20 thermal resistance data and supposing 4.7 cm^2, but the reliability and longevity of any semiconductor device is (roughly) inversely proportional to the square of the junction temperature change. Thus halving the junction temperature will result in approximately 4 times more the expected life of the component. A worthwhile increase in reliability and component lifetime can be achieved by a relatively small reduction in the operating temperature due to this square relationship.

So it is clear that the installation of a heatsink of heat dissipater is mandatory.

The kind of package that the regulator uses is slug-up 'PowerSO20', so they are optimized to dissipate the power through the upper side. The junction-case thermal resistance of this package is around 2 ºC/W.

The junction to ambient thermal resistance using good layout techniques is around 53 for the SO20 package, as shown previously. This value is quite pessimistic, as the experimental data will show up.

$\Rightarrow$ $R_{th\text{-}j\text{-}case} = 2 \,ºC/W$ $\rightarrow$ Thermal resistance between junction and case.
$\Rightarrow$ $R_{th\text{-}c\text{-}dis} = 1 \,ºC/W$ $\rightarrow$ Thermal contact resistance between case and heatsink.
$\Rightarrow$ $R_{th\text{-}d\text{-}amb} = 14 \,ºC/W$ $\rightarrow$ Thermal resistance between the chosen heatsink and ambient.
$\Rightarrow$ $R_{th\text{-}j\text{-}amb} = 53 \,ºC/W$ $\rightarrow$ Thermal resistance between pin and ambient through the PCB copper.

The copper area around the regulators is ~14 cm^2, so the effective area for one of them is ~4,7cm^2. We will share 2 heatsinks between the 3 regulators, so the effective thermal resistance for 1 regulator can be considered to be $3 \times 14/2 = 21$ ºC/W.

So the total thermal resistance for one regulator is:

$\Rightarrow$ $R_{j\text{-}amb} = (R_{th\text{-}j\text{-}case} + R_{th\text{-}c\text{-}dis} + R_{th\text{-}d\text{-}amb} \times 3/2) \,||\, R_{th\text{-}j\text{-}amb} = 16.5 \,ºC/W$

The regulator that dissipates most of the power is the 3.3V one, with a current of ~1.5 A and a voltage drop of 3.7 V in case we apply 7 V to its input. So the power dissipated and the temperature difference between silicon and ambient is:

$\Rightarrow$ $P_{reg33} = 5.55 \,W$
$\Rightarrow$ $T_j - T_{amb} = P_{reg33} \times R_{j\text{-}amb} = 91.7 \,ºC$

Adding a parallel heatsink of 17 ºC/W to the previous one, we improve the resistance to:

$\Rightarrow$ $R_{j\text{-}amb} = 11.38 \,ºC/W$ and a

$\Rightarrow$ $T_j - T_{amb} = P_{reg33} \times R_{th-j-amb} = 63.18$ ºC,

which is a valuable improvement.

In order to certify that the previous are realistic and even pessimistic the value of the thermal resistance has been obtained empirically: precise measurement of the current delivered by each regulator and the voltage drop on it are necessary; some temperature measurements with the electronics switched on need to be performed.

The current values for the different regulator in the CB are under the test conditions:

$\Rightarrow$ $I_{3\_3VReg} = 0.64$ A

$\Rightarrow$ $I_{5VReg} = 0.38$ A

$\Rightarrow$ $I_{2\_5VReg} = 0.02$ A

The test has been performed with and without the 14 ºC/W heatsinks plugged for three different input voltages. The temperature measurements have been performed by directly touching the regulator case with conductive glue for the T_{reg} measurement and at a distance 10 cm from the regulators for the T_{amb}.

As we can see, the theoretical values have been a little bit pessimistic. We supposed the PCB copper thermal resistance to be 52 ºC/W ($R_{th-j-amb}$), when from the previous measurements we get a value around ~23. From this follows that with heatsinks plugged we obtain an experimental value ~11.7, whereas the theoretical one should be 16.5.

Regulators with heatsinks plugged			
Vin	Tamb	Treg	Rj-amb = [Treg-Tamb]/[I3_3VReg×(Vin-Vout)]
6.5	27	51.2	Rj-amb = 24.2/2.048 ≈ 11.8
7	25.3	53.0	Rj-amb = 27.7/2.368 ≈ 11.7
7.5	25.5	56.5	Rj-amb = 31/2.688 ≈ 11.5
Regulators without heatsinks			
Vin	Tamb	Treg	Rj-amb = [Treg-Tamb]/[I3_3VReg×(Vin-Vout)]
6.5	27	76	Rj-amb = 49/2.048 ≈ 23.9
7	24.6	78.1	Rj-amb = 53.5/2.368 ≈ 22.6
7.5	24.6	88	Rj-amb = 63.4/2.688 ≈ 23.6

Table 3.2: Thermal resistance empirical results.

From the experimental results we can conclude that adding the 2 heatsinks in parallel we can obtain a thermal resistance around:

$\Rightarrow$ $R_{th-j-amb} = (R_{th-j-case} + R_{th-c-dis} + [R_{th-d-amb1} \| R_{th-d-amb2}] \times 3/2) \| R_{th-j-amb} = 14.5 \| 23.5 = $ **8.96 ºC/W**

And a,

$\Rightarrow$ $T_j - T_{amb} = P_{reg33} \times R_{th-j-amb} = $ **49.76 ºC**

Which is a reasonable value. This value could also be improved by setting a lower voltage at the regulators input. According to the regulator datasheet, the regulator drop-out voltage is 0.5 V @ $I_0 = 1$ A and 1.5 V @ $I_0 = 3$ A so an input voltage of 6.5 V is high enough and implies a valuable improvement:

$\Rightarrow$ $P_j = (6.5 - 3.3) \times 1.5 = $ **4.8 W**

$\Rightarrow$ $T_j - T_{amb} = P_{reg33} \times R_{th-j-amb} = $ **43 ºC**

3.2.3   HEAT DISSIPATION IN THE SERVICE BOX

Despite the careful heat dissipation study performed showing quite promising results that could lead us to think about a solution based upon air heatsinks, after installing the first CBs in the final svce boxes, the results showed up that this option is not reasonable anymore.

When having all the electronics powered working under nominal conditions, the temperature measured in the power regulators heatsink show up a value of up to **~70ºC** at ambient temperature of **~25ºC**. This implies a roughly estimated temperature of about **~90ºC** in the regulator silicon junction. This value has been considered to be rather high for operating the regulator in the long term so the decision of building water

cooled cooling block. The first design of the cooling circuit for the CB is shown in Figure 3.7. It consists of a brass cooling block and two polyurethane hoses. This design was later replaced by one completely metallic (Figure 3.8).
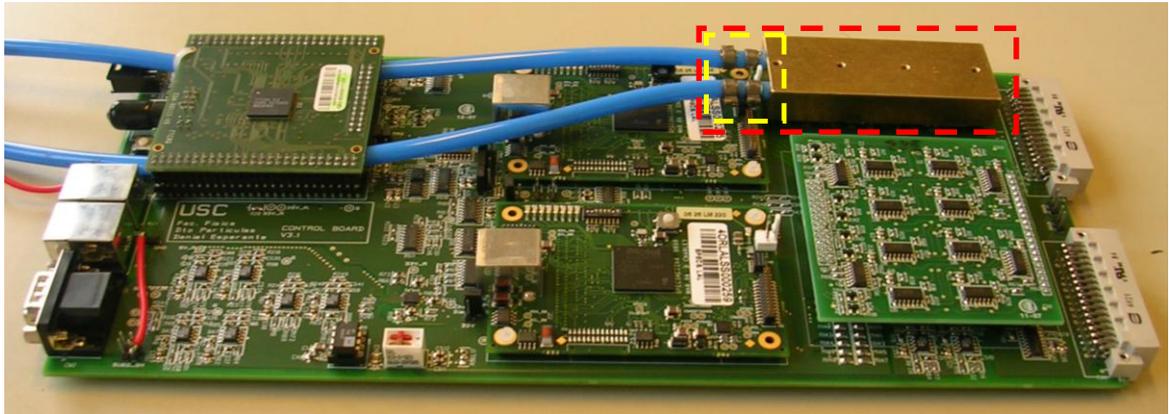
Figure 3.7: Final Control Board with water cooled cooling block with polyurethane hoses. This was the first design installed in the experimental area.

Figure 3.8: Final Control Board with the final water cooled cooling block.

This discrepancy in the measured value when running the CB once installed in the final SB with respect to what has been shown previously comes from three main points:

1. The previous calculations are performed assuming an input voltage value of 7 V, while in the experiment up to 7.3 V are foreseen.
2. In the previous tests, the heat dissipation coming from other neighboring boards was not taken into consideration.
3. The values for the heat dissipation resistances of the heatsinks are given taking into account a "good" convection, that is, they are foreseen for being used in horizontal position. In the service box all the boards are installed vertically so the effectiveness of the heatsink fins gets considerably worse and hence the effectiveness of the heatsinks.

Apart from the previous reasons, we could add two more for the TT SBs:

1. The 3.3 V regulators has to deliver ~0.6 A of extra current since the power for the backplane clock LVDS drivers is taken from the CB itself.
2. The TT SBs are installed in a stack of six, so the CB in the top is always running at higher environment temperature than the other ones.

The first design of the cooling block circuit (the one with the blue water hoses) was tested in the lab and found to be effective for dissipating the power under nominal working conditions. The only weak point of this solution was the potential risk of leaks on the hose to cooling block connection. When running in the final experiment a water pressure in the outlet circuit of ~3-4 bar is foreseen so the pressure in the inlet must at least range from 4 to 5 bar so that we guarantee a minimal water flow. Pressure tests up to 10 bar have

been carried out in the lab and after some research and several trials the following solution was adopted: clamp the hoses to the cooling block with 2 rings (see yellow box in the picture) and use some 'Araldite' glue. Like this a leak tight connection can be achieved. For more safety each cooling block has been individually temperature cycled and tested at 10 bar before it was mounted on the Control Board.

After the first operation of the LHC and during the long shutdown caused by the LHC machine failure, the cooling circuit was replaced by one constructed in cooper (the popes) and brass (the block). This design is more reliable against water leaks in the circuit interconnections and also longer term solution since the polyurethane hoses can degrade as they get old.

3.2.4 FRONTAL AND BACKPLANE CONNECTORS

The CB features two connectors with the backplane in the rear side and four in the front side for connecting the "outside world". The backplane connectors are two half size 'DIN 41612', whereas the frontal ones are 2 × 'RJ45' for SPECS communication, 1 × 'SubD' high density 15 pin for environmental signals and one additional one for monitoring the box position switches.

3.2.5 BACKPLANE CONNECTORS

The Figure 3.9 shows the backplane connectors and the signals pinout assignment.

The regulators control and monitoring signals are located in the top connector and the I^2C and backplane temperature monitoring in the bottom one. The power lines are in the top connector and there are ground connections in both of them. The TFC signals, i.e. the *L0accept* (*TRIGGER*), the *L0reset* (*FE_RST*), the test-pulse or *calib0* (*TEST_PULSE*), the *clockDes2* (*CLOCKADCGOL*) and the *clockDes1* (*CLOCKBEETLE*) are also distributed in both.

3.2.6 FRONTAL CONNECTORS

There are four connectors in the frontal side:

- The ECS connector (Figure 3.10), for the detector box environmental measurements. It provides 8 lines for the 4 × PT1000 sensors (2 lines per sensor), 4 lines for the HMX2000 humidity sensor [45] (2 for sensor biasing and 2 for the sensor signal output) and 3 lines for a non rad-hard HIH-3600 humidity sensor which could be used for calibration and testing purposes. A more detailed explanation on how the biasing and readout of the sensors is done in section 3.2.19. The connector type is 15-pin high density male SUB-D connector. Signals:

  1. PT1000_2- : 2nd PT1000 sensor inverting pin.
  2. *RH_HIH_SIG* : HIH-3600 humidity sensor voltage output.
  3. *RH_VEXC-* : HMX2000 sensor negative bias pin.
  4. *RH_VEXC+* : HMX2000 sensor negative bias pin.
  5. *RH_VSIG+* : HMX2000 sensor non inverting signal output pin.
  6. *PT1000_4-* : 4th PT1000 sensor inverting pin.
  7. *PT1000_2+* : 2nd PT1000 sensor non inverting pin.
  8. *5V_R*: 5 V from regulator. This connection is left open unless the jumper is plugged (Figure 3.10). It was foreseen for the HIH-3600 supply.
  9. *GND*: CB ground. This connection is left open unless the jumper is plugged (Figure 3.10). It was foreseen for the HIH-3600 ground.
  10. *RH_VSIG-* : HMX2000 sensor inverting signal output pin.
  11. *PT1000_4+* : 4th PT1000 sensor non inverting pin.
  12. *PT1000_3-* : 3rd PT1000 sensor inverting pin.
  13. *PT1000_3+* : 3rd PT1000 sensor non inverting pin.
  14. *PT1000_1+* : 1st PT1000 sensor non inverting pin.
  15. *PT1000_1-* : 1st PT1000 sensor inverting pin.
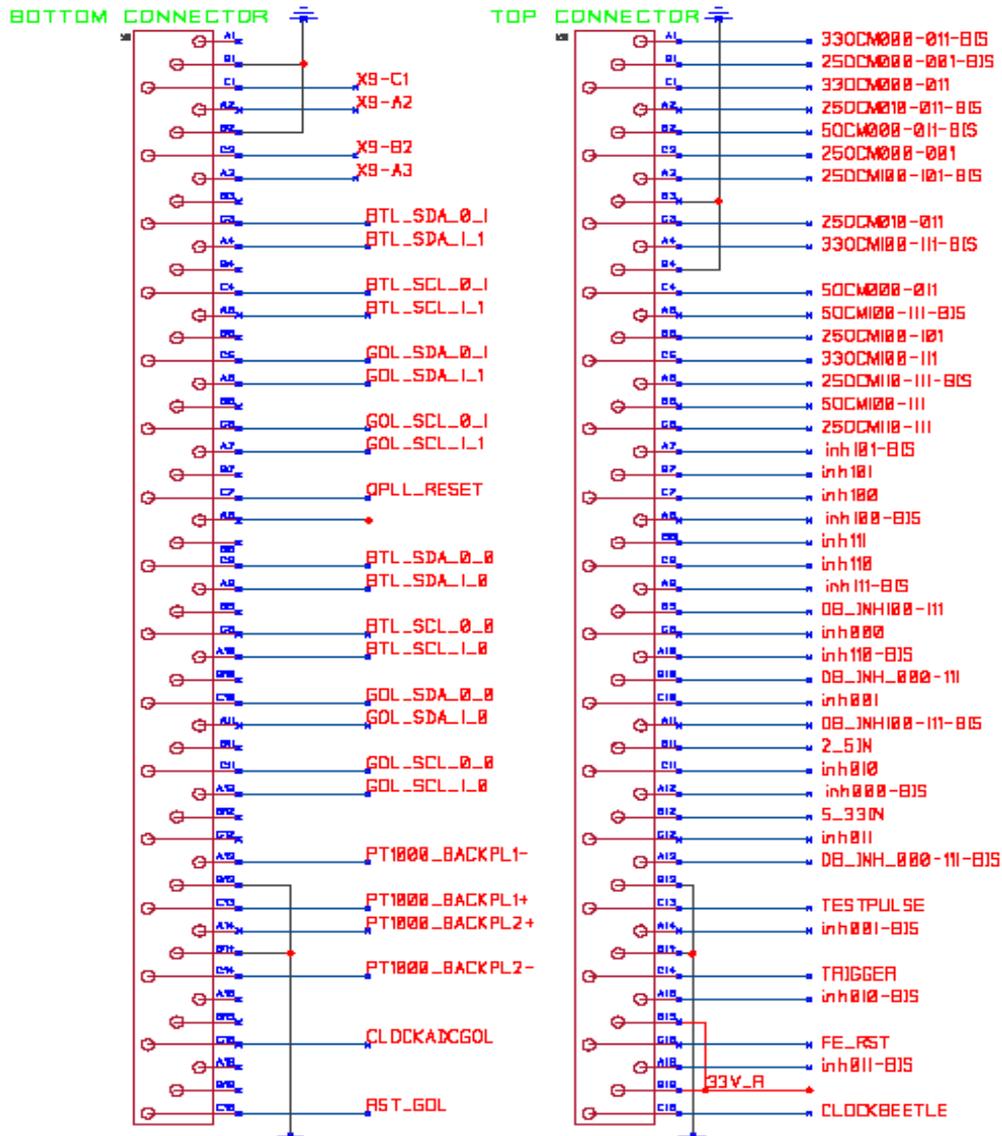
Figure 3.9: Backplane connectors.

- The two RJ45 connectors for the SPECS bus (Figure 2.9). The upper connector carries the signals coming from the previous SPECS slave or from the master, and the lower one is for connecting to the next CB in the chain. The used cables must be shielded CAT6 Ethernet cables with 100Ω differential impedance. The connector type is an 8-pin RJ45. The signals for connector *RJ1* (the upper one) are the following. The signals for connector *RJ2* are equal to the *RJ1* but instead of connecting to the slave1 (the upper one), they connect to the slave2:

1. *SDA_MS1_IN-* : SDA master to slave -.
2. *SDA_MS1_IN+* : SDA master to slave +.
3. *SCL_MS1_IN-* : SCL master to slave -.
4. *SCL_SM1_OUT+* : SCL slave to master +.
5. *SCL_SM1_OUT-* : SCL slave to master -.
6. *SCL_MS1_IN+* : SCL master to slave +.
7. *SDA_SM1_OUT-* : SDA slave to master -.
8. *SDA_SM1_OUT+* : SDA slave to master +.

- The station position monitoring connector. This extra connector (Figure 3.12) has been added for monitoring the voltage value of the position monitoring switches. This monitoring capability can be used provided that a shielded cable is used and the input signals are protected against over and

under-voltages that could damage the electronics. The connector type is a double row header, 2×4 way right angle, 2.54 mm, from TYCO electronics. Signals:

1. *AMP_SH* : This is the pin where the cable shield must be connected.

2. *AMP_CON1*: Beam pipe distance monitoring digital output 1.

3. *AMP_CON2*: Beam pipe distance monitoring digital output 2.

4. *AMP_CON3*: Beam pipe distance monitoring digital output 3.

5. *AMP_CON4*: Beam pipe distance monitoring digital output 4.

6. *33V_R* : 3.3 V from CB regulator.

7. *GND* : CB ground.

8. *GND* : CB ground.
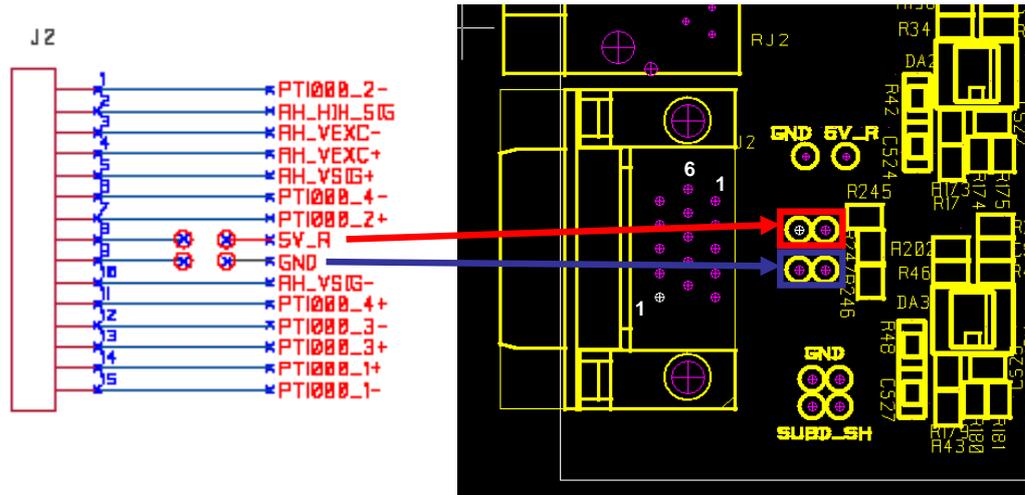
Figure 3.10: ECS connector and jumpers pinout.
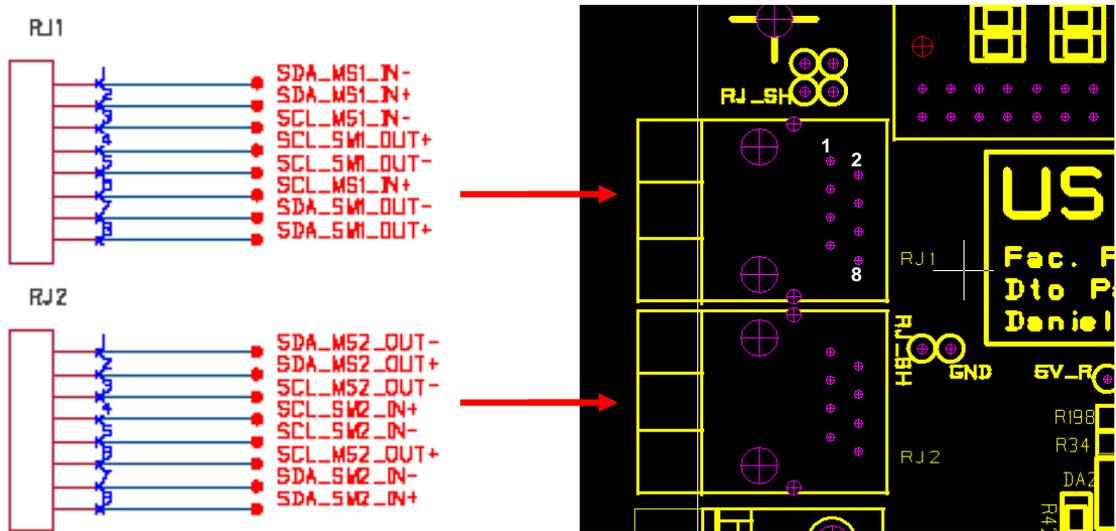
Figure 3.11: RJ45 SPECS connectors.

The shields from the cables connecting to the frontal part of the Control Board must be connected to the LHCb ground mesh in order to comply with the Silicon Tracker grounding policy [38]. For this reason, isolated copper areas have been created in the CB ground layer, which are accessible via some pins implemented next to the connectors (see Figure 3.13).
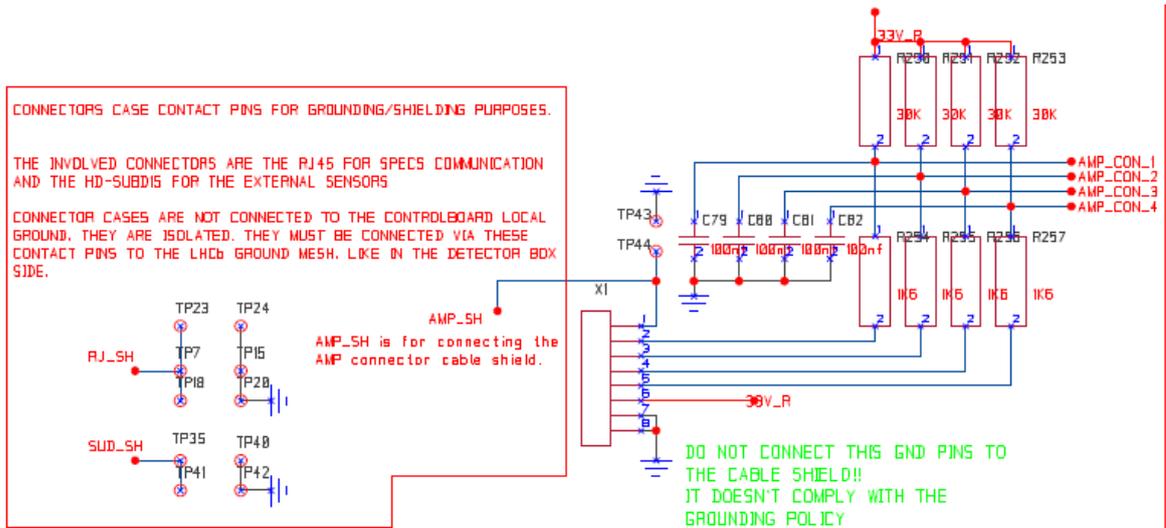
Figure 3.12 : Beam pipe approach monitoring connector and cable shield connections.

3.2.7   THE TTCRQ

The TTCrq is a mezzanine card developed by the CERN microelectronics group containing the TTCrx ASIC and other associated components such as a pin-preamplifier and a QPLL (more info in [41]). This card is placed in the CB frontal side because this mezzanine is the interface to the TTC optical network. The TTCrq features a connectorized PIN photodiode with a trans-impedance amplifier which connects to the TTC network.

The TTCrx ASIC implements the interface between the FE electronics and the TTC system making the TTC coding and multiplexing schemes transparent for us. Among the TTC signals that the TTCrx delivers, the following ones (using the TTCrx signal naming convention) need to be handled in the CB, as depicted in the TTCrx simplified block diagram (Figure 3.14):
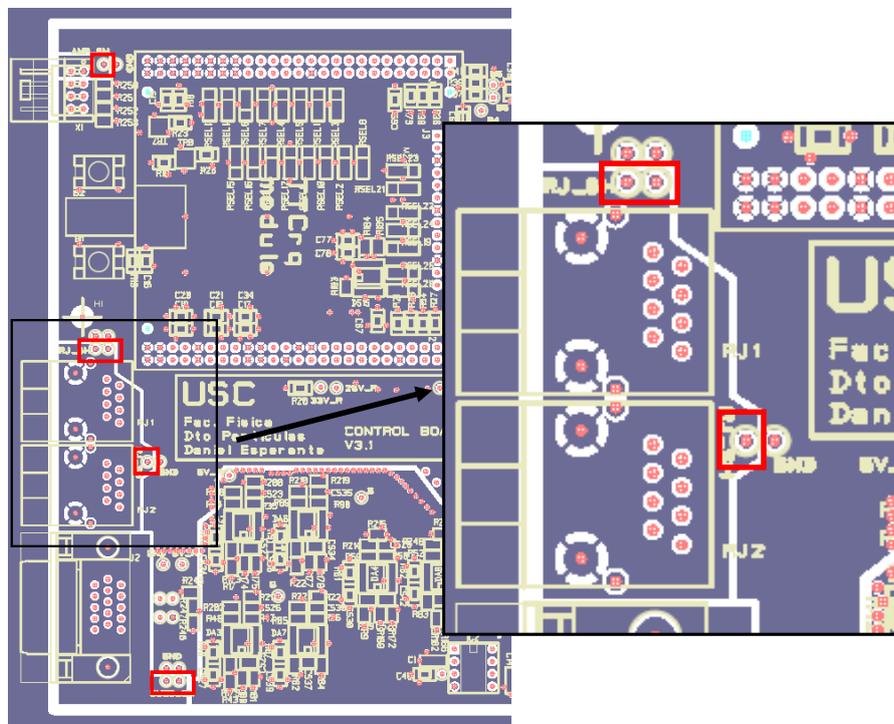
Figure 3.13: Control Board ground layer view showing the connectors metal case connection pins. For each connector a copper island has been implemented in the ground layer. The red boxes enclose the pins that permit connecting the cable shields to the ground mesh for complying with the grounding and shielding policy. The zoomed view shows the SPECS RJ45 metal enclosure connections.

- *Clock40Des1*: This is the 40.08 MHz deskewed reference clock 1. The fine phase delay factor is controlled by writing into the '*fine delay register 1*'. Both the *L1Accept* signal and the channel B broadcast command signals are synchronized to the *Clock40Des1*. This signal also provides the clock for the Beetle chips, so that the reset signal will arrive at the Beetle with a fixed phase relation. The same applies for the calibration signal. However, its phase relation has to be adjustable with respect to the Beetle clock to perform timing scans of the front-end electronics. This is accomplished by using the Delay25 chip together with the SPECS channel B decoder.

- *Clock40Des2*: This is the 40.08 MHz deskewed reference clock 2 controlled by the '*fine delay register 2*'. This output is used to provide the ADCs and the GOL serializers with a clock. As the *Clock40Des2* output can be independently shifted with respect to the *Clock40Des1*, the timing relation between the Beetle and the ADCs can be adjusted to optimize the sampling point of the ADCs on the Beetle data. After this optimization has taken place, *Clock40Des1* and *Clock40Des2* can be shifted together to optimize the timing of the readout with respect to the particle passage through the detector.

- *L1Accept*: First level trigger-accept signal. Although in the TTCrx manual it is named *L1Accept*, in LHCb this signal corresponds to *L0accept* trigger signal.

- *Brcst[7:2]*: Bits from 2 to 7 of the channel B broadcast command. Among other commands, it distributes the *L0reset* and *calib0* which are decoded into signals in the SPECS slave.

- *BrcstStr1*: Broadcast command strobe to validate newly received broadcast data. It is synchronized to the *Clock40Des1*.

- *BrcstStr2*: Broadcast command strobe to validate newly received broadcast data. Not used.
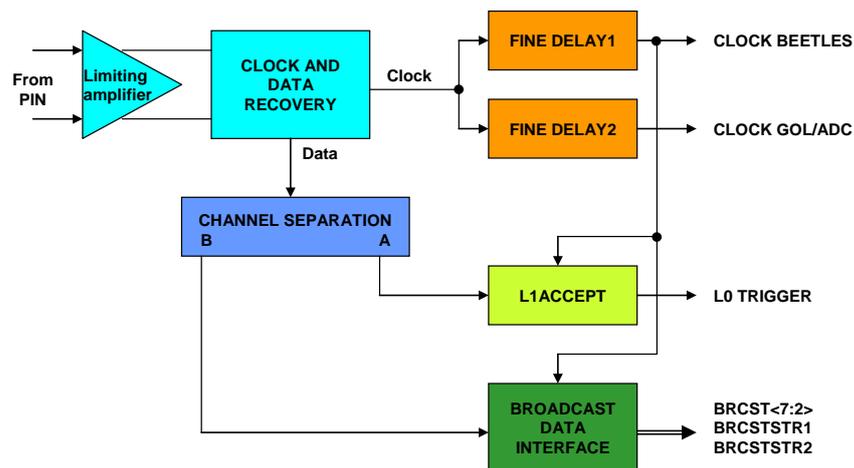
Figure 3.14: TTCrx simplified block diagram

The TTCrq needs 3 different supply voltages: 2.5 V for the QPLL, 5 V for the PIN preamplifier and either 3.3 V or 5 V for the TTCrx. We have decided to supply the TTCrx with 3.3 V since TTCrx timing characteristics are better (jitter is lower) although it is a little less immune to SEUs. Using 3.3 V also eases the design since it is voltage compatible with the SPECS FPGA (the channel B decoder is implemented in the SPECS FPGA). The Figure 3.15 shows the schematic for the TTCrq.

The TTCrx is controlled using the internal I^{2}C line from the SPECS slave1 and a level shifter to 3.3 V as will be explained in section 3.2.11. The I^{2}C address for the TTCrq has been set to "b'101111X" by external resistors in the CB.

In order to use the TTCrq in the way required by the ST (*Clock40Des1* for the Beetles, *Clock40Des2* for the GOL's and source the QPLL with the *Clock40Des2*), some few modifications were performed on it. According to the TTCrq schematic shown in the TTCrq manual, the changes were (see in Figure 3.16 the bottom layout of the TTCrq and the comments):

- Move *R40* to *R41*: This way we source the QPLL with the *Clock40Des2*.

- Change *ST19*: so we hardwire the *Clock40Des1* to the pin 2 of the *J1* connector.

- Remove *R55*: the QPLL LVDS output clock (*Clock40Des2*) must be terminated in the Control Board, so this termination resistor inside the TTCrq must be removed.

- Remove the 2.5 V regulator and solder *R59* (0Ω). As the TTCrq internal regulator is not radiation tolerant, we supply 2.5 V to the TTCrq through the pin 39 of connector *J2*.
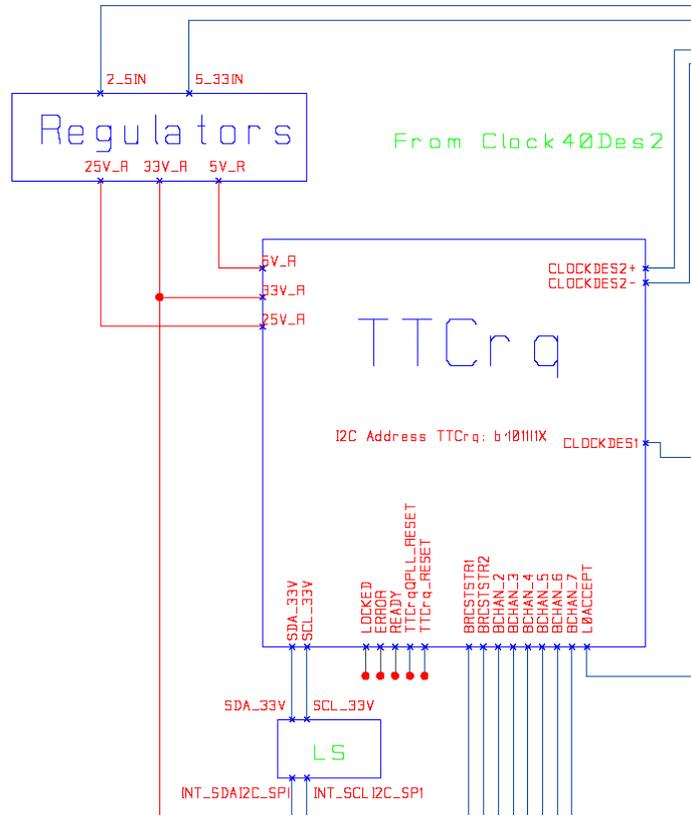
Figure 3.15: TTCrq schematic block diagram
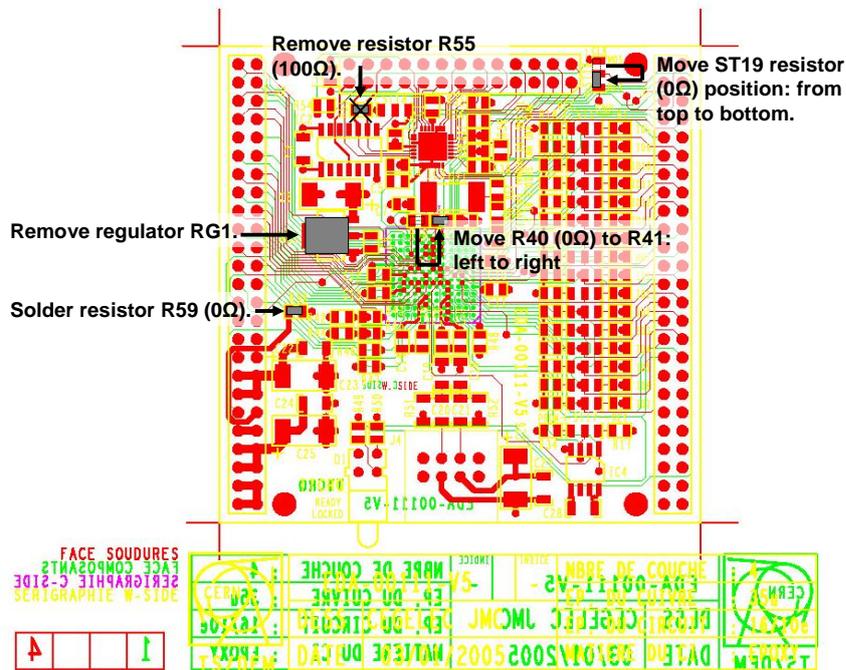
Figure 3.16: TTCrq modifications.

3.2.8 THE TTC AND THE CLOCK SIGNAL DISTRIBUTION

The Figure 3.17 shows the clock and TTC signal distribution diagram implemented in the CB.
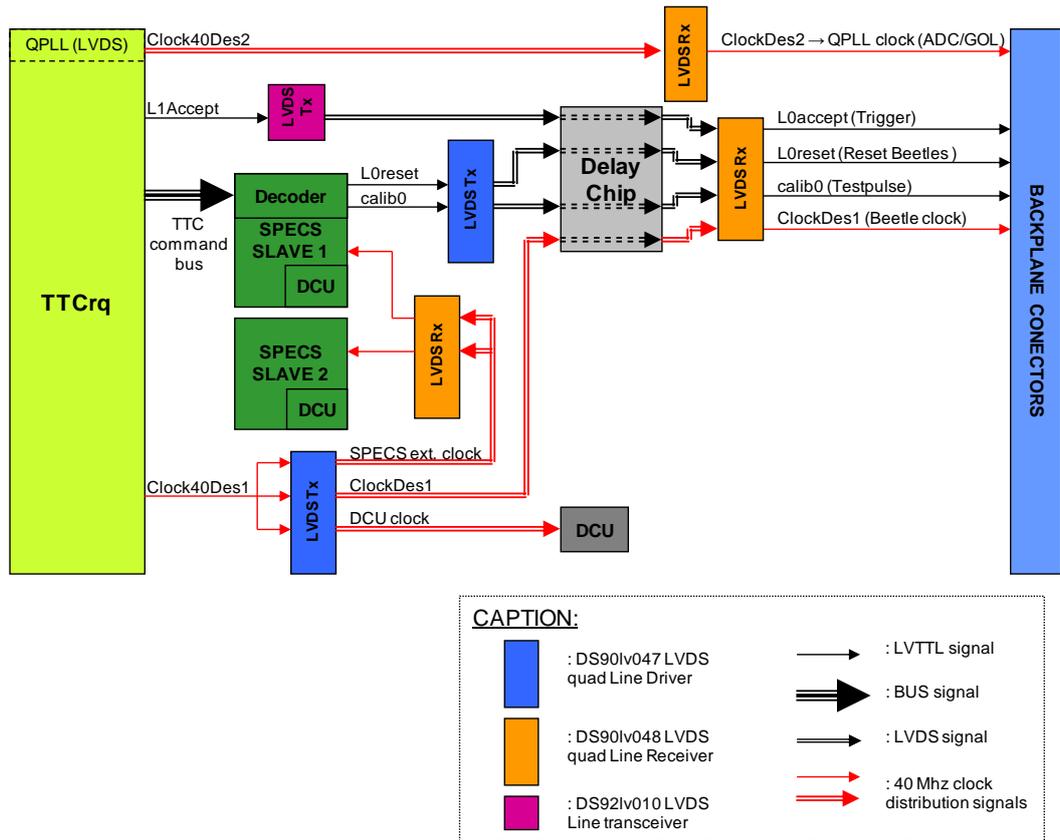
Figure 3.17: 40 MHz clock and TTC signals distribution diagram.

The TTCrq delivers two different clocks (*ClockDes1* and *ClockDes2*), the *L0accept* and the TTC channel B command bus from which the *L0reset* and the *calib0* signal are decoded in the SPECS FPGA. The phase of the *calib0* can be delayed within the SPECS in steps of 25 ns. All these signals but the *clockDes2* pass through the Delay25 chip so they can be phase delayed within a range of 32 ns in steps of 0.5 ns.

The TTC command bus and the *L0accept* signals are synchronized to the *clockDes1*, so this is the clock which must drive the SPECS slaves.

The Delay25 chip needs a reference clock for the signals that are to be phase aligned, so the *clockDes1* is used. This clock is also used for the DCU.

To distribute these signals, LVDS radiation tested transmitters and receivers situated close to the driving or receiving device have been used. The signals delivered by the TTCrq, those arriving at the SPECS decoder at the backplane connectors need to be 3.3 V CMOS/TTL[1] compatible, but a conversion in between to LVDS is done for several reasons:

- The difficulty to find reliable radiation tolerant devices for redistributing the clock signal to many devices. These LVDS transmitters and receivers have been qualified for radiation beyond the levels present in the SB.

- The track lengths of most of these signals is more than 25 cm, so using LVDS signaling avoids signal integrity and EMC/EMI problems that can easily appear in single ended mode, such as undesirable reflections that can cause EMI or increase clock jitter.

- The voltage compatibility. The TTCrx and the SPECS work at 3.3 V and are LVTTL compatible, but the Delay25 chip is only 2.5 V and LVDS compatible. So using LVDS makes the interconnection easier.

---

1 Except the *Clock40Des2* from the TTCrq QPLL which is LVDS.

- The DCU needs a differential 40 MHz clock.
- The *Clock40Des2* from the TTCrq is LVDS.

These LVDS drivers provide an enable/disabled line, so the interruption of the signals transmission can be controlled by the SPECS.

3.2.9   DELAY25 CHIP

The Delay25 is a 5-channel CMOS programmable delay line featuring 4 channels that allow phase delaying of periodic or non-periodic digital signals and a master channel that can be used to phase delay a clock signal. The phase of each channel can be independently programmed with a resolution of 0.5 ns in a range from 0 to 32 ns. The chip can be controlled and configured via I^2C.

The master channel, which serves as a calibration reference, is the *ClockDes1*. We use the delay chip to phase delay the *L0reset,* the *L0accept* and the *calib0* signal, so all of the TTC signals can be time aligned. The signal/channel assignment is the following:

- Channel 0 → empty
- Channel 1 → *L0reset*.
- Channel 2 → *L0accept*.
- Channel 3 → *calib0*.
- Clock channel → *ClockDes1*.

The Delay25 reset line, active at low level, is connected to a specific reset output of the SPECS slave1, as will be shown in the SPECS blocks diagram in section 3.2.15.

The input and output buffers of the channels can either work in single ended CMOS or in LVDS mode. We use it in LVDS mode as mentioned earlier. This is done by setting the *IOMODE* pin to "1".

The chip is connected to SPECS internal I^2C bus. The chip address is set to "b'0100XXX" as can be seen in the Figure 3.18. The "XXX" bits indicate which of the Delay25 chip internal register is being accessed.
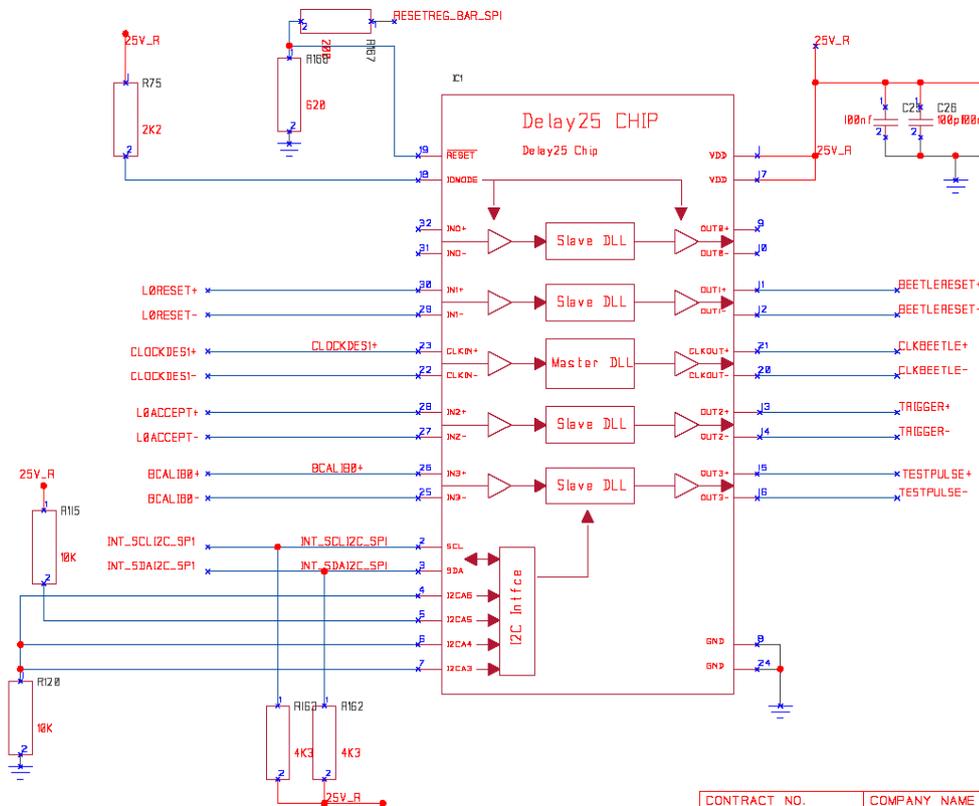
Figure 3.18: Delay25 chip schematic connections.

3.2.10 SPECS SLAVES

SPECS (Serial Protocol for the Experiment Control System) is a single master multi-slave bus, designed to allow a simple, fast and cheap means of communication between electronics systems. It is also designed to be efficiently protected against errors, and to be flexible.

The SPECS slaves are the devices that allow interfacing all our chips in the SB and detector box. They are implemented as mezzanine cards which provide several communication interfaces such as I^2C, JTAG and parallel bus, and also other features like a TTC channel B decoder, programmable I/O registers, etc. The list of features that a mezzanine can provide is next:

- One long distance point to point differential SPECS interface (coming from the SPECS master).
- One single ended SPECS local interface for multi-load bus applications.
- Two local I^2C buses of 2.5 V and 3.3 V.
- One output for long distance I^2C.
- One JTAG bus.
- $12 \times$ JTAG or $12 \times I^2C$ chip-select control bits for external drivers.
- $12 \times$ JTAG or $12 \times I^2C$ direction control bits for external drivers.
- One parallel bus offering 16 data bits and 8 address bits.
- One decoder for the channel B of the TTCrx for the signals:
  - *a.*   *L0reset* reset.
  - *b.*   Testpulse calibration (*calib0*), which can be delayed with a programmable counter.
- One 32-bit static register to control or read back the local environment. The bits [31:24] can be individually configured either in output or in input mode. In input mode, they may be configured as an interrupt vector, each of them being then able to generate a SPECS interrupt. The bits [23:0] can be individually programmed as well in input or output mode. This register does not need any external clock to be written by the SPECS. After a hardware reset, all I/Os are configured in input mode by default for safety reasons. Thus pull-up resistors may be necessary if some levels are mandatory at power-up.
- One reset signal. This output can be triggered without the need of any clock on the board.
- One local 40 MHz oscillator. It is also provided as an output of the mezzanine and can be enabled by the software.
- One EEPROM which will allow the ECS system to obtain some information about the front-end element housing the mezzanine. It can be read and written by one dedicated I^2C bus.
- One DCU chip with 6 ADC channels of 12-bit resolution and 1 non calibrated temperature channel.

Figure 3.19 represents the block diagram of the SPECS slaves use in the CB. The functionalities we use are the following:

- The internal I^2C of the Slave1 for interfacing the TTCrq, the Delay25 chip and the CB DCU.
- The Channel B decoder of the slave1.
- The long distance I^2C lines together with the first $8 \times I^2C$ chip-select and direction control bits of slave1.
- The single-ended SPECS local interface for multi-load bus applications of both slaves.
- The reset outputs of both slaves
- The I/O configurable register of both slaves.
- The DCU device of both slaves.

Although it's not show in the diagram, also the internal oscillator and the PROM will be used.

3.2.11 THE INTERNAL I^2C BUS

For onboard I^2C slaves, the mezzanine provides one 2.5 V bus: *scl_i2c_int* and *sda_i2c_int*.

The Delay25 chip, the DCU and the TTCrq hosted in the CB will be controlled using this internal I^2C bus. Both the Delay25 chip and the DCU cannot work with I^2C levels other than 2.5 V, so we use this bus with them. However, the TTCrq works at 3.3 V and this is the reason for the addition of a level shifter (see section 3.2.22). The possibility of directly using a 3.3 V I^2C bus coming from the SPECS slave has been left open. In principle two lines have been reserved in the SPECS mezzanine for a 3.3 V bus, so they have been routed to

a switch made of jumpers where we can chose either using the level shifter or the internal 3.3 V SPECS I^{2}C bus. By default the jumpers are set to use the level shifter (see Figure 3.20).

This bus must work at frequencies not much higher than 100 KHz, so attention must be paid when using the SPECS slaves internal I^{2}C bus which is set to work at 1 MHz by default. Neither the devices nor the level shifter and even the bus layout have been designed to work at high rates, so this restriction must always kept in mind when using the CB.
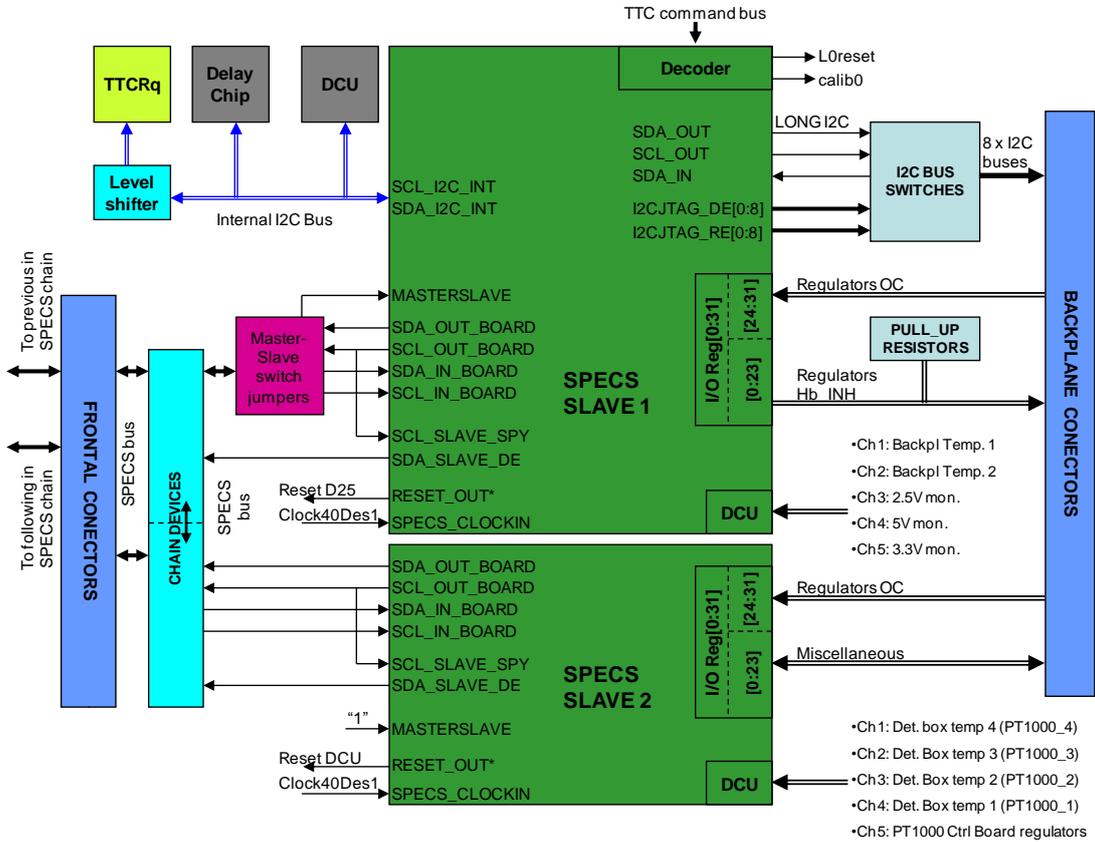
Figure 3.19: SPECS slaves block diagram.
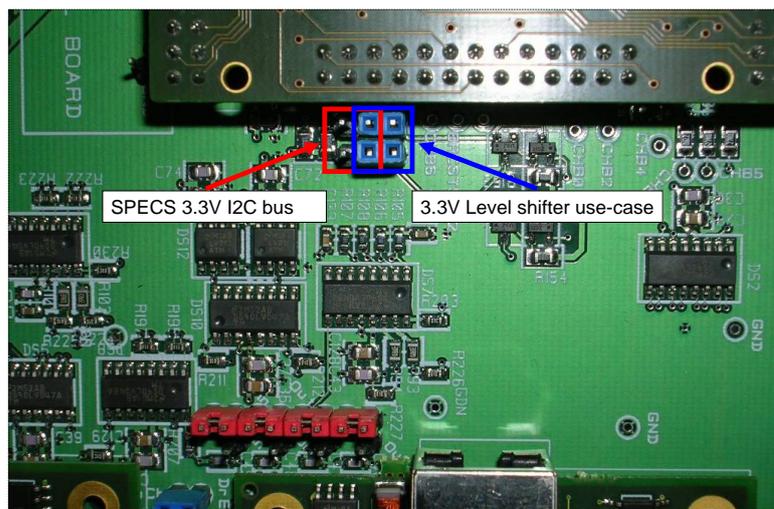
Figure 3.20: Jumper position for the selection of the TTCrq 3.3V I^{2}C bus. In the figure the jumpers are set to use the 3.3V I^{2}C level shifter (blue box position). To use the SPECS 3.3V I^{2}C bus they must be moved to the red box position.

3.2.12 THE CHANNEL B DECODER

The *Brcst[7:2]* and the two broadcast strobe signals *BrcstStr1* and *BrcstStr2* from the TTCrx are connected to the SPECS slave to decode the Channel B. Two commands are decoded:

1. *L0reset*, which is validated with the *BrcstStr1* strobe signal as shown in Figure 3.21. A pulse of 25 ns, synchronous to the failing edge of the SPECS clock (should be *Clock40Des1*) is delivered.
2. Bit '0' of the calibration command or *calib0*: this signal can be delayed with a programmable delay up to 255 cycles, in steps of 25 ns, as shown in Figure 3.22, by writing the value of the delay in the mezzanine register *CHANB_DELAY*. The signal is synchronous to SPECS clock which must also be the *Clock40Des1*. A pulse of one clock cycle is generated.
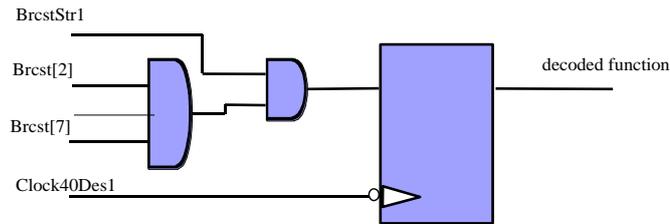
Figure 3.21: Channel B decoding diagram.
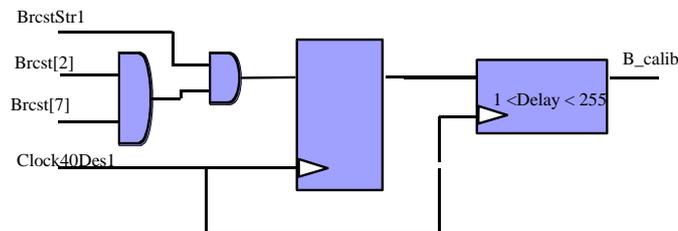
Figure 3.22: Calibration pulse decoding and delaying diagram.

It also implements the additional feature of enabling/disabling the output of each of the signals individually. This feature is mandatory for the calibration command because the calibration pulse injection command is a global TTC broadcast to all systems, and in case we do not disable it, it could affect the Beetles data while data taking since it will be pulsing them whenever a command is sent.

3.2.13 THE LONG DISTANCE I²C

One can implement on this board up to 12 long distance I²C buses. The SPECS system is seen in this case as an I²C provider, and its internal features can be completely ignored. The use of the long distance I²C provided by the SPECS slave is mandatory to communicate with boards situated in high radiation areas, like the Beetles, since it is not possible to put the SPECS FPGA in these locations.

The long distance I²C bus is based on 3 unidirectional signals: *scl_i2c*, *sda_i2c*, and *sda_i2c_in*. These signals must be buffered by drivers, and the equivalent of an open collector/drain circuit must be implemented. The frequency of the I²C signals can be selected by a register in the master and thus decreased for slow I²C slaves (1 MHz, 500 kHz, 250 kHz, 125 kHz, 65 kHz, 33 kHz, 16 KHz and 8 KHz). To implement these busses, 12 selection signals and 12 direction signals (*i2cjtag_de* and *i2cjtag_re*) are provided. The *i2cjtag_de* signals (but not the *i2cjtag_re*) can also be commanded statically (e.g. for external multiplexer commands).

The way this long I²C is used to command the Beetles, GOLs and DCUs is detailed in section 3.2.21.

3.2.14 THE SINGLE-ENDED SPECS LOCAL INTERFACE

On the SPECS mezzanine two SPECS interfaces are provided: one point to point for long distance interconnection and one for local bus connection. Thanks to this feature, we can handle several mezzanines on the same SPECS bus.

The mezzanine can be configured in master mode or in slave mode. In master mode, the mezzanine can deal with two SPECS busses:

- One point to point long distance bus. This bus is implemented with RJ45 connectors on the mezzanine, which permits a direct connection to the master through an Ethernet cable. It can also make use of the SMC connectors if the RJ45 plug cannot be used due to mechanical constraints.
- One multi-drop SPECS bus, located in the SMC connector. On this bus one can connect up to 31 slaves, respecting always the signal integrity on it.

In slave mode, the mezzanine controls the SPECS bus on the SMC connector, applying the same signal integrity rules as before.

It is usually recommended to place the drives close to the connectors to avoid signal integrity issues related to the SPECS signaling. The figures 3.23 and 3.24 show how to implement a multi-mezzanine SPECS bus using LVDS drivers.

The point to point distance bus implemented in the RJ45 connector of the mezzanine implements pole-zero compensation for long distance (100 m) communications, so using the mezzanine in master mode also permits getting profit of this extra feature. The LVDS driving stage of the mezzanine uses two SMD air coils (one for SDA and one for SCL) which act as a pre-emphasis filter for compensating the high frequency attenuation caused by the cable length, which improves the signal quality in the receiver.

The design of the CB SPECS chaining stage has been done in a way that makes possible the use either the slave1 mezzanine in master mode or slave mode, although for mechanical reasons the slave mode is recommended. A thorough explanation of this part is done in section 3.2.18.

3.2.15 THE RESET OUTPUTS

The asynchronous reset signal of the slave1 is used to reset the Delay25 chip, whereas the reset from the slave2 resets the CB DCU. The reset signal is active at low level. They can be used in case of chip malfunctioning.
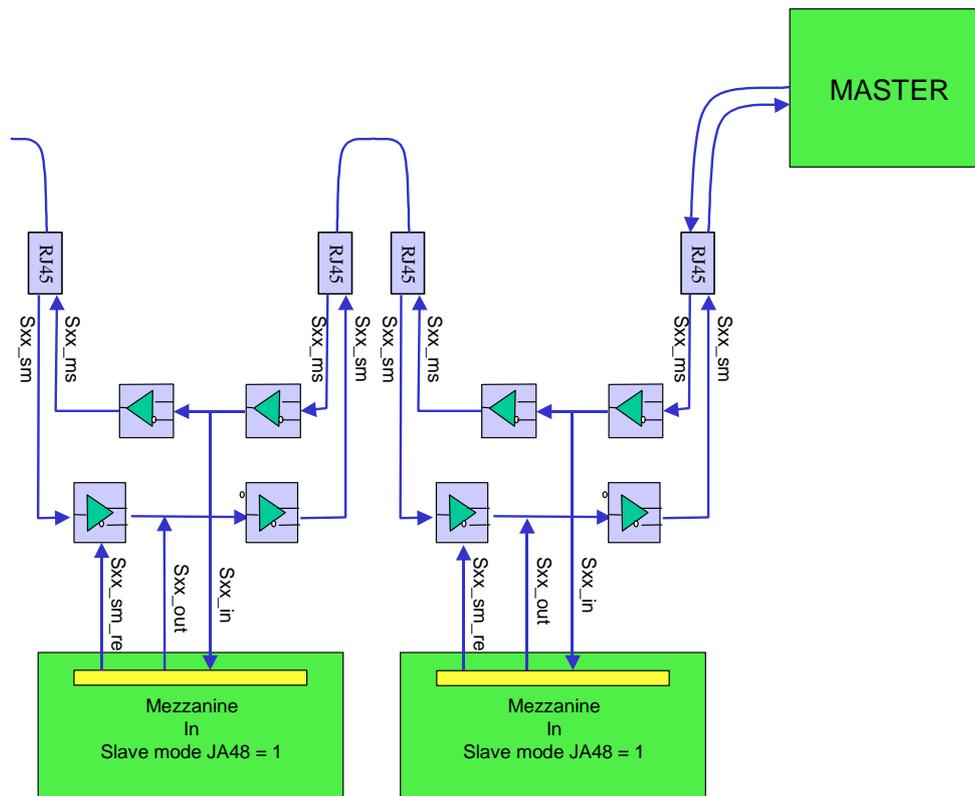
Figure 3.23: Chaining with all mezzanines in slave mode.
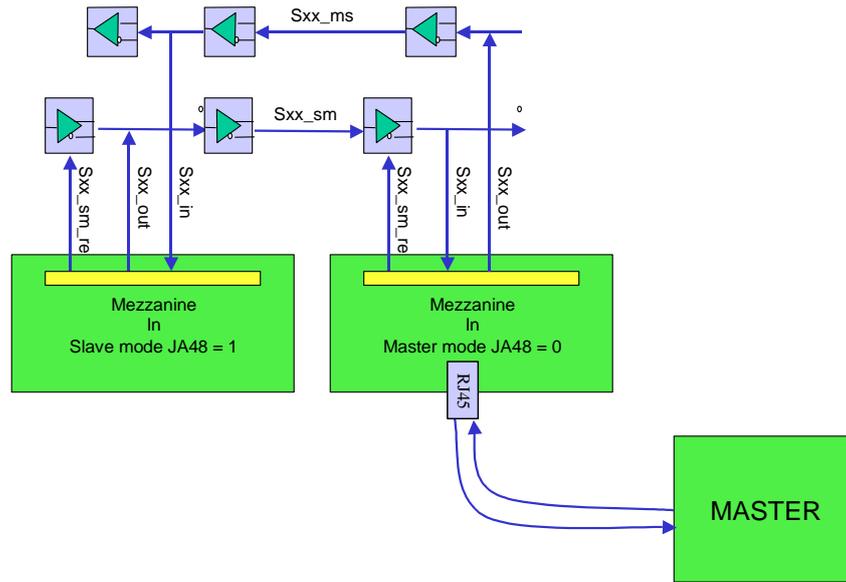
Figure 3.24: Chaining with one mezzanine in master mode.

3.2.16 THE I/O CONFIGURABLE REGISTER

On each mezzanine 32 user configurable I/O are implemented. They can be read or controlled by two 16 bit registers, *REGOUT_MSB* and *REGOUT_LSB*. These registers can be configured either in output or in input mode by two internal registers, *CONFREGOUT_LSB* and *CONFREGOUT_MSB*. By default (at power on or after a reset), the registers are configured in input mode to avoid any conflict with the signal. The logic is LVTTL (3.3 V).

The 32 I/O of the slave1 are basically used for the backplane regulators control, whereas the I/O lines from the slave2 are used for the regulators control and other miscellaneous purposes. The lines used for the regulators inhibit (*INH*) control are pulled-up to 2.5 V, hence they are disabled at start-up or after a slave reset. The lines for reading the *OCM* (Over Current Monitoring) signals of the 3.3 V and 5 V regulators are scaled using voltage dividers in order to assure high levels less than 3.3 V in the SPECS inputs.

The figures 3.25 and 3.26 show the distribution of the backplane regulators *OCM* and *INH* signals for the IT and TT in relation to the SB slot and partition number. The scheme and therefore the slot and partition numbering is done assuming the SB is seen from the frontal side (which means looking at the backplane from the back). It also shows the SPECS I²C bus numbering for the different partitions.

The I/O register signals' assignment for the IT and TT is the following (in parenthesis is indicated if the signal must be programmed as input or output):

IT I/O register assignment:	TT I/O register assignment:
Slave1 bit-signal assignment:	**Slave1 bit-signal assignment:**
1) Hb_INH_7 (O)	1) Not used in backplane
2) Hb_INH_6 (O)	2) Hb_INH_5 (O)
3) Hb_INH_5 (O)	3) Hb_INH_4 (O)
4) P1_DBs_INH (O)	4) P1_DBs_INH (O)
5) Hb_INH_4 (O)	5) Hb_INH_3 (O)
6) P0_DBs_INH (O)	6) P0_DBs_INH (O)
7) Hb_INH_2 (O)	7) Hb_INH_2 (O)
8) Hb_INH_3 (O)	8) Not used in backplane
9) Hb_INH_0 (O)	9) Hb_INH_0 (O)
10) Hb_INH_1 (O)	10) Hb_INH_1 (O)
11) Hb_INH_9 (O)	11) Hb_INH_7 (O)

IT I/O register assignment:	TT I/O register assignment:
12) Hb_INH_8 (O)	12) Hb_INH_6 (O)
13) Hb_INH_11 (O)	13) Not used in backplane
14) Hb_INH_10 (O)	14) Hb_INH_8 (O)
15) P2_DBs_INH (O)	15) P2_DBs_INH (O)
16) Hb_INH_12 (O)	16) Hb_INH_9 (O)
17) P3_DBs_INH (O)	17) P3_DBs_INH (O)
18) Hb_INH_13 (O)	18) Hb_INH_10 (O)
19) Hb_INH_14 (O)	19) Hb_INH_11 (O)
20) Hb_INH_15 (O)	20) Not used in backplane
21) TTCrq RESET (O)	21) TTCrq RESET (O)
22) TTCrq_QPLL RESET (O)	22) TTCrq_QPLL RESET (O)
23) QPLLs reset (O)	23) QPLLs reset (O)
24) RST_GOLs (O)	24) RST_GOLs (O)
25) P0_OC25V_#2 (I)	25) P0_OC25V_#2 (I)
26) P0_OC5V (I)	26) P0_OC5V (I)
27) P0_OC33V (I)	27) Not used in backplane
28) P0_OC25V_#1 (I)	28) P0_OC25V_#1 (I)
29) P1_OC5V (I)	29) P1_OC5V (I)
30) P1_OC25V_#2 (I)	30) P1_OC25V_#2 (I)
31) P1_OC25V_#1 (I)	31) P1_OC25V_#1 (I)
32) P1_C33V (I)	32) Not used in backplane
Slave2 assignment:	**Slave2 assignment:**
1) OCM_25V_CtrlBoard (I)	1) OCM_25V_CtrlBoard (I)
2) L0ACCEPT_DE_Ctrl (O)	2) L0ACCEPT_DE_Ctrl (O)
3) OCM_5V_CtrlBoard (I)	3) OCM_5V_CtrlBoard (I)
4) I2CSwitch_Ctrl4 (O)[2]	4) I2CSwitch_Ctrl4 (O) [2]
5) Amp_Con4 (I) [3]	5) Amp_Con4 (I) [3]
6) I2CSwitch_Ctrl3 (I) [2]	6) I2CSwitch_Ctrl3 (I) [2]
7) DRV_DOWN_EN (O)	7) DRV_DOWN_EN (O)
8) X9-C1 (I/O) [4]	8) X9-C1 (I/O) [4]
9) X9-A2 (I/O) [4]	9) X9-A2 (I/O) [4]
10) X9-B2 (I/O) [4]	10) X9-B2 (I/O) [4]
11) X9-A3 (I/O) [4]	11) X9-A3 (I/O) [4]
12) DRV_UP_EN (O)	12) DRV_UP_EN (O)
13) Not Connected in CB	13) Not Connected in CB

---

[2] These signals where foreseen to extend the control of the mezzanine board that implements the I^2C bus switches.

[3] These signals come from the TYCO-electronics connector in the front.

[4] These I/O have been routed to the backplane connector for any general purpose.

IT I/O register assignment:	TT I/O register assignment:
14) Not Connected in CB	14) Not Connected in CB
15) TTCrqREADY (I)	15) TTCrqREADY (I)
16) TTCrqLOCKED (I)	16) TTCrqLOCKED (I)
17) TTCrqERROR (I)	17) TTCrqERROR (I)
18) I2CSwitch_Ctrl2 (O) [2]	18) I2CSwitch_Ctrl2 (O) [2]
19) Amp_Con3 (I) [3]	19) Amp_Con3 (I) [3]
20) I2CSwitch_Ctrl1 (O) [2]	20) I2CSwitch_Ctrl1 (O) [2]
21) Amp_Con2 (I) [3]	21) Amp_Con2 (I) [3]
22) CtrlExt1 (O) [2]	22) CtrlExt1 (O) [2]
23) Amp_Con1 (I) [3]	23) Amp_Con1 (I) [3]
24) CtrlExt2 (O) [2]	24) CtrlExt2 (O) [2]
25) P3_OC33V (I)	25) Not used in backplane
26) P3_OC25V_#1 (I)	26) P3_OC25V_#1 (I)
27) P3_OC25V_#2 (I)	27) P3_OC25V_#2 (I)
28) P3_OC5V (I)	28) P3_OC5V (I)
29) P2_OC25V_#1 (I)	29) P2_OC25V_#1 (I)
30) P2_OC33V (I)	30) Not used in backplane
31) P2_OC5V (I)	31) P2_OC5V (I)
32) P2_C25V_#2 (I)	32) P2_C25V_#2 (I)

Slot numbering	8	9	10	11	12	13	14	15		0	1	2	3	4	5	6	7
Partition numbering	2				3					0				1			
5V OCM	P2_OC5V				P3_OC5V					P0_OC5V				P1_OC5V			
3.3V OCM	P2_OC33V				P3_OC33V					P0_OC33V				P1_OC33V			
2.5V OCM #1	P2_OC25V_#1				P3_OC25V_#1					P0_OC25V_#1				P1_OC25V_#1			
2.5V OCM #2	P2_OC25V_#2				P3_OC25V_#2					P0_OC25V_#2				P1_OC25V_#2			
Partition inhibit	P2_DBs_INH				P3_DBs_INH					P0_DBs_INH				P1_DBs_INH			
Bettle regulators Hb_INHibit signals. ON/OFF control of Hybrid Analog and Dig. Supply.	Hb_INH_8	Hb_INH_9	Hb_INH_10	Hb_INH_11	Hb_INH_12	Hb_INH_13	Hb_INH_14	Hb_INH_15	Ctrl Board position in Svce Box (Frontal view)	Hb_INH_0	Hb_INH_1	Hb_INH_2	Hb_INH_3	Hb_INH_4	Hb_INH_5	Hb_INH_6	Hb_INH_7
SPECS long I2C bus number for Beetles	4				6					0				2			
SPECS long I2C bus number for GOLs	5				7					1				3			

Figure 3.25: Logical distribution of the OCM and inhibit signals and also I^2C buses according to the slot and partition numbering shown (looking at the svce box from the frontal) for the IT subdetector.

Slot numbering	6	7	8	9	10	11	Ctrl Board position in Svce Box (Frontal view)	0	1	2	3	4	5
Partition numbering	2			3				0			1		
5V OCM	P2_OC5V			P3_OC5V				P0_OC5V			P1_OC5V		
2.5V OCM #1	P2_OC25V_#1			P3_OC25V_#1				P0_OC25V_#1			P1_OC25V_#1		
2.5V OCM #2	P2_OC25V_#2			P3_OC25V_#2				P0_OC25V_#2			P1_OC25V_#2		
Partition inhibit	P2_DBs_INH			P3_DBs_INH				P0_DBs_INH			P1_DBs_INH		
Bettle regulators Hb_INHibit signals. ON/OFF control of Hybrid Analog and Dig. Supply.	Hb_INH_6	Hb_INH_7	Hb_INH_8	Hb_INH_9	Hb_INH_10	Hb_INH_11		Hb_INH_0	Hb_INH_1	Hb_INH_2	Hb_INH_3	Hb_INH_4	Hb_INH_5
SPECS long I2C bus number for Beetles	4			6				0			2		
SPECS long I2C bus number for GOLs	5			7				1			3		

Figure 3.26: Logical distribution for the OCM and inhibit signals and also I²C buses according to the slot and partition numbering shown (looking at the svce box from the frontal) for the TT.

3.2.17 THE DCU

A DCU25F chip is housed on each mezzanine. A 10 Ω resistor is used in each input. The input range of the analog channels is 0 to 2.5 V and the reference resistor is set to 120 KΩ, giving a conversion factor of 2.1 mV^{-1}. Channel 7 is a non-calibrated internal temperature sensor.

The SPECS internal DCU channels are used to monitor the Detector Box and backplane temperatures, the control board regulators voltages and even an additional input has been foreseen for monitoring the regulators temperature.

DCU slave1:

- Channel 0 → Not used (10 µAcurrent source).
- Channel 1 → Backplane temperature 1.
- Channel 2 → Backplane temperature 2.
- Channel 3 → CB 2.5 V monitoring.
- Channel 4 → CB 5 V monitoring.
- Channel 5 → CB 3.3 V monitoring.

DCU slave2:

- Channel 0 → Not used (10 µAcurrent source).
- Channel 1 → Detector box temperature 4 (PT1000_4).
- Channel 2 → Detector box temperature 3 (PT1000_3).
- Channel 3 → Detector box temperature 2 (PT1000_2)
- Channel 4 → Detector box temperature 1 (PT1000_1).
- Channel 5 → PT1000 for the CB regulators temperature monitoring.

In section 3.2.19 a more detailed description about the PT1000 temperature sensors conditioning and readout can be found.

Figure 3.27 shows the simple circuit foreseen for an eventual SMD PT1000 sensor in the opposite side of the 3.3 V voltage regulator. Also 2 pins have been added so that instead of using a SMD sensor an aerial one attached to the heatsinks could be used by means of a small twisted cable.
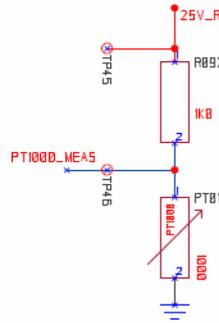
Figure 3.27: Extra PT1000 sensor and polarization circuit.

3.2.18 SPECS Daisy Chaining Circuitry

To daisy chain several SPECS slaves in the same bus, driving devices have to be employed in the Control Board. In our system we want to connect up to six control boards to the same link, which means up to twelve SPECS slaves in the same bus.

The four signals that connect the SPECS master to the slaves: *SDA_MS*, *SDA_SM*, *SCL_MS*, *SCL_SM* are LVDS, whereas the SPECS slaves FPGAs I/O are single ended, so a conversion from LVDS to TTL/CMOS in the CB is needed. Figure 3.28 shows the implementation using LVDS drivers and receivers. The diagram shows 3 different types of LVDS devices:

- The DS90LV047 (grey) and DS90LV048 (grey) are LVDS quad line driver and receivers respectively.
- The SN65LVDS32 (blue) is a quad-line receiver with a higher CMRR (Common Mode Rejection Ratio) than the DS90LV048, and for this reason it is used as the LVDS external link receiver in order to cope with common mode voltage differences between SPECS Master in the counting room and the slaves and also among the different SBs connected to the same SPECS link.
- The DS92LV010 (brown) is a BLVDS transceiver used in transmitter mode. Since it is BLVDS its current swing is twice than in LVDS, which is necessary in order to deal with the attenuation introduced by the 100 m cable between the cavern and the counting room.
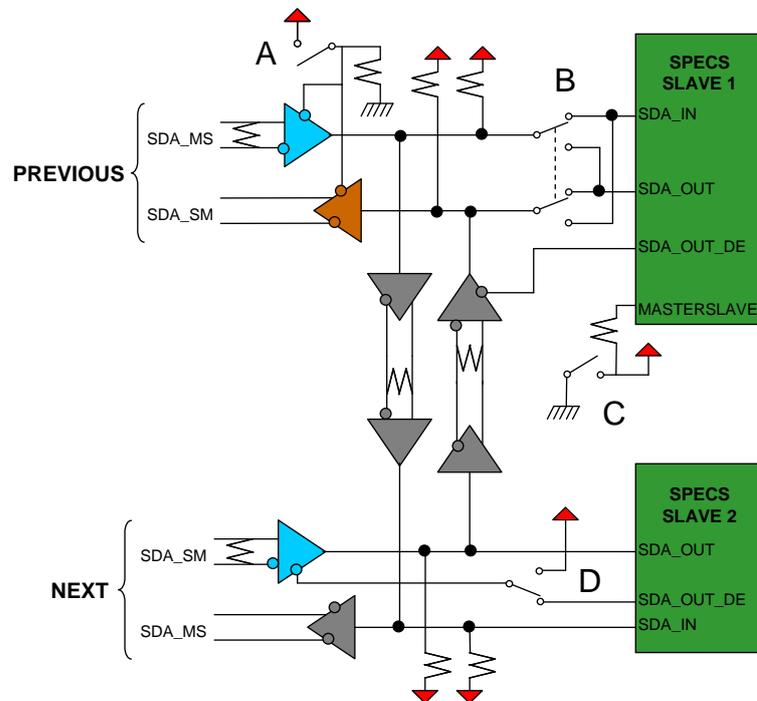
Figure 3.28: SPECS chaining block diagram for the SDA signal.

Several switches, which are done with jumpers, are also necessary:

- The "A" is to disable the LVDS receivers in case we decide to use the 1st slave in master mode, and therefore we connect the cable directly to the SPECS slave RJ45 connector.
- The "B" pair of switches is used to swap the interconnection of the *SDA_IN* and *SDA_OUT* signals depending on the *MASTERSLAVE* status of the SPECS slave. If we decide to use the 1st mezzanine in master mode the jumpers must be plugged as shown in Figure 3.29, while to work in slave mode they must be set as in Figure 3.30.
- The "C" is for setting the slave to work either in master mode or in slave mode. The slave 2 is by default set to slave mode.
- The "D" switch can be set to two different positions: the situation shown in the block diagram (and also in Figure 3.29) is the normal one and permits the slave "isolating" the link from the previous slaves in the chain so it can send a transmission to the master; in the other position (yellow box in Figure 3.30), the LVDS receiver is disabled, useful in case it is the last slave in the chain.
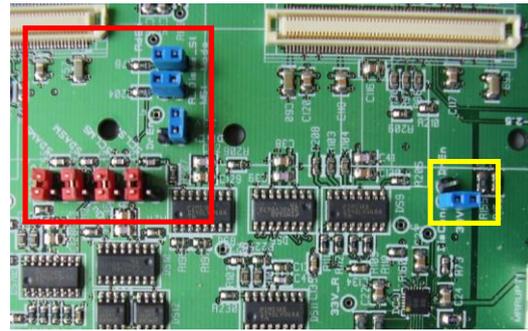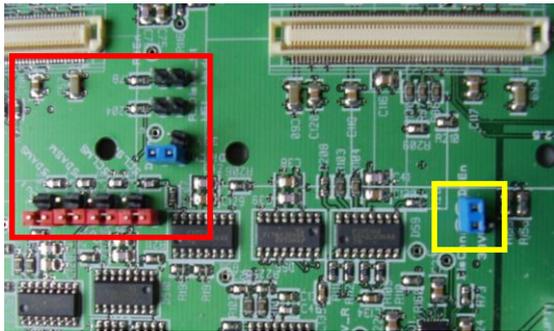
Figure 3.29: Inside de red box the jumper positions for using the slave1 in slave mode. In yellow the jumper position for enabling the link to other boards in the chain.

Figure 3.30: Within the red box the jumper positions for using the slave1 in master mode. In the yellow box the jumper position for "cutting" the SPECS link.

3.2.19 SIGNAL CONDITIONING DEVICES

Signal conditioning of the temperature and humidity signals before digitization is necessary in order to have a proper measurement. The temperature sensors signals are configured in a wheatstone bridge and amplified using the radiation tested AD8129 differential amplifier [20] (Figure 3.31). The humidity sensor is by itself a wheatstone bridge sensor, so it provides a differential output which is also amplified with the AD8129, but with a different gain (Figure 3.32).
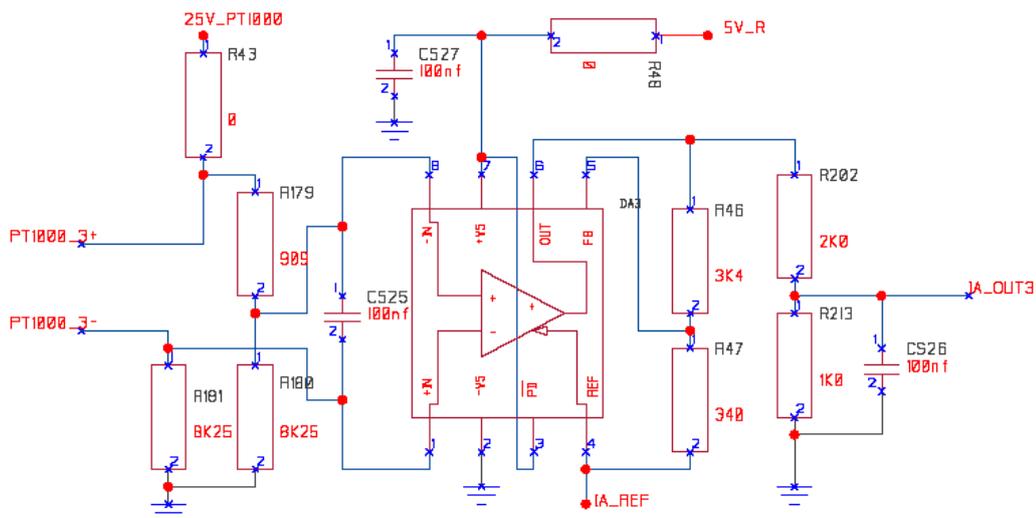
Figure 3.31: PT1000 sensor signal conditioning.
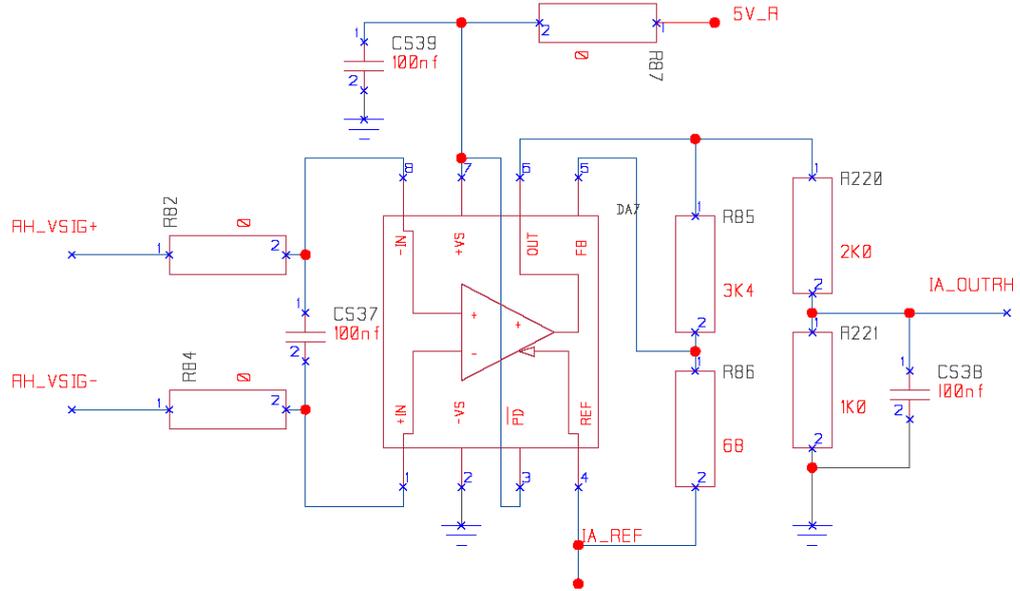
HMX2000 RH SENSOR CONDITIOING

Figure 3.32: HMX2000 bridge output signal conditioning circuit.

The radiation tested voltage reference AD780 [46] is used for the amplifiers reference voltage and also for the humidity sensor biasing. The voltage reference goes to a double toggle switch and then to the instrumentation amplifiers reference pins and humidity sensor bias line (Figure 3.33). This switch permits choosing between the voltage reference and the 2.5 V regulator supply. The 2.5 V from the regulator will give a less accurate measurement for several reasons: the regulator output voltage is noisier; the output voltage tolerance is higher because it is not precisely set to 2.5 V also shows a higher temperature dependent drift (bang-gap voltage references are auto-compensated).

The AD8129 is not a rail to rail differential amplifier, so the input and output voltage swing for any of the terminals is only guaranteed to work properly from 1.1 V to 3.9 V. The gain of the amplifier is set to 11 for the temperature signals and to 51 for the humidity sensor. The output of the instrumentation amplifier is scaled down 1:3 with a voltage divider which then connects to the DCU input. This voltage divider has been implemented in order to ensure that the input voltage at the DCU is always in the range from 0-1.25 V, so we use the DCU in LIR mode (Low Input Range) and also the DCU reading is referenced to ground.

The readout value for the temperature sensors is calculated as follows:

$\Rightarrow \quad V_{inDCU} = [(V_{in\,+} - V_{in\,-}) \times G + V_{ref}] \times A_{div}$

$\Rightarrow \quad V_{in\,+} = V_{25V\_PT1000} \times \left(\dfrac{R_{8K25}}{R_{8K25} + R_{909}} \right)$

$\Rightarrow \quad V_{in\,-} = V_{25V\_PT1000} \times \left(\dfrac{R_{8K25}}{R_{8K25\,+}\,R_{PT1000}} \right)$

$\Rightarrow \quad G = 1 + \dfrac{R_{3K4}}{R_{340}}$

$\Rightarrow \quad A_{div} = \dfrac{R_{2K0}}{R_{2K0} + R_{1K0}}$

$\Rightarrow \quad V_{ref} = 2.5V$

$\Rightarrow \quad V_{25V\_PT1000} = 2.5V$

Merging everything together we finally get:

$\Rightarrow \quad \boxed{R_{PT1000} = \dfrac{8250 \times (19158 - 10990,8 \times V_{inDCU})}{81591 + 10990,8 \times V_{inDCU}} \quad \Omega}$

And from PT1000 resistance we get the temperature value as follows:

$$\Rightarrow \quad \boxed{T_0 \; = \; \frac{(R_{PT1000} - 1000)}{3.85} \quad °C}$$

The HMX2000 is a radiation hard humidity sensor [45] tested and characterized by the CMS tracker group. The sensor output value does not follow a generic equation, so the manufacturer provides a sheet of calibration data for each sensor on demand (see appendix of [25]). The sensor output depends not only in the humidity, but also slightly on the environmental temperature. The output voltage also scales with the bias voltage and can go up to 5V, as stated by the company.

From the calibration data, and assuming temperatures from 5 to 10 ºC, we find that the lowest output voltage value is around -12.5 mV for 0% RH. As we will supply the sensor with 2.5 V instead of 1.25 V, the output scales to ~-25 mV, which multiplied by a gain of 51 produces and output value of -1.275 V. As the amplifier output is mounted on a 2.5 V offset, the resulting output value is 1.225 V which is above the 1.1 V minimum output voltage required for the amplifier. This output is divided by 3 in the voltage divider at the DCU input so we will work with the DCU in LIR mode.
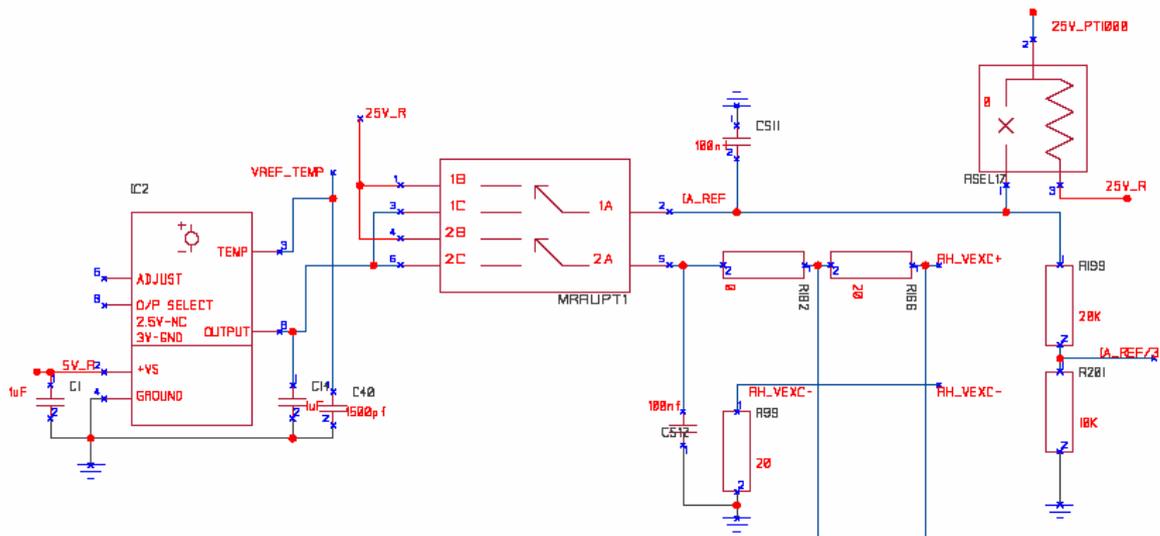
Figure 3.33: Schematic diagram of the voltage reference together with the toggle switch and the humidity sensor biasing resistors.

3.2.20 DCU

Together with the two DCUs of the SPECS slaves, the CB houses an extra DCU. This DCU is basically used for the readout of the HMX2000 humidity sensor:

- Channel 1 → *IA_IMEAS*: estimation of the current through the HMX2000 sensor.
- Channel 2 → *HIH_RH/4*: input downscaled by 4 for a HIH-3610 humidity sensor signal.
- Channel 3 → *IA_OUTRH*: humidity sensor signal after being conditioned.
- Channel 4 → *IA_REF/3*: voltage reference output (for monitoring the AD780) downscaled by 3.
- Channel 5 → *RH_VEXC-*: bias voltage in the negative terminal of the HMX2000.

As mentioned earlier, this DCU can be accessed using the SPECS internal I^2C bus. The DCU address is set to "b'x1100XXX", where the "XXX" bits are used for addressing the different DCU registers.

The reset line is connected to the slave2 external reset through a voltage divider. The external reference resistor is set to 120 KΩ, giving a conversion factor of 2.1 mV^{-1}. The 40 MHz differential clock needed by the DCU comes from the TTCrq.
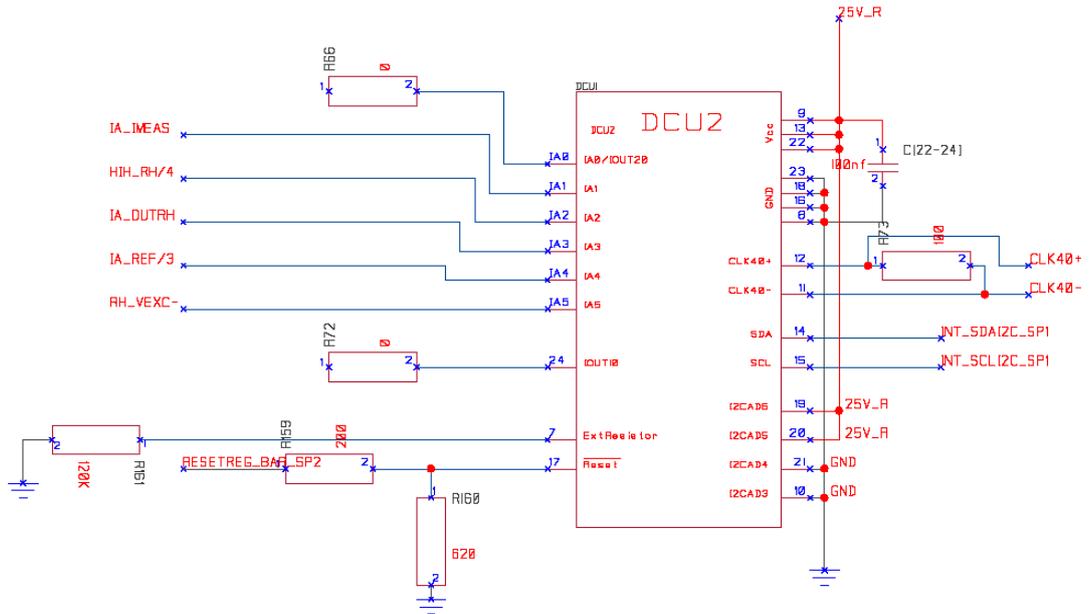
Figure 3.34: Control Board DCU schematic.

3.2.21 I²C BUS SWITCHES

The long I²C bus from the SPECS consists of three unidirectional lines for SDA and SCL signals and eight bus select lines. In the standard I²C, SDA and SCL are bidirectional lines (they are open drain or open collector), whereas the SPECS long I²C consists of *SDA_OUT*, *SDA_IN* and *SCL_OUT* (SCL_IN is not available since the SCL is an unidirectional clock line). Therefore interfacing circuitry is necessary in order to build eight I²C "standard compatible" buses from the SPECS signals.

From this SPECS long I²C we build four 3.3 V I²C buses that will go to the Beetles and four 2.5 V that will go to the DBs, based on a partitioning scheme described in the figures 3.25 and 3.26. The DB I²C needs to be 2.5 V because the GOLs and DCU are not compatible with other voltage levels. However, the hybrid links need necessarily to be 3.3 V since the Beetles I²C lines are 3.3 V compatible and some tests showed up that working at 2.5 V could compromise the reliability of the link connection due to the cable length and its associated capacitance and voltage drop along it.

This interface circuitry is implemented in a mezzanine card, so the CB features some pin connectors with the signals shown in Figure 3.35. In the left hand of the picture we can observe the mentioned signals *SDA_OUT*, *SDA_IN*, *SCL_OUT* and *I2C_DE_SP1[0:7]* (the bus selection signals) together with the power supply lines and some other control lines (*I2CSWITCH_CTRL[1:4]* and *CTRLEXT[1:2]*), and in the right hand we can find the I²C bus outputs that go to the hybrids and DBs.

The Figure 3.36 shows the mezzanine behavioral model in a normalized way. From the DEMUX block follows that all the outputs are tri-state and are connected either to the SDA or to the SCL lines. Each output pair SDA-SCL is also controlled by one of the *I2C_DE* input lines, and the odd pairs are also controlled by *EN* signals. The output lines are pulled-up to 3.3 V or 2.5 V depending on whether they connect to a Hybrid or to a DB. There are also two *CtrlExt* lines (8 and 9) which could also control the whole I²C DEMUX. As I²C is a bidirectional bus, feedback from SDA the outputs must be sent back to the SPECS *SDA_IN* input line, and this is done in the MUX block.

The aim of the *I2CSWITCH_CTRL[1:4]* lines is to pull to 0 V the GOLs I²C lines and thus avoid the GOL start-up problem. These lines should be pulled up or down in the mezzanine depending on the default initial state we desire at the SPECS start-up (at SPECS start-up all these lines are configured in input mode).

The *CTRLEXT1* and *CTRLEXT2* I/O lines have just been added in case additional functionalities need to be implemented in the I²C mezzanine.

The initial idea was to build this mezzanine using an anti-fuse radhard FPGA implementing the described behaviour. A first implementation in VHDL together with the logical and timing tests was done. Minor tests of this implementation have also been performed in an APA150 FPGA test-board, but not a real test with the actual hardware.
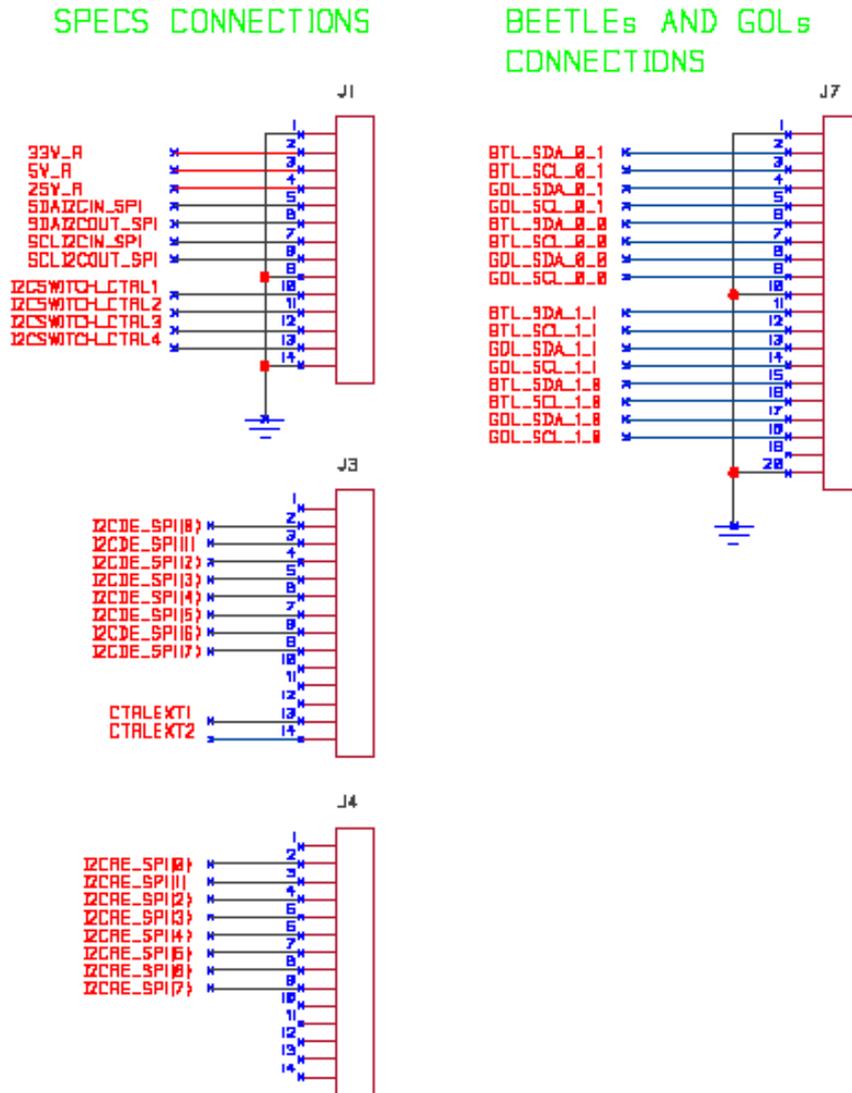
Figure 3.35: I²C bus MUX/DEMUX interface connectors.

In parallel to the FPGA mezzanine development a simple one based on tri-state drivers that fulfils all the basic requirements has been built. The Figure 3.37 shows the schematic for a SDA-SCL pair, and the figure Figure 3.38 the prototype PCB plugged in the CB. The entire mezzanine replicates by 8 the number of pairs keeping in mind that four of the output pairs are 3.3 V and the other four are 2.5 V. It uses the 74F125SC (bipolar technology) quad buffer tri-state drivers from 'Fairchild' and BSR117 NPN transistors for inverting the *I2C_DE* bus select lines. This tri-state based mezzanine just implements the basic functionalities. The components comprised in the dashed area implement the MUX while the other ones implement the DEMUX. The drawbacks of this mezzanine are:

- The tri-state drivers are known to work properly at least till 24.5 krad radiation dose and the maximum expected dose in the SB is ~15 krad, but applying safety factors, the devices should withstand till 60 krad. This means that for reliability reasons, they should be exchanged after ~5 years of normal operation.

- They do not implement GOL I²C lines disable capability. It should be also mentioned that the ST detector has hardly ever suffered the GOLs start-up problem for the long time that it has been running.

The initial idea of the FPGA based mezzanine has been put aside due to the quite limited manpower available. It still can be considered as a future upgrade of the tri-state based one.

3.2.22 LEVEL SHIFTER

The level shifter just consists of two head-to-head NPN transistors [47] working in saturation mode, together with the pull-up resistors mandatory for the I^{2}C bus. It is used to translate the I^{2}C voltage level from 2.5 V to 3.3 V, keeping in mind that the I^{2}C bus is bidirectional. The Figure 3.39 shows the scheme, where the 2.5 V I^{2}C lines are on the left side (*INT_SDAI2C_SP1* and *INT_SCLI2C_SP1*) and the 3.3 V ones on the right (*SDA_33V*, *SCL_33V*) and the 4K3 Ω pull-up resistor at both sides are not shown.

I^{2}C signals come from open-collector or open-drain outputs, so can only pull down. As the transistors base voltage is set to 2.5 V, when one of the input terminals goes low, the opposite transistor acts as a common base amplifier that saturates, pulling the other terminal low to within one V_{cesat}. This action also pulls the emitter of the input transistor low, to within one V_{cesat} of its collector, thereby cutting off collector current (although base current flows) and preventing latch-up.

For our application, BSR17A NPN switching transistors have been chosen, which allow working at rates of some hundreds of kHz, enough for TTCrq I^{2}C communication. The maximum V_{eb} that the transistor can withstand is 6 V, which is high enough since we will have 2.5 V in the base and 3.3 V in one of the emitters. The transistor has successfully been tested for radiation by the CERN AB-BI ('Accelerators and Beams Department – Beam instrumentation') up to 87.5 krad.
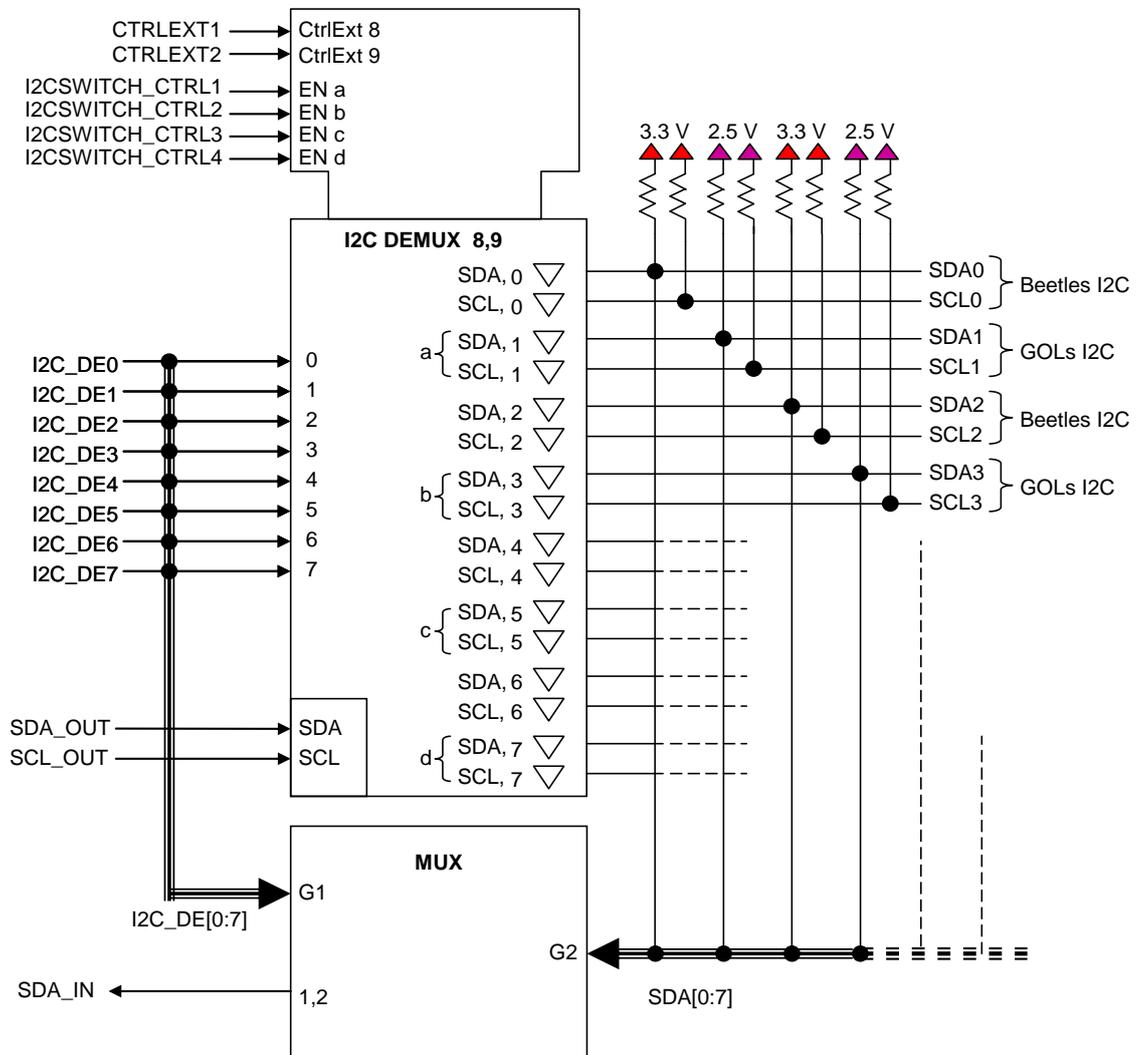
Figure 3.36: I^{2}C DEMUX and MUX normalized symbols.
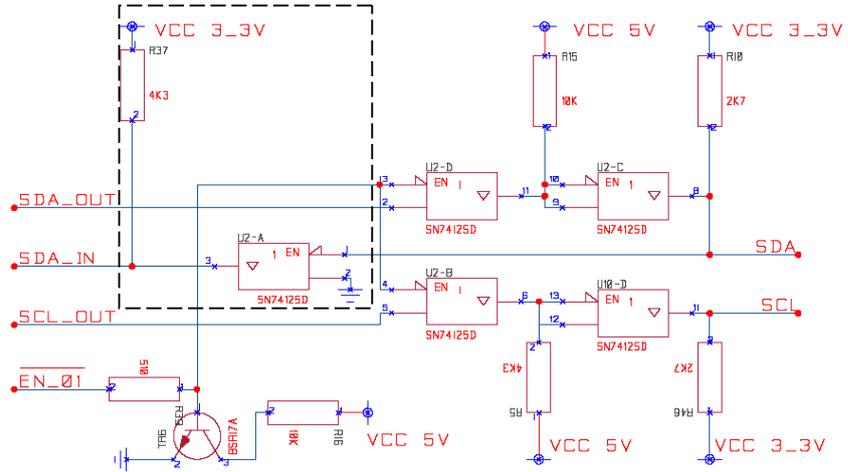
Figure 3.37: I^2C DEMUX and MUX implementation based on tri-state drivers.
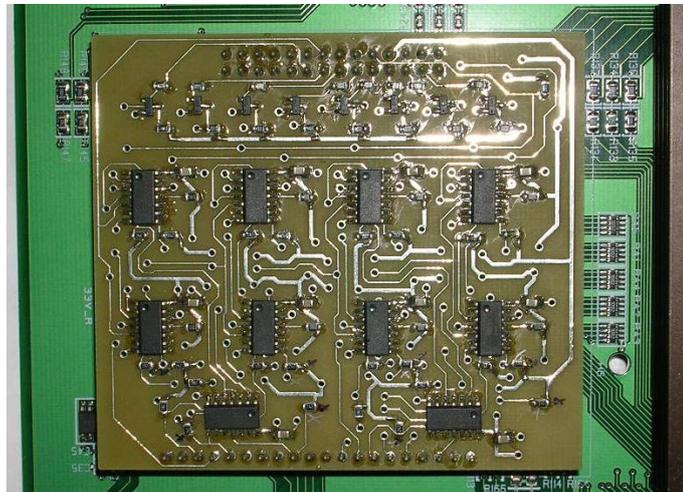
Figure 3.38: I^2C mezzanine based on 74F125SC tri-state drivers.
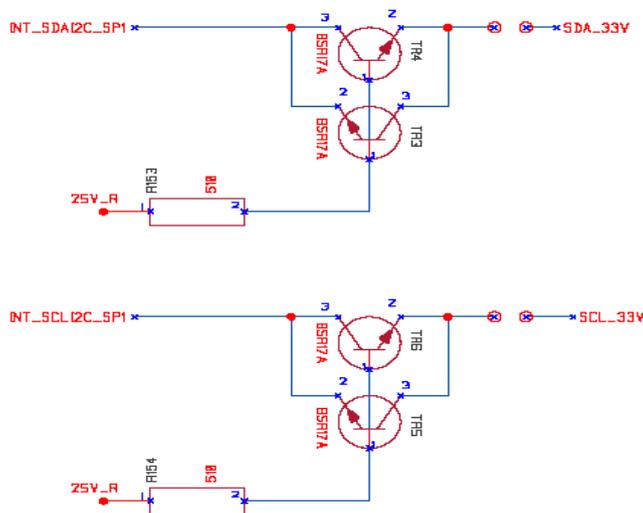
Figure 3.39 : I^2C level shifter schematic.

3.3   CONTROL BOARD PCB DESIGN AND CONSTRUCTION

The CB is produced in a multilayer technology with impedance controlled traces. The structure of the PCB is shown in Figure 3.40. The thicknesses were chosen to provide controlled impedance traces for the layers 2 and 5 (from top to bottom).

Copper layer from top to bottom:

1. Top Layer: signal layer flooded with ground copper.
2. Signal 1: signal layer.
3. PW: splitted power plane layer.
4. GND: ground layer.
5. Signal2: signals + fast signals.
6. Bottom layer: signal layer flooded with ground copper.

The images in the appendix D show the controlled impedance tracks, which are highlighted in red. Also the manufacturer track impedance solver results are shown.
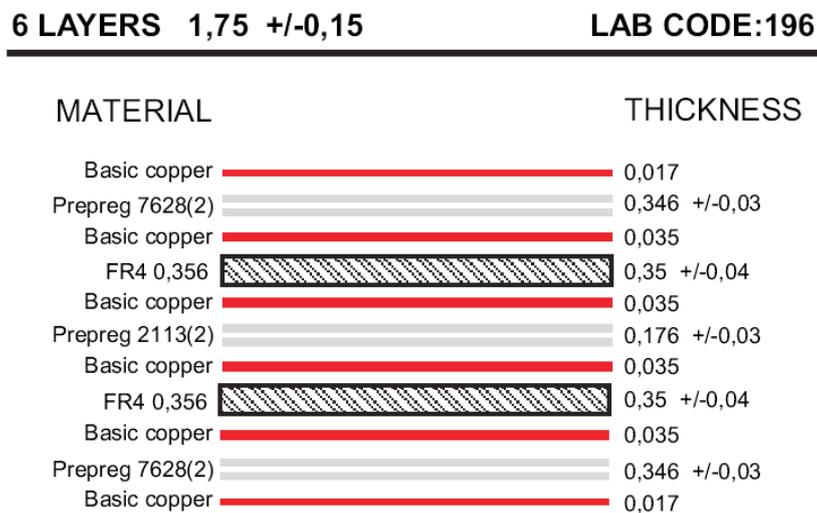
Figure 3.40: PCB structure (values in µm).

The drills are all of them through hole. Below is shown the drilling report for all the tools:

Plated through hole:

```
Drill No.  Tool No.    Size(mils)  Count   File Name
------------------------------------------------------------
        0          1      12.0     1288
        1          2      28.0      225
        2          3      32.0       55
        3          4      35.0      112
        4          5      39.0       70
        5          6      43.0        4
        6          7      51.0        4
        7          8     122.0        2
        8          9     125.0       10
------------------------------------------------------------
                          TOTAL     1770
```

Non plated through hole:

```
Drill No.  Tool No.    Size(mils)   Count   File Name
------------------------------------------------------------
        0          1      71.0        8
        1          2     126.0        8
------------------------------------------------------------
                          TOTAL       16
```

3.4 I2C MEZZANINE PCB

The I^{2}C mezzanine that plugs in the CB is a two layer PCB. The layout can be seen in the appendix E.

The drills are all plated through hole:

Plated through hole:

```
Drill No.  Tool No.    Size(mils)  Count   File Name
-----------------------------------------------------------
        0        1         20       167
        1        2    39.37007874   48
-----------------------------------------------------------
                            TOTAL    215
```

3.5 COMPONENTS RADIATION TOLERANCE

Due to the relatively high radiation levels present in the service boxes, some care must be taken when choosing the components for the CB. The devices response must be characterized for Total Ionizing Dose (TID) effects, for Single Event Effects (SEE) and Non Ionizing Energy Loss (NIEL), and they will have to survive for the lifetime of the experiment (expected to be ~10 years).

After 10 years of running at nominal LHCb luminosity, the following radiation levels are expected at the location of the ST service boxes:

LOCATION	TID (rad)	NIEL (1MeV equivalent neutrons)	SEE (>20 MeV hadron/cm^2)
T1 service box	$13 \cdot 10^3$	$1.2 \cdot 10^{12}$	$7.4 \cdot 10^{10}$
T2 service box	$14 \cdot 10^3$	$1.5 \cdot 10^{12}$	$8.1 \cdot 10^{10}$
T3 service box	$15 \cdot 10^3$	$1.6 \cdot 10^{12}$	$9.3 \cdot 10^{10}$
TT service box	$14 \cdot 10^3$	$2 \cdot 10^{12}$	$1.2 \cdot 10^{11}$

Table 3.3 : Expected radiation levels for ST service boxes.

Including the safety factors:

- Radiation qualification uncertainty → ×2
- TID/NIEL – component variations on the same fabrication line → ×2

The following radiation test levels have to be reached:

TID (rad)	NIEL (n/cm^2)	SEE (p/cm^2)
$60 \cdot 10^3$	$80 \cdot 10^{11}$	$4 \cdot 10^{11}$

Table 3.4 : Radiation levels to be reached.

Active components radiation assurance:

- TTCrq : CERN rad-hard development. The 2.5 V regulator must be removed from the board.
- Delay25: CERN rad-hard development.
- LHC4913 regulators: CERN rad-hard development.
- DCU: CERN rad-hard development.
- SPECS: tested successfully up to ~30 krad.
- AD8129 diff amplifier: tested successfully by the ST group.
- AD780 voltage reference: tested by NASA for TID (up to 100 krad) and Agapito et al. [46] for neutrons (up to $5 \cdot 10^{12}$, when it should withstand for safety up to $8 \cdot 10^{12}$). It has been mounted on DIP socket so it can be replaced.
- DS90LV047A and DS90LV048A LVDS transmitter and receiver: tested by Atlas and LHCb.
- DS92LV010A transceiver: tested by Atlas and LHCb.

- BSR17A ('Fairchild') NPN transistor: tested by Eva Calvo (CERN 'Accelerators and Beams Department – Beam instrumentation') till 80 krad in her WBTN (Wide Band Time Normalizer) board for the beam position monitoring.
- SN74125D quad tri-state gate: tested by the LHCb Orsay Calo group up to 24.5 krad.

3.6 CONTROL BOARD INSTALLATION PROCEDURE

For the installation of the CB in the SB some things should always be kept in mind.

- All the SPECS slaves in the same SPECS bus need to have different addresses. The address is set in the mezzanine by means of a micro-switch. As a general convention we can assign consecutive numbers from *1* to *N* to the mezzanines, with the address number increasing as the SPECS bus grows: 1 and 2 addresses for the 1st CB, 3 and 4 for the 2nd and so on. This implies that whenever a mezzanine is replaced, the same address as in the previous one must be set.
- The jumpers need to be plugged as explained in sections 3.2.11 (for connecting the TTCrq I^{2}C bus) and 3.2.18 (for the SPECS bus).
- The upper RJ45 in the CB is for connecting the cable coming from the master or the previous CB, and therefore the bottom one is for continuing the SPECS chain.
- In case the 1st mezzanine is used in master mode (option that up to now has been discarded), the RJ45 cable from the master will connect to the mezzanine connector. In this case, the resistors that connect the connector case to ground in the mezzanine must be removed. Like this the cable shield is just connected to ground in the master side, so a little bit of "bricolage" has to be done for connecting the cable shield to ground in the SB. Also the mezzanine needs to be fastened to the CB using screws in order to avoid cable tensions that could unplug it. For this reason is recommended to use all the mezzanines in slave mode, but in case signal integrity issues arise.
- Due to the SPECS daisy-chaining, whenever a CB has to be powered off, that is, the low voltage power supply lines for the SB, the whole SPECS link is lost since the SPECS chaining devices get the power from the CB. This implies loosing half station in the IT detector and one quadrant in the TT. Re-cabling of the SPECS link must be done as soon as access to the SB is granted.
- The cable shields must be connected to the SB case using the dedicated pins in the CB as shown in section 3.2.6.
- It is necessary to keep track of every installed HMX2000 sensor since their calibration curves are unique.
- In case the I^{2}C mezzanines described in 3.2.21 are not upgraded to the rad-hard FPGA version, for safety reasons they should be replaced after 5 years of experiment running.
- The same applies to the AD780 voltage reference, although in this case they can most probably survive the 10 years of running. Anyhow using the dual switch we can get rid of the AD780 and use the 2.5 V from the regulator though with a worse performance.
- A hole in the CB frontal has been foreseen for hooking a tool for easing the extraction of the board.
- The position where each board is installed must be written down as each has a unique calibration for the temperature and humidity DCU channels.
- The SPECS bus termination cable must be installed in the last CB in the chain.

CHAPTER 4

THE ST LV SYSTEM INFRASTRUCTURE

ABSTRACT

This chapter describes the technical details of LHCb Silicon Tracker Low Voltage (LV) infrastructure in a top to bottom approach.

"Un hombre que sabe que ha cometido un error y no lo corrige, está cometiendo otro error"

Confucio

4.1 INTRODUCTION

This chapter thoroughly describes the LV system of the IT and TT sub-detectors of the LHCb experiment from top to bottom. The information shown in this note is indispensable for anybody who wants to understand how the LV power distribution is like in the ST detectors and hence for any expert who wants to perform the development of the ST control system. Due to the nature of the information reported in this note, some hardware descriptions of the system and of parts of it will be necessary.

On one hand, a description of the solution adopted for the LV power distribution will be given. This comprises a description of the MARATON power supplies architecture and distribution and then a description of the power distribution in the SB side. A brief discussion about the low voltage power cables installed is given as well.

A detailed explanation on how to perform the operation of the detector from the ECS LV point of view is given in the appendix E. In that part all the technical details necessary for setting up the LV system will be presented.

4.2 THE ST LV SYSTEM

To power the ST electronics in the cavern (SB electronics + detector box hybrids), 3 regulated power voltage levels are needed: 2.5 V, 3.3 V and 5 V. In Table 4.1 the estimated LV power loads for both IT and TT service boxes are shown. The system has been designed in such a way that the commercial Wiener MARATON PS will provide the bulk power to single radiation hard regulators, type L4913 (developed by CERN microelectronics group) that are located on the backplane of the SB and will provide the voltages for the electronics.

	IT Service Box			TT Service Box		
	2.5 V	3.3 V	5 V	2.5 V	3.3 V	5 V
1 Hybrid	~0.75 A			~1 A		
1 Dig Board	~0.75 A		~0.4 A	~1 A		~0.5 A
Backplane		~0.8 A			0	
Ctrl Board	~0.2 A	~1 A	~0.2 A	~0.2 A	~1 A + 0.8 A	~0.2 A
Total Svce Box	~24.2 A	~1.8 A	~6.6 A	~24.2 A	~1.8 A	~6.2 A

Table 4.1: IT and TT service boxes power loads summary.

The voltage drop in the cables cannot be considered negligible in our system due to the cables length and to the considerable current loads that they will have to carry, so they have been carefully chosen.

4.2.1 THE LV MARATON SYSTEM

The MARATON system (Magnet field- and Radiation Tolerant New Power Supply System) is a low noise and low ripple modular power supply system developed to be located in hostile (radiation and magnetic field) environments. As shown in Figure 4.1 it consists of three separate parts:

1. Power box modules, located in the cavern, supplied by a high DC voltage of 385 V which they convert into the nominal voltages needed by the service boxes.
2. The primary rectifier: AD/DC + APFC board, located in the counting house, which provide the DC supply for the power box modules and the Active Power Factor Correction (for EMC issue).
3. The RCM: remote Control Module located in the counting house used for controlling and monitoring.

The AC/DC + APFC and the RCM are not tolerant to radiation and magnetic field, so they are located in the safe area of the counting house. For this reason, the settings of the modules output Voltage, Current Limit and the maximum module voltage is done by manually trimming of a potentiometer in the power modules. Limited remote control provided by the RCM is available for ON/OFF channel switching and for monitoring the voltage and current values as well as for setting voltage and current software limits. Two 36-line shielded cables are needed between each MARATON bin (containing 12 power modules) and each RCM, making a total of for 36 differential control signals.
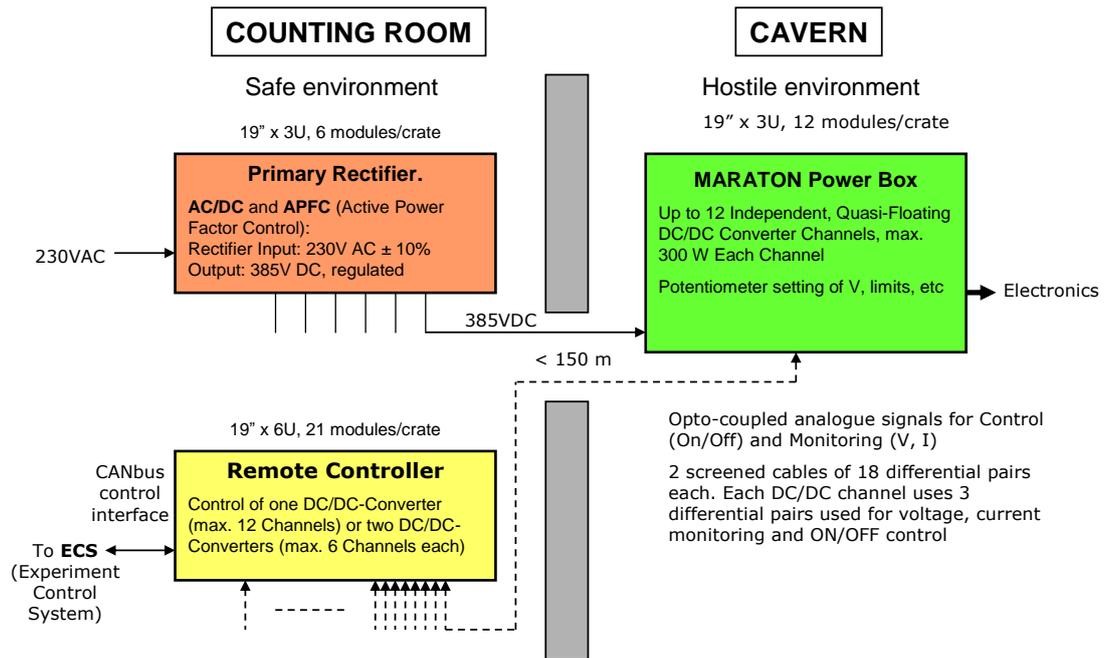
Figure 4.1: MARATON system overview.

The power box modules are located in the hostile area, and provide quasi-floating voltage channels with a channel to channel resistance greater than 10 kOhm, and a possible maximum channel to channel voltage of 100 V. These power modules are water cooled (air cooling is not possible since fans do not work properly in magnetic field environments) and all their inductive elements are completely shielded. The Power Box is limited to have as much, two different types of channels, one group in the first six slots and the other in the second six. Available modules that can be used and are of our interest are: 1-8 V @ 50 A (250 W) and 2-15 V @ 20 A (300 W).
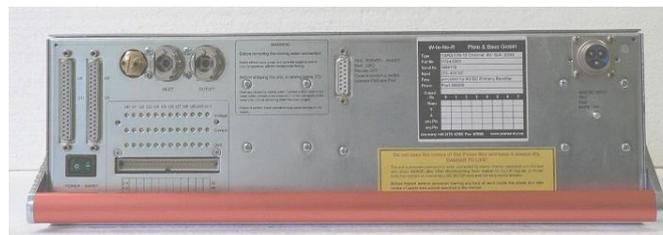
Figure 4.2: MARATON Power Box frontal view.

4.2.2   THE MARATON POWER SUPPLIES DISTRIBUTION

According to the IT and TT power requirements shown in Table 4.1, all of the ST power boxes will be equipped with two different types of power modules/channels: $6 \times$ 1-8 V @ 50 A and $6 \times$ 2-15 V @ 20 A modules as shown in Figure 4.2. The slots from 0 to 5 will provide the power for the 2.5 V regulators, whereas the slots from 6 to 11 will do it for the 3.3 V and 5 V.

The amount of power channels needed for the whole ST is 48 of each type. The IT consists of 24 service boxes so 24 channels of each type are necessary. For the TT 24 SBs need to be powered as well.

The Figure 4.4 shows the channels assignment of the four MARATON power boxes for the IT SBs. The 12 channels are represented looking at PS from the rear side, that is, from the power outputs. The IT consists of 3 stations, made up of 2 half-stations each: one for the access side and another for the cryo side. In terms of SBs, the half-station consists of 4 SBs, each of which will be powered by 2 MARATON channels as in the figure.
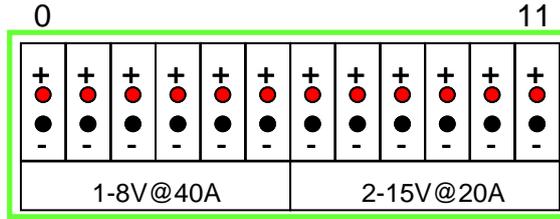
Figure 4.3: Power Boxes configuration viewed from the rear side.

For the TT the requirements are the same, as we can find from Figure 4.5. The only difference relies on sub-detector geometry, which consists of 4 quadrants, each of them with 6 SBs. As for the IT, 4 different MARATON Power bins are needed.

4.2.3   THE POWER REGULATORS DISTRIBUTION

The bulk power from the MARATON channels is "redistributed" in the SB backplane using low voltage power regulators of the type L4913 [40]. This is a special regulator developed in radiation hard technology by the CERN microelectronics group in collaboration with 'ST microelectronics'. The regulators that provide the necessary 2.5 V, 3.3 V and 5 V for the hybrids and DBs are allocated in the backplane, but the regulators for powering the CB have been allocated in the board itself due to the lack of space in the backplane.

4.2.4   POWER CONSUMPTIONS

The tables from 4.2 to 4.8 show a confident estimation of the current consumptions of the different boards. These estimations have been checked in the laboratory to be within a 10% of accuracy. These are the values that have been used for defining the final distribution of regulators that we will explain below, as well as for choosing the right power cables cross-section. It is important to remark that the value of current consumption of the TT backplane shown in Table 4.7 does not match with the one in Table 4.1 because the power is taken from the CB 3.3 V regulator due to the lack of space in the TT backplane for a dedicated 3.3 V regulator.
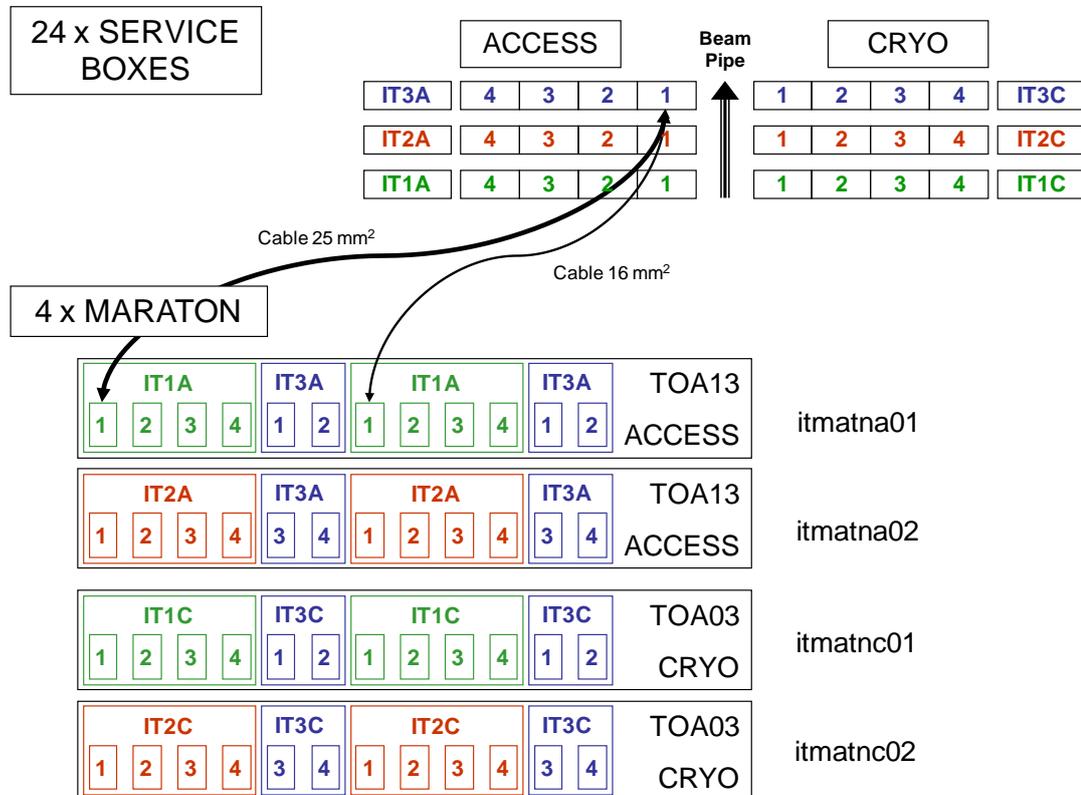
Figure 4.4: MARATON channels distribution on the IT sub-detector.
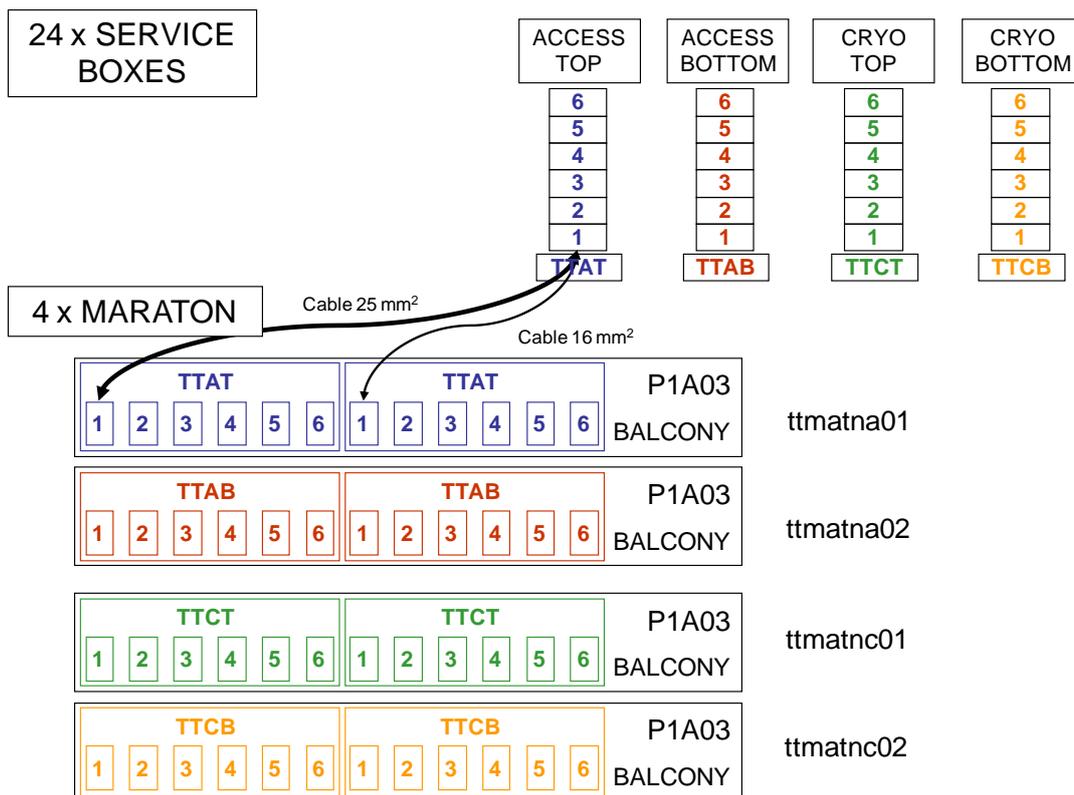
Figure 4.5: MARATON channels distribution on the TT sub-detector.

Board	Voltage [V]	Device	Consumption per device [A]	Devices per board	Consumption per board [A]	Total [A]
IT Hybrid	2,5 Dig	Beetle	0,050	3	0,150	0,15
	2,5 Ana	Beetle	0,200	3	0,600	0,6

Table 4.2: IT Hybrid current consumption.

Board	Voltage [V]	Device	Consumption per device [A]	Devices per board	Consumption per board [A]	Total [A]
TT Hybrid	2,5 Dig	Beetle	0,050	3	0,150	0,15
	2,5 Ana	Beetle	0,200	3	0,600	0,6

Table 4.3: TT Hybrid current consumption.

Board	Voltage [V]	Component	Current per device [A]	Components per board	Comsumption per board [A]	Total [A]
IT DB	2,5	TSA0801 ADC	0,016	12	0,192	0,732
		GOL serializer	0,160	3	0,480	
		DCUF	0,012	1	0,012	
		QPLL	0,040	1	0,040	
		Ex64	0,008	1	0,008	
	5	AD8129 diff rx	0,011	13	0,143	0,338

Table 4.4: IT Digitizer Board current consumption.

Board	Voltage [V]	Component	Current per device [A]	Components per board	Comsumption per board [A]	Total [A]
TT DB	2,5	TSA0801 ADC	0,016	12	0,192	0,732
		GOL serializer	0,160	3	0,480	
		DCUF	0,012	1	0,012	
		QPLL	0,040	1	0,040	
		Ex64	0,008	1	0,008	
	5	AD8129 diff rx	0,011	13	0,143	0,338

Table 4.5: IT Digitizer Board current consumption.

Board	Voltage [V]	Device	Consumption per device [A]	Devices per board	Consumption per board [A]	Total [A]
IT Backpl.	3,3	DS90LV047A	0,020	30	0,600	0,744
		DS90LV048A	0,009	16	0,144	

Table 4.6: IT backplane current consumption.

Board	Voltage [V]	Device	Consumption per device [A]	Devices per board	Consumption per board [A]	Total [A]
TT Backpl.	3,3	DS90LV047A	0,020	26	0,520	0,628
		DS90LV048A	0,009	12	0,108	

Table 4.7: TT backplane current consumption.

Board	Voltage [V]	Component	Current per device [A]	Components per board	Comsumption per board [A]	Total [A]
Control board	3,3	TTCrx	0,060	1	0,060	0,929
		SPECS	0,300	2	0,600	
		DS92LV047A	0,020	5	0,100	
		DS92LV048A	0,020	5	0,100	
		DS92LV010	0,013	1	0,130	
		SN65LVDS32B	0,023	2	0,460	
		I2CMEZZ	0,010	1	0,010	
	5	AD8129 diff rx	0,011	13	0,143	0,313
		AD780	0,010	1	0,010	
		I2CMEZZ	0,140	1	0,140	
		TTCrx	0,020	1	0,020	
	2.5	DCU2	0,012	1	0,012	0,172
		Delay25 Chip	0,030	1	0,030	
		I2CMEZZ	0,100	1	0,080	
		TTCrx	0,050	1	0,050	

Table 4.8: Control Board current consumption.

4.2.5 REGULATORS DISTRIBUTION

The power regulators distribution in the backplane has been done taking into account the general partitioning of the detector [48] as well as the maximum current that can be regulated by a single regulator. For the IT detector, the partitioning has been done in such a way that groups of four boards are always operated together, while for the TT the groups are comprised of three boards. Each FE hybrid has its own set of two regulators (analogue and digital) with their associated sense lines to supply 2.5 V to the Beetle readout chips. This permits to switch off individual hybrids (Figure 4.6). For an efficient use of the regulators, it was decided to supply more than one DB from a single regulator. According to the current requirements, as listed in Table 4.4, it is possible to supply two DBs from a single 2.5 V regulator and up to five DBs from a 5 V regulator.

For the IT a regulator set supplying a group of four DBs/hybrids contains (see Figure 4.7):

- $4 \times$ analogue and $4 \times$ digital 2.5 V regulators (four hybrids), max. 0.6/0.15 A each.
- $2 \times$ 2.5 V regulators (deep submicron devices of four Digitizer Boards), 1.5 A each.
- $1 \times$ 3.3 V regulator (backplane clock branch), 185 mA.
- $1 \times$ 5 V regulator (line receivers and clock tree of four Digitizer Boards), 1.6 A.

For the TT, as the groups are made up of three boards an there was not enough space in the backplane, it was decided to remove the 3.3 V regulator from the backplane and get the power from the CB. As a result we get (see Figure 4.8):

- $3 \times$ analogue and $3 \times$ digital 2.5 V regulators (three hybrids), max. 0.8/0.2 A each
- $2 \times$ 2.5 V regulators (deep submicron devices of four Digitizer Boards), 2 A each.
- $1 \times$ 5 V regulator (line receivers and clock tree of four Digitizer Boards), 1.5 A.

Keeping in mind that each SB affords for 4 partitions of hybrids/DBs, the total number of regulators used in the IT backplane is $4 \times 12 = 48$, and in the same manner it is $4 \times 9 = 36$ for the TT.

As mentioned earlier, the power regulators for the CB have not been allocated in the backplane due to the lack of space on it, so it is the board itself that contains three regulators for supplying 2.5 V, 3.3 V and 5 V.

4.2.6 REGULATORS CONTROL AND MONITORING LINES

The inhibit control (*INH*) of all the backplane regulators is performed using some dedicated CB I/O lines. However the over current status (*OCM*) signals are monitored both by dedicated I/O and by the Digitizer Board DCU channels.
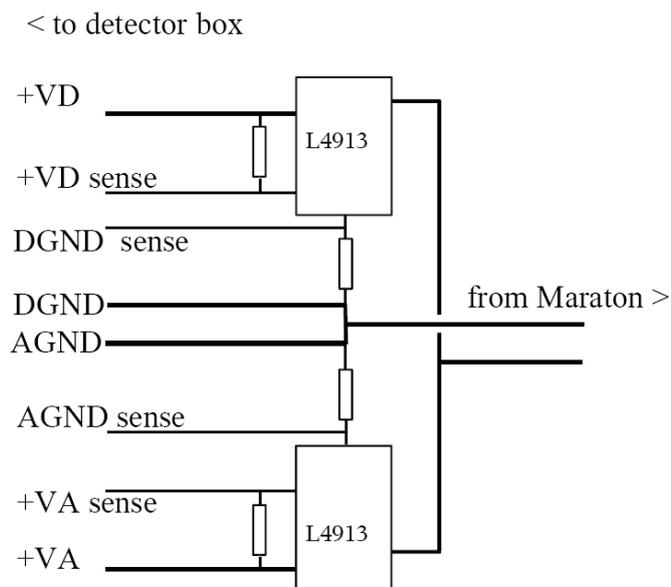
Figure 4.6: The use of the radiation hard regulators for 2.5V supply of the hybrids.

CTRL BOARD I/O lines

OCM: Over Current Monitoring
INH: Inhibit
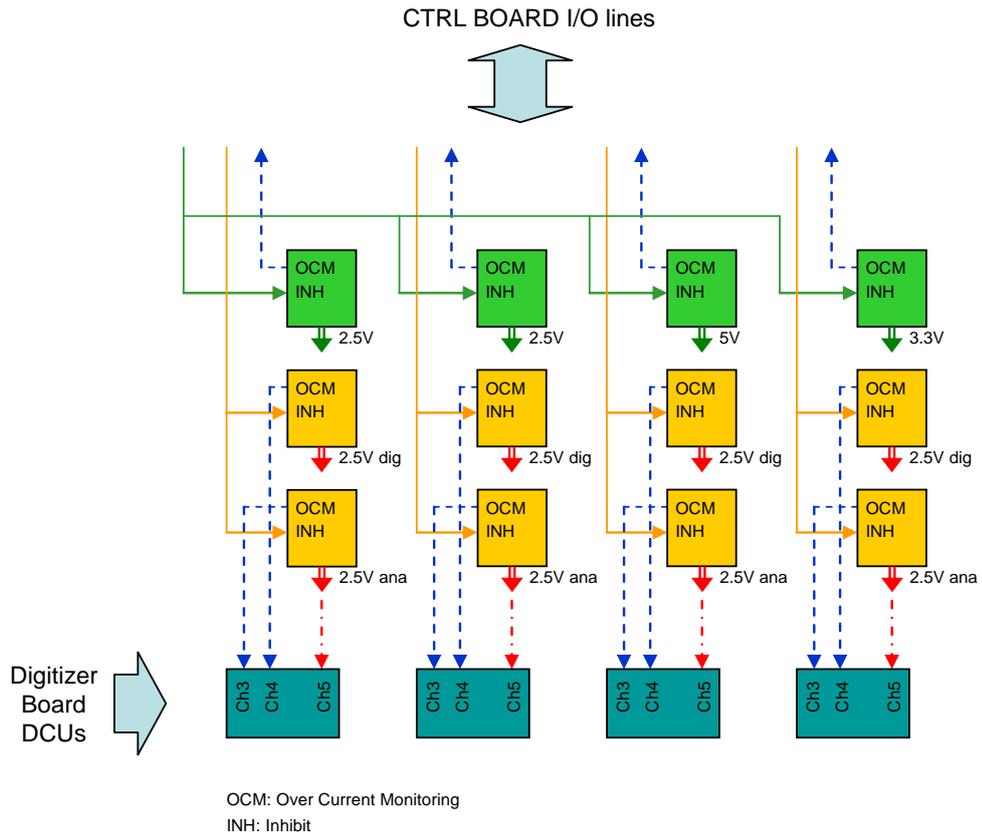
Figure 4.7: Scheme of a regulators set for a partition of 4 hybrids/DBs in the IT detector.

CTRL BOARD I/O lines

OCM: Over Current Monitoring
INH: Inhibit
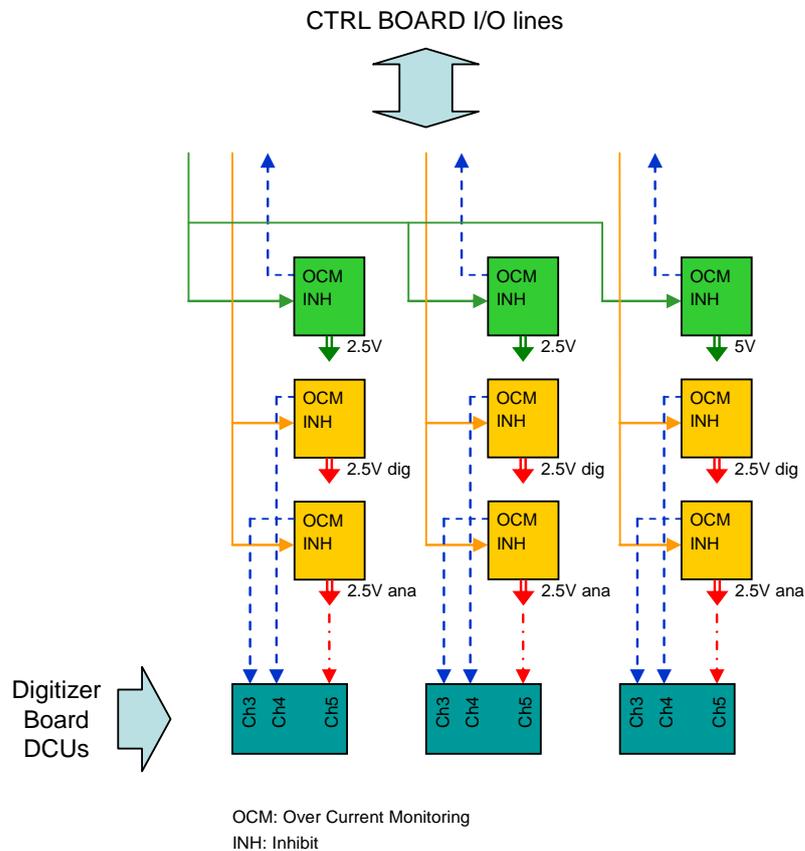
Figure 4.8: Scheme of a regulators set for a partition of 3 hybrids/DBs in the TT detector.

- There is a single line to switch off the regulators powering a whole partition of Digitizer Boards (4 in the IT detector and 3 in the TT) and one line per hybrid.

- The *OCM* lines for the regulators powering the DBs are monitored using the CB I/O lines, whereas the *OCM* lines for the hybrid regulators are monitored by the corresponding DB DCU channel. The DCU hosted in the DB will monitor the regulators of the hybrid connected to it, as it would make sense.

- The DCU also monitors the voltage value of the 2.5 V analog line.

The I/O lines of the CB are provided by the 2 SPECS slaves allocated in the CB. On each slave 32 user configurable I/O are implemented. There are two 16-bit registers to write and read the value in these 32 lines, and another 2 to configure the lines either in input or output mode. By default (at power on or after a reset), the registers are configured to input mode to avoid any conflict with the signal. The lines assignment is given in Table 4.9.

Figures 4.9 and 4.10 show the distribution of the backplane regulators *OCM* and *INH* signals for the IT and TT in relation to the SB slot and partition number. The representation, and therefore the slot and partition numbering has been done assuming that the SB is observed from the frontal side (which means looking at the backplane from the back).

The lines used for the regulators inhibit (*INH*) control are pulled-up to 2.5 V, hence the regulators are disabled at start-up or after a slave reset.

IT I/O register assignment:	TT I/O register assignment:
Slave1 bit-signal assignment:	**Slave1 bit-signal assignment:**
1. Hb_INH_7 (O)	1. Not used in backplane
2. Hb_INH_6 (O)	2. Hb_INH_5 (O)
3. Hb_INH_5 (O)	3. Hb_INH_4 (O)
4. P1_DBs_INH (O)	4. P1_DBs_INH (O)
5. Hb_INH_4 (O)	5. Hb_INH_3 (O)
6. P0_DBs_INH (O)	6. P0_DBs_INH (O)
7. Hb_INH_2 (O)	7. Hb_INH_2 (O)
8. Hb_INH_3 (O)	8. Not used in backplane
9. Hb_INH_0 (O)	9. Hb_INH_0 (O)
10. Hb_INH_1 (O)	10. Hb_INH_1 (O)
11. Hb_INH_9 (O)	11. Hb_INH_7 (O)
12. Hb_INH_8 (O)	12. Hb_INH_6 (O)
13. Hb_INH_11 (O)	13. Not used in backplane
14. Hb_INH_10 (O)	14. Hb_INH_8 (O)
15. P2_DBs_INH (O)	15. P2_DBs_INH (O)
16. Hb_INH_12 (O)	16. Hb_INH_9 (O)
17. P3_DBs_INH (O)	17. P3_DBs_INH (O)
18. Hb_INH_13 (O)	18. Hb_INH_10 (O)
19. Hb_INH_14 (O)	19. Hb_INH_11 (O)
20. Hb_INH_15 (O)	20. Not used in backplane
21. **************	21. **************
22. **************	22. **************
23. **************	23. **************
24. **************	24. **************
25. P0_OC25V_#2 (I)	25. P0_OC25V_#2 (I)

IT I/O register assignment:	TT I/O register assignment:
26. P0_OC5V (I)	26. P0_OC5V (I)
27. P0_OC33V (I)	27. Not used in backplane
28. P0_OC25V_#1 (I)	28. P0_OC25V_#1 (I)
29. P1_OC5V (I)	29. P1_OC5V (I)
30. P1_OC25V_#2 (I)	30. P1_OC25V_#2 (I)
31. P1_OC25V_#1 (I)	31. P1_OC25V_#1 (I)
32. P1_C33V (I)	32. Not used in backplane
Slave2 assignment:	**Slave2 assignment:**
1. OCM_25V_CtrlBoard (I)	1. OCM_25V_CtrlBoard (I)
2. **************	2. **************
3. OCM_5V_CtrlBoard (I)	3. OCM_5V_CtrlBoard (I)
4. **************	4. **************
5. **************	5. **************
6. **************	6. **************
7. **************	7. **************
8. **************	8. **************
9. **************	9. **************
10. **************	10. **************
11. **************	11. **************
12. **************	12. **************
13. Not Connected in CB	13. Not Connected in CB
14. Not Connected in CB	14. Not Connected in CB
15. **************	15. **************
16. **************	16. **************
17. **************	17. **************
18. **************	18. **************
19. **************	19. **************
20. **************	20. **************
21. **************	21. **************
22. **************	22. **************
23. **************	23. **************
24. **************	24. **************
25. P3_OC33V (I)	25. Not used in backplane
26. P3_OC25V_#1 (I)	26. P3_OC25V_#1 (I)
27. P3_OC25V_#2 (I)	27. P3_OC25V_#2 (I)
28. P3_OC5V (I)	28. P3_OC5V (I)
29. P2_OC25V_#1 (I)	29. P2_OC25V_#1 (I)
30. P2_OC33V (I)	30. Not used in backplane
31. P2_OC5V (I)	31. P2_OC5V (I)
32. P2_C25V_#2 (I)	32. P2_C25V_#2 (I)

Table 4.9: SPECS slaves INH and OCM lines assignment for IT and TT.

Slot numbering	8	9	10	11	12	13	14	15		0	1	2	3	4	5	6	7
Partition numbering	2				3					0				1			
5V OCM	P2_OC5V				P3_OC5V					P0_OC5V				P1_OC5V			
3.3V OCM	P2_OC33V				P3_OC33V					P0_OC33V				P1_OC33V			
2.5V OCM #1	P2_OC25V_#1				P3_OC25V_#1					P0_OC25V_#1				P1_OC25V_#1			
2.5V OCM #2	P2_OC25V_#2				P3_OC25V_#2					P0_OC25V_#2				P1_OC25V_#2			
Partition inhibit	P2_DBs_INH				P3_DBs_INH					P0_DBs_INH				P1_DBs_INH			
Bettle regulators Hb_INHibit signals. ON/OFF control of Hybrid Analog and Dig. Supply.	Hb_INH_8	Hb_INH_9	Hb_INH_10	Hb_INH_11	Hb_INH_12	Hb_INH_13	Hb_INH_14	Hb_INH_15	Ctrl Board slot in Svce Box (Frontal view)	Hb_INH_0	Hb_INH_1	Hb_INH_2	Hb_INH_3	Hb_INH_4	Hb_INH_5	Hb_INH_6	Hb_INH_7

Figure 4.9: Logical distribution for the OCM and inhibit signals according to the slot and partition numbering shown (looking at the service box from the frontal) for the IT.

Slot numbering	6	7	8	9	10	11		0	1	2	3	4	5
Partition numbering	2			3				0			1		
5V OCM	P2_OC5V			P3_OC5V				P0_OC5V			P1_OC5V		
2.5V OCM #1	P2_OC25V_#1			P3_OC25V_#1				P0_OC25V_#1			P1_OC25V_#1		
2.5V OCM #2	P2_OC25V_#2			P3_OC25V_#2				P0_OC25V_#2			P1_OC25V_#2		
Partition inhibit	P2_DBs_INH			P3_DBs_INH				P0_DBs_INH			P1_DBs_INH		
Bettle regulators Hb_INHibit signals. ON/OFF control of Hybrid Analog and Dig. Supply.	Hb_INH_6	Hb_INH_7	Hb_INH_8	Hb_INH_9	Hb_INH_10	Hb_INH_11	Ctrl Board slot in Svce Box (Frontal view)	Hb_INH_0	Hb_INH_1	Hb_INH_2	Hb_INH_3	Hb_INH_4	Hb_INH_5

Figure 4.10: Logical distribution for the OCM and inhibit signals according to the slot and partition numbering shown (looking at the service box from the frontal) for the TT.

As the CB must always be on, the *INH* lines of its internal regulators have been left open. The 2.5 V and 5 V regulators *OCM* lines are monitored but not the 3.3 V one since any over-current on it would mean a major malfunctioning in the CB (the SPECS slaves power would be missing) and hence no monitoring could be carried out. The three voltage values can be monitored via three different DCU channels (Figure 4.11).
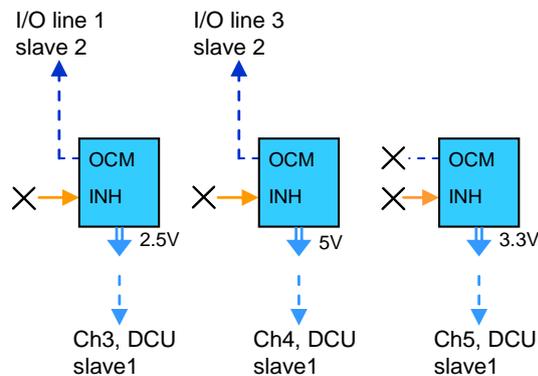
Figure 4.11: Control Board regulators control signals.

4.2.7   THE LV CABLES

The cables infrastructure needed for the LV ST basically consists of three different types of cables, as one can conclude from the scheme of Figure 4.1.

1. The 385 V DC cable (Primary Rectifier ↔ MARATON Power Box).
2. The control cable (RCM ↔ MARATON Power Box).
3. The LV power cables (MARATON Power Box ↔ Service Boxes).

For the cable types 1 and 2 there are some recommended specifications from the MARATON manufacturer [49].

The 385 V DC cable selection is based on maximum 10 V voltage drop at the cable, at maximum current of 10 A (total cable resistance ≤ 1 Ω). Both for IT and TT 4 mm^2 section shielded cables have been chosen. The estimated cables length is 64 m for the IT and 50 m for the TT.
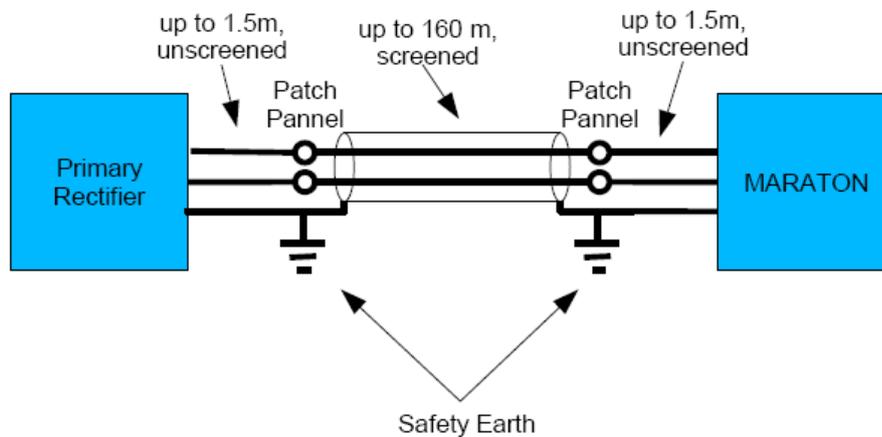
Figure 4.12: 385 V DC cable wiring.

The control cables must be a multi-conductor screened twisted pair cable of 18 pairs (36 conductors). Each RCM needs two cables for controlling one MARATON Power Box. The conductor cross-section has been chosen such that attenuation for the lengths of 64 m and 50 m does not affect the signal integrity.

The LV power cables are sub-detector specific, so it has been the ST responsibility to choose the most appropriate ones. The criterion used for selecting the type of cable needed has been based on allowing a maximum voltage drop in the cable of 1 V. It's been decided to use also shielded cables wired in the same way as the 385 V ones (Figure 4.12).

The voltage drop in the cables depends on the length as well as in the current flow and in the cable resistivity following the equation:

$$\Rightarrow \quad V_{drop} = \rho \times I \times L \big/ A$$

Where ρ is the cable resistivity given in Ohm·m/mm^2 . For this calculation we have decided to use a value of ρ = ~0,02 taken from a cable manufacturer for multi-wire flexible copper cables at 20 ºC. The current (I) and cable length (L) vary for the different service boxes in the IT detector as well as for the TT.

The 1-8 V @ 50 A channels will source more current than the other ones, and that's why the V_{drop} will be higher for those cables. This has leaded us to choose two different cable cross-sections, one for each channel type.

The two different cable cross-sections that have been selected are: 25 mm^2 cables for the 1-8 V @ 50 A MARATON channels and 16 mm^2 for the 2-15 V @ 20 A ones. The Table 4.10 and Table 4.11 show the detailed calculations for the voltage drop in the cables for all the different SBs of IT and TT. The contact resistance in the connectors has been considered to be negligible. All the values from the table have been successfully validated with real measurements in the experimental area.

		IT 1-8V@50A							IT 2-15V@20A						
		Svce box		Cable			contact losses		Svce box		Cable			contact losses	
		V [V]	I [A]	length [m]	xsec [mm²]	Vdrop [V]	R [Ω]	Vdrop [V]	V [V]	I [A]	length [m]	xsec [mm²]	Vdrop [V]	R [Ω]	Vdrop [V]
IT16	IT1 Out	4,50	24	18	25	0,35		0,00	7	8	18	16	0,18		0,00
	IT3 In	4,50	24	22	25	0,42		0,00	7	8	22	16	0,22		0,00
IT12	IT1 Out	4,50	18	20	25	0,29		0,00	7	6,5	20	16	0,16		0,00
	IT3 In	4,50	18	22	25	0,32		0,00	7	6,5	22	16	0,18		0,00

Table 4.10 : IT LV power cables voltage drop summary.

		TT 1-8V@50A							TT 1-15V@20A						
		Svce box		Cable			contact losses		Svce box		Cable			contact losses	
		V [V]	I [A]	length [m]	xsec [mm²]	Vdrop [V]	R [Ω]	Vdrop [V]	V [V]	I [A]	length [m]	xsec [mm²]	Vdrop [V]	R [Ω]	Vdrop [V]
TT	Access Bottom	4,50	24	17	25	0,33		0,00	7	8	17	16	0,17		0,00
	Access Top	4,50	24	19	25	0,36		0,00	7	8	19	16	0,19		0,00
TT	Cryo Bottom	4,50	24	32	25	0,61		0,00	7	8	32	16	0,32		0,00
	Cryo Top	4,50	24	34	25	0,65		0,00	7	8	34	16	0,34		0,00

Table 4.11: TT LV power cables voltage drop summary.

CHAPTER 5

THE ST HV SYSTEM INFRASTRUCTURE

ABSTRACT

This chapter describes the technical details of LHCb Silicon Tracker High Voltage (HV) infrastructure in a top to bottom approach.

"Algo malo debe tener el trabajo porque si no, los ricos lo habrían acaparado"

Mario Moreno - Cantinflas

5.1 INTRODUCTION

This chapter exhaustively describes the HV system of the IT and TT sub-detectors of the LHCb experiment. The information shown here is indispensable for anybody who wants to understand how the HV power distribution is like in the ST detectors and hence for any expert that wants to perform the development of the ST control system. Due to the nature of the information reported in this note, some hardware descriptions of the system and of parts of it will be necessary.

On one hand, a description of the solution adopted for the HV power distribution will be given. This comprises a description of the CAEN power supplies chosen and then a description of the power distribution performed in the different patch-panels.

A detailed explanation on how to perform the operation of the detector from the ECS HV point of view is given in the appendix F. In that part all the technical details necessary for setting up the HV system will be presented.

5.2 THE ST HV SYSTEM

The detection technology of the ST detector is based on single-sided silicon micro-strip detectors. The way they work can be easily explained as reverse biased diodes which generate current carriers when ionizing particles pass through the depletion zone.

The ST silicon sensors have been designed to withstand biasing voltages up to 500 V and have depletion voltages ranging from 80 V to 350 V depending on the radiation damage. The sensor bias leakage current is the other important parameter and it can range from several hundreds of nA when the sensor is new to ~1 mA after the ageing due to radiation damage.

Once these two parameters were known the decision of building the HV system based on commercial PS was taken. The CAEN modules A1511B were chosen: they can supply up to 500 V and 10 mA.

Since the current that these modules can source is by far enough for a single sensor, the decision of sharing one HV channel among several sensors was taken. This is implemented in several patch-panels present in the HV system, as it is shown in the schemes in Figure 5.1 and Figure 5.2.

The cables and connectors selection has been carefully taken in order to avoid any problem derived from any high voltage leak on them. Both cables and connectors need to cope with voltages up to 500 V in the long term and that's why material characterized to withstand this voltage is used.
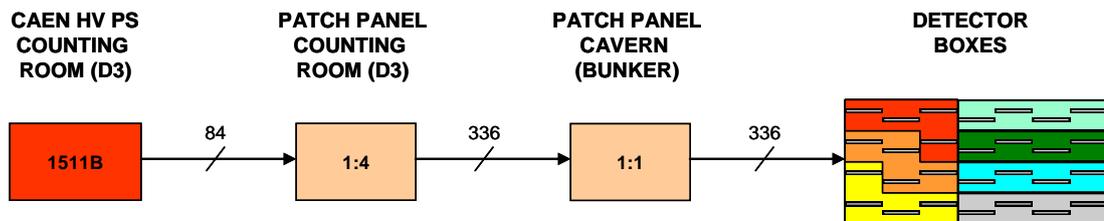
Figure 5.1: IT HV system simplified scheme.
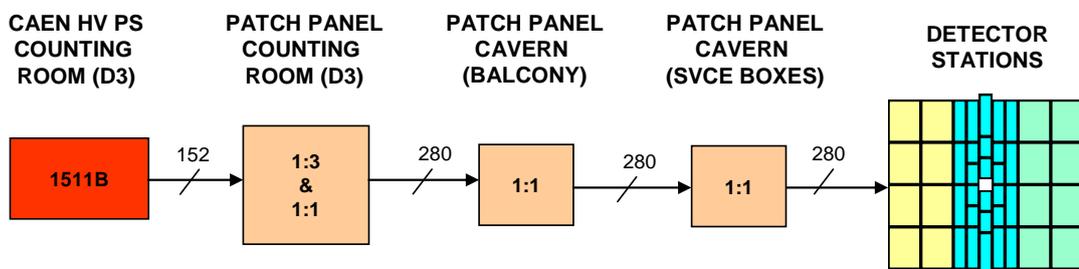
Figure 5.2: TT HV system simplified scheme.

5.2.1 THE HV CAEN SYSTEM

The CAEN system is based, as mentioned, on A1511B modules and SY1527LC "Low Cost" crates.

The SY1527 mainframe is housed in a 19"-wide, 8U-high euro-mechanics rack and hosts four main sections:

- The Board Section, with 16 slots to house boards, distributors and branch controllers.
- The Fan Tray Section, housing 6 fans disposed on two rows.
- The Power Supply Section, which consists of the primary power supply and up to 3 power supply units.
- The CPU and Front Panel Section which includes all interface facilities.

The model A1511B single width board comprises 12 HV floating channels (i.e. the channels do not share any ground reference). The output voltage can be programmed and monitored in the range 0-500 V with 100 mV resolution. The floating channels allow on-detector grounding, thus avoiding ground-loops which may increase noise level. The model A1511B offers 1 mA / 10 mA dual current Full Scale Range (selectable via internal jumpers).

Figure 5.3:
A1511B
front panel.
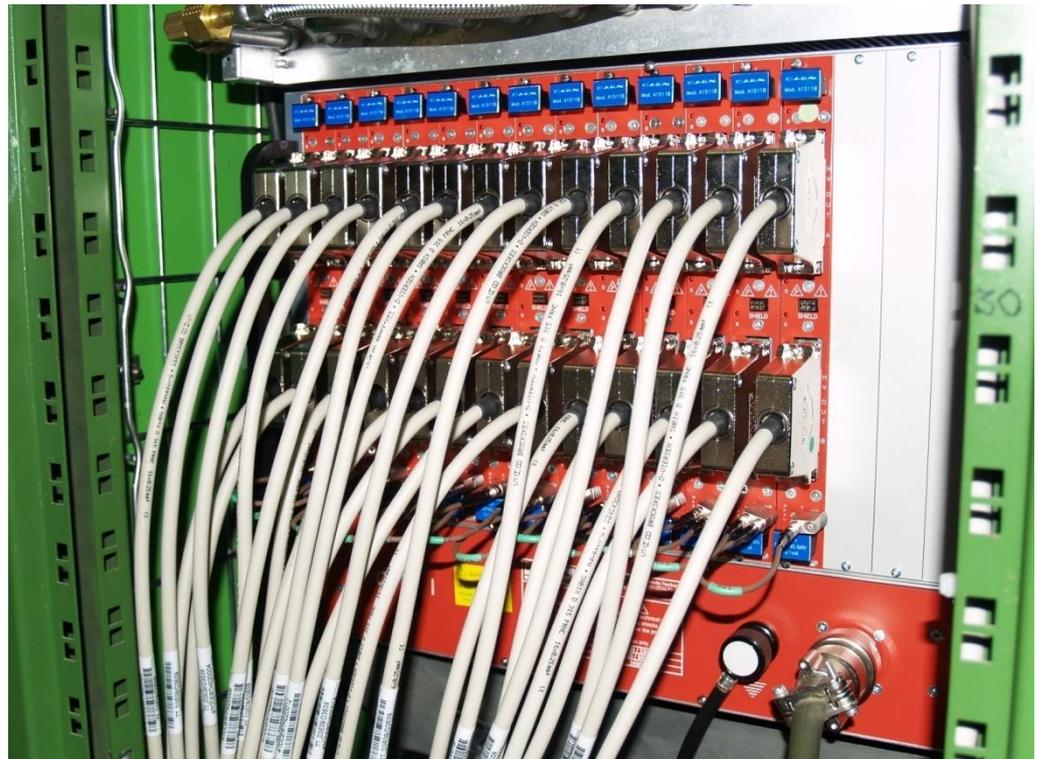
Figure 5.4: CAEN crate with A1511B boards plugged in and cabled.

5.3   THE HV DISTRIBUTION

The silicon sensors of the ST will be biased to a maximum of 500 V. As mentioned above, to save HV modules, neighbouring readout sectors are grouped together via a custom PP (Patch-Panel) placed in the counting room which allows for individual disconnection or readout sectors. This design still leaves the future option of independent HV channels for each readout sector. From these PPs in D3, the HV channels are routed to the cavern with long distance HV compliant cables. In the cavern there are other patch panels that re-route the lines to the detector. See the details in Figure 5.6 and Figure 5.7.

The CAEN power supply system consists of a CAEN crate housing A1511B boards providing 12 HV floating channels each. As the channels do not share any ground reference each consists of 2 lines (HV+ and HV-), which need to be driven up to the detector boxes, where the common ground point of the detector is located.

Figure 5.5: IT CAEN and HV patch panel crate.

The IT detector needs 84 CAEN HV channels which are split into 4 via patch panel in the counting room to accommodate the 336 HV lines needed for all the IT hybrids. Eight A1511B modules provide 96 channels, out of which 84 connect to the "1:4 patch-panel" so we get the 336 lines. Both the CAEN PS and the PP are located in the same rack in the counting room as one can see in the figure 1.5. From this PP twelve HV 56-wire screened cables, one per detector box, connect to the patch-panels in the cavern. The twelve multi-wire cables split in two groups of six for each side of the detector. The cavern patch-panels consist of one to one connections. After this second PP the cable is fanned out into four bundles of 7 channels, each corresponding to a layer of 7 ladders, and connected to the exterior through a printed circuit board feedthrough.

The HV distribution for the TT follows a scheme similar to the IT, differing in the channel splitting and the number of patch-panels. As the TT sensors leakage current will be considerably higher than in the IT due to the sensor structure, the sensor's grouping (up to 4 sensors) and mostly due to the radiation damage, it was decide to do a different splitting of the HV channels. A 1:3 ratio is used for the external TT sectors and a 1:1 in the internal ones. For this reason, 152 HV channels are needed, from which just 64 are being 1:3 split off. From the TT PP in the counting room 8 cables with 24 channels and 4 with 22 go out to the patch panel placed on balcony. This PP just performs a one to one cable connection and from there the cables are routed to the patch-panels in TT service boxes location. These last patch panels on the service boxes just perform a 1:1 connection and re-arrange the lines to fit in the cables as depicted in the scheme.
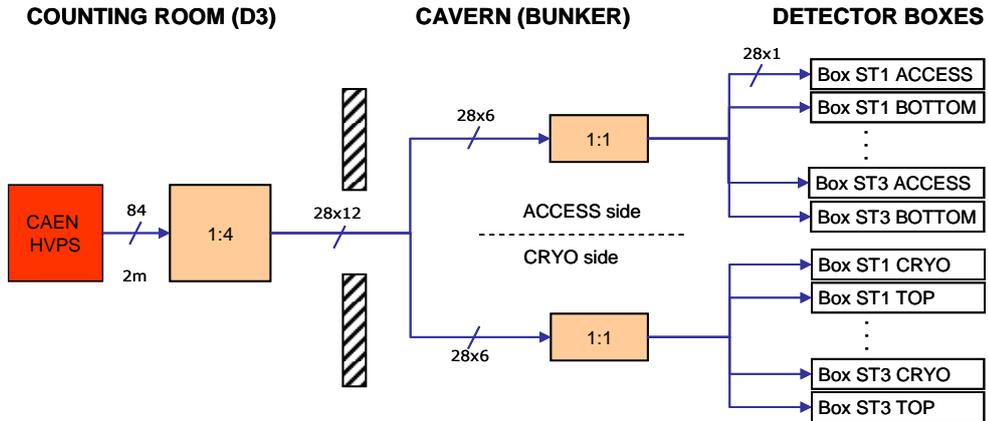
Figure 5.6: IT HV infrastructure general view.
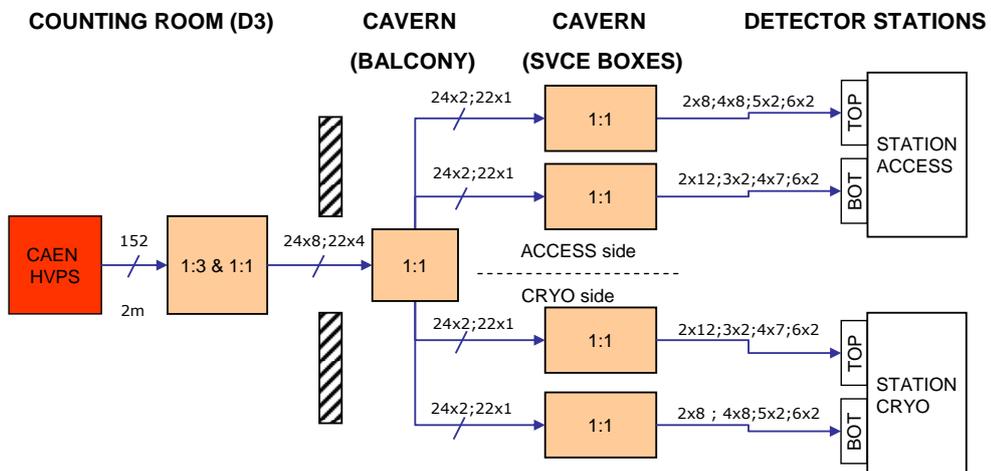
Figure 5.7: TT HV infrastructure general view.

5.3.1   THE PATCH-PANELS

There are three types of HV patch panels in the ST system:

- The PCB split off patch panels in the counting room.
- The one to one 56-wire cable PP in bunker (IT) and balcony (TT).
- The patch-panel in the TT service boxes that rearranges the HV cabling.

The PCB patch-panels in the counting room perform the channel splitting and they implement jumpers that permit disconnection of individual readout sectors. The printed circuit board is standard technology FR4 construction with high voltage (500 V) design rules applied to the trace and part spacing. High voltage traces run on internal layers whenever possible to minimize electrical leakage due to high humidity or condensation. The board is composed four layers evenly distributed in thickness giving a total profile of 3.4 mm. The enable lines available in each of the 2 output connectors of the HV modules are terminated in this patch-panel with 0 Ω resistors.

Figure 5.8: IT patch-panel in Counting room.

Figure 5.9: : TT patch-panel in Counting room.

The PP silkscreen shows for every pair of jumpers the associated readout module. To disconnect one of the modules the channel mapping must be known beforehand and then both jumpers removed.

Both patch-panels are secured against unintended manipulation with transparent 'makrolon' covers, which are key-locked (Figure 5.8 and Figure 5.9).

5.3.2   THE HV INTERLOCK

The HV patch-panel door and the HV modules enable lines are directly connected to the DSS (Detector Safety System) for safety reasons. In case anybody opens the PP door a DSS alert will be triggered which will afterwards shut-down the HV modules system.

The interlock box shown in Figure 5.11 connects to DSS by a 4-wire shielded cable. Two of the lines are an input to the DSS and signal the status of the HV PP door, and the other two are an output and will open/close the circuit that disables/enables the HV modules operation. This interlock box is attached to the rear side of the patch panel and the electrical wiring diagram is shown in Figure 5.10. All the connectors shown in the scheme are polarized and the cables shielded and connected to ground, as shown in the figure.

5.3.3   CHANNEL MAPPING

As explained earlier, in order to save costs, several readout modules both in the IT and TT are grouped together and biased by the same HV channel. This implies that 84 HV channel will bias the 336 IT readout modules and 152 will bias the 280 modules of the TT.

The grouping of the IT modules is the same for the HV and the DAQ readout according to the scheme in the Figure 5.12. Each group of 4 ladders, determined by the background color, is independent from the others, so the fact that in the colors are repeated among boxes does not have any implication.
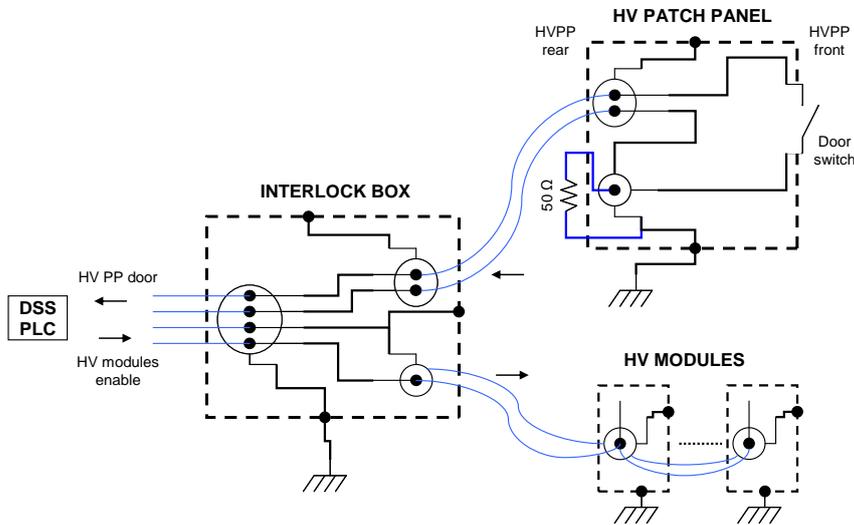
Figure 5.10: Interlock box, HV patch panel and HV modules electrical wiring to the DSS.   Figure 5.11: Interlock box.
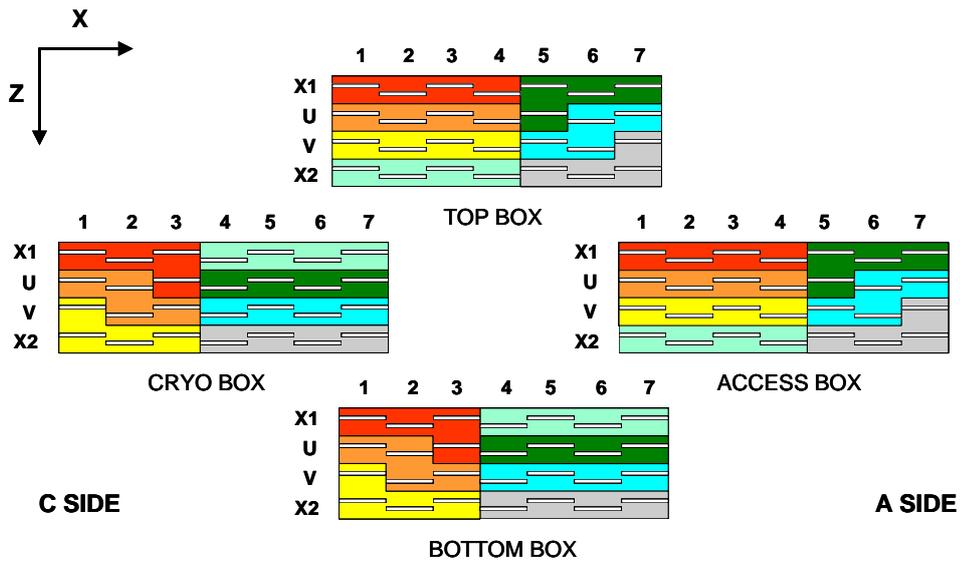
Figure 5.12: IT ladder grouping.

The Figure 5.13 shows how the CAEN crate is filled in: just the even slots are plugged with boards. In three different colors is shown the HV channel distribution according to the three different stations, namely: blue for the station 1, yellow for the station 2, light green for station 3, and the white background means that the channel is not used. According to the channel naming convention used in the figure, every channel name indicates the slot where the board is plugged in (from 0 to 15), the connector assignment (top or bottom) and the channel number in the board (0 to 11). Cross-referencing the Figure 5.13 and the Figure 5.14 one can univocally match the CAEN HV channel to the readout sector.

The TT HV partitioning follows a mixed grouping: while the groups in A (Access) and C (Cryo) regions are made of three readout modules, the B (Beam) region consists of single module groups. This grouping also matches the DAQ readout. The Figure 5.15 depicts the DAQ readout grouping and the modules numbering. The yellow and light green colors represent the C and A regions respectively, where the groups are outlined with at thicker line, and the blue color represents the B region.

The mapping between HV channel and biased readout module can be deferred from Figure 5.16 and Figure 5.17. These two figures follow the same coloring scheme as the previous one, so one can easily find that the number of channels dedicated to the B region is more than for the others due to the one to one assignment.
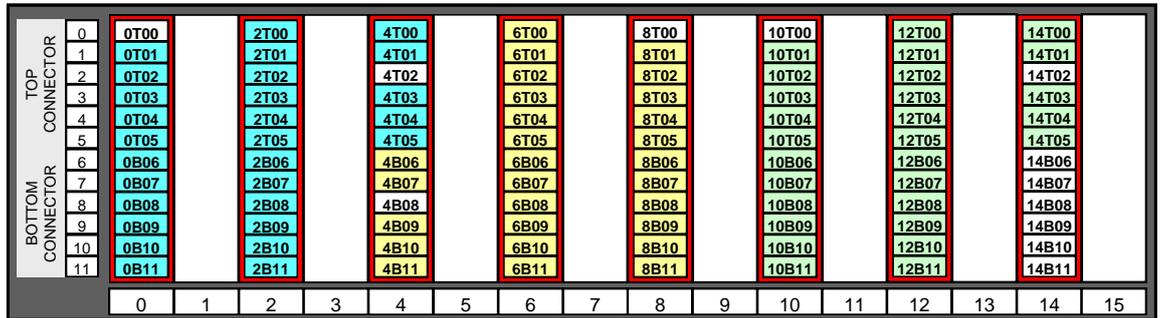
Figure 5.13: IT CAEN crate HV channels filling.
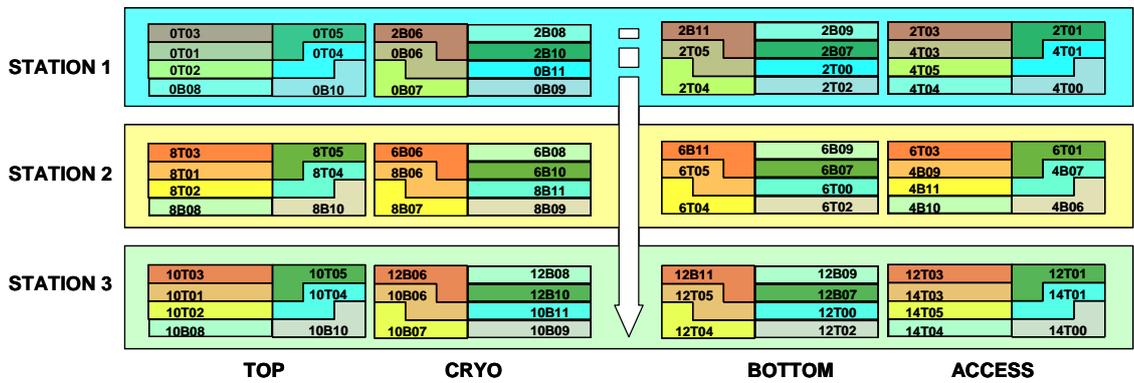
Figure 5.14: IT detector HV channels mapping.
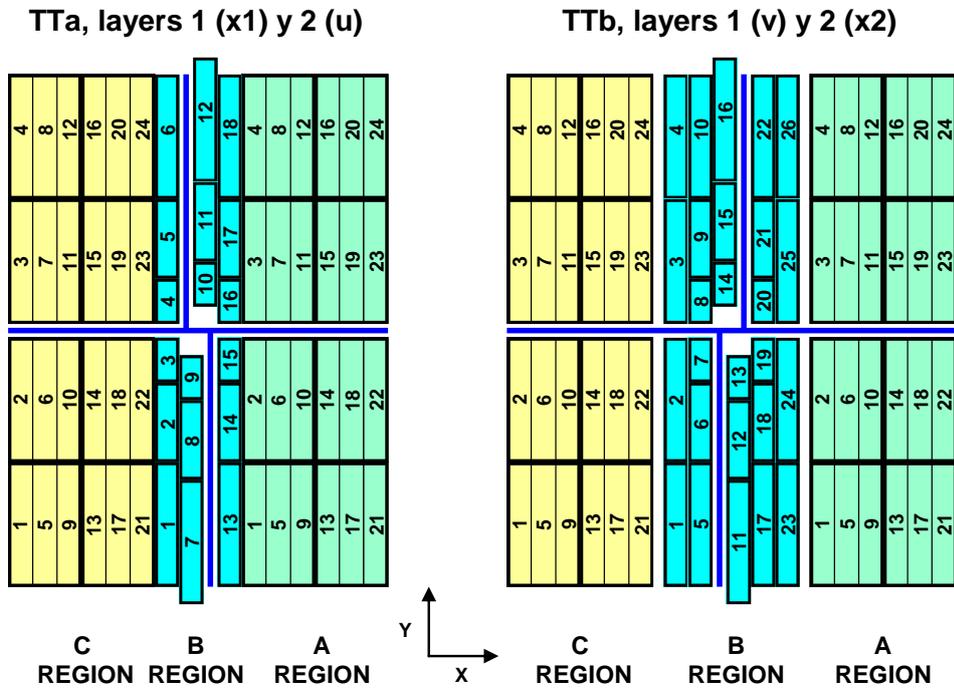
Figure 5.15: TT readout modules grouping.

Figure 5.16: TT CAEN crate HV channels filling.

TTa Layer X1 **TTa Layer U** **TTb Layer V** **TTb Layer X2**
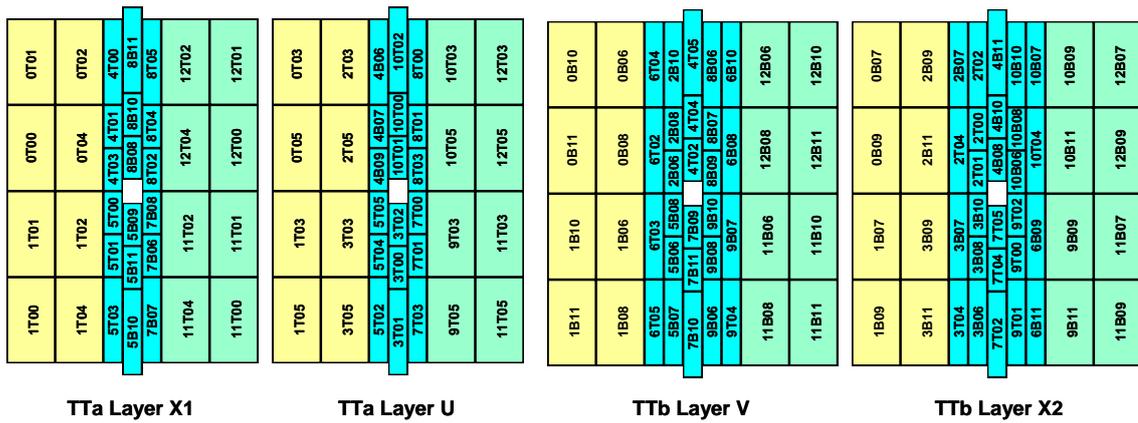
Figure 5.17: TT detector HV channels mapping.

CHAPTER 6

THE ST TEMPERATURE AND HUMIDITY INFRASTRUCTURE

ABSTRACT

This chapter describes the LHCb Silicon Tracker temperature and humidity monitoring infrastructure from a technical point of view.

"No es rico el que más tiene, sino el que menos necesita"

San Agustín

6.1 INTRODUCTION

This chapter describes the means implemented in the ST to perform the detector temperature and humidity monitoring. Hardware descriptions of the system elements will be given whenever needed.

Due to the irradiation damage in the silicon, the ST sensors have to be operated in a cold and dry environment to prevent the generation of too much shot noise caused by increased leakage currents. Since temperature has also a very strong effect on the reverse annealing of the full depletion voltage [50], operation at lower temperatures makes the detector lifetime longer. In addition, the Beetle FE chip dissipates ~0,625 W and if this heat is not removed the chip could get damaged. Keeping the chip temperature low also improves its S/N performance.

A cooling system is then mandatory to remove the dissipated power of the FE electronics located on the silicon ladder end in order to keep the detector box temperature below 5 °C [51]. Temperature monitoring of the silicon sensors environment and of the readout hybrid inside the box is needed to verify that the cooling is properly working. Thermal studies about the IT and TT detector boxes are reported in [52] [53].

The temperature monitoring inside the detector boxes is provided by the thermal sensors located on each hybrid and on the detector box enclosure. IT and TT detector boxes are populated with platinum PT1000 temperature sensor over the whole volume. Platinum sensors are well characterized to be radiation resistant and hence can be placed in the detector.

In addition to the temperature monitoring of the detector boxes also the temperature of the SB backplane and of the CB is supervised. A mixed water system is used to cool down the cooling blocks in the CB and backplane and PT1000 sensors are used for the monitoring.

The relative humidity inside the box must also be keep low and a dry gas (N_2) circuit is used to flush detector boxes in order to prevent condensation on the cold surfaces inside the box. The humidity measurements are provided by the HMX2000 sensor from 'Hygrometrix' which has been characterized by the CMS tracker group for radiation hardness.

The acquisition electronics for the sensors is placed both in the Control Board and in the Digitizer Boards and a clear explanation about the readout is given as well.

6.2 THE ST TEMPERATURE MONITORING SYSTEM

The purpose of the cooling system of the ST detector is to remove the dissipated heat from the front end hybrids and to maintain the detector modules at a constant temperature of about 5 °C during operation.

PT1000 temperature sensors have been installed in the detector boxes to monitor the box internal temperature. Each SB backplane houses two PT1000 sensors as well. In each CB was foreseen a place for a PT1000 for monitoring of the regulators temperatures just in the rear side of the PCB, and they were finally assembled in order to assess that they do not overheat. All these sensor signals are sampled and read out to the ECS system by the CB. Finally, each IT and TT hybrid houses a PT1000 that is read out by the corresponding DB and then transmitted to the ECS by the CB.

Many reasons led to chose this type of sensor: their relatively low cost and fast response, linearity, stability, accuracy and repeatability. But a crucial point was the radiation hardness properties that platinum temperature sensors show. Apart from this, the PT1000 advantage with regard typical PT100 is that they have higher nominal resistance (1000 Ω compared to 100 Ω) and as a consequence the bias current can be kept lower. Therefore the voltage drop in the cables is almost negligible hence there is a lower dependence with wire resistivity when using a 2-wire measurement. This effect could be avoided using a typical 4-wire, but 2-wire design is preferred because of the limited number of available pins on the readout hybrid interface connector. The PT1000 simplified formula is given by:

$\Rightarrow$ $R_{PT1000} = R_0 + T[°C] \times 3.85\ [\Omega]$, with $R_0 = 1000\ \Omega$

The temperature readings are sent out to the ECS system via SPECS provided by the CB. The PT1000 biasing, signal conditioning and acquisition electronics is placed in the CB for the box and service box temperatures and in the DBs for the hybrids.

In order to get a good accuracy in the box environmental temperature reading a calibration of all the CB readout channels was performed. For the hybrids temperature just a rough calibration was performed since the temperature is just needed in order to detect a major cooling problem in the hybrid.

6.2.1 THE READOUT CIRCUIT

The readout circuit (see Figure 6.1) consists of wheatstone bridge for the PT1000 biasing, an instrumentation amplifier with gain G_A, a voltage scaling of factor G_D and finally the ADC input of the DCU.

The DCU chip is multi-channel analogue to digital converter of single slope type that implements an I^2C interface. In this type of circuits a constant current charges a capacitor. The resulting voltage "ramp" is compared with the input voltage. A counter at the LHC clock frequency (~40 MHz) is started at the beginning of the conversion cycle and measures the time the capacitor voltage takes to reach the input voltage value. The final count is proportional to the input voltage and represents the conversion result. An external 120K Ω resistor is used to convert the internal DCU voltage reference into a 10 μA reference current that generates all the currents used in the A/D.

The ADC gain depends both in the clock and the external resistor and for this reason the calibration of all the DCU channels of the CB to read the temperatures and humidity was done obtaining a calibration curve as explained in 6.2.2. Although the DCU gain provided in the datasheet approximates the real gain with a confidence of ~5%, this is not true for the DCUs in the SPECS slaves which due to an implementation feature show a deviation bigger than 10%.
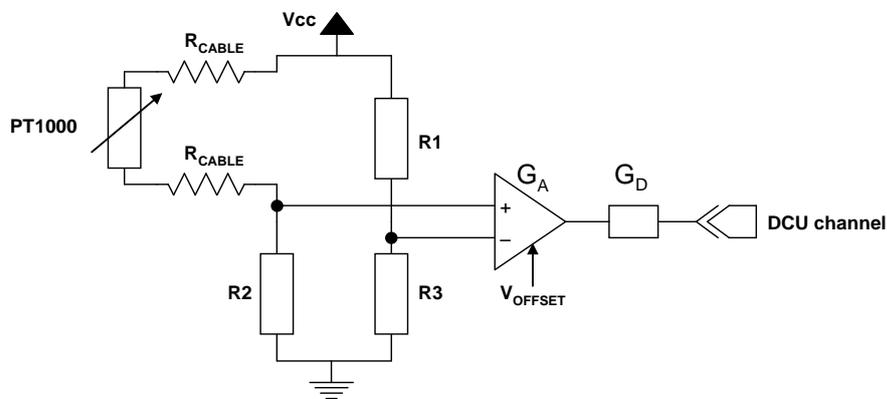
Figure 6.1: Scheme of the PT1000 wheatstone bridge biasing circuit and signal conditioning.

The wheatstone bridge uses high precision and low tolerance resistors (0.1% and 10ppm/°C) with the nominal values of $R2 = R3 = 8k25\ \Omega$ and $R1 = 909\ \Omega$. V_{cc} and V_{OFFSET} are 2.5 V. G_A is set to 11 and G_D is equal to 1 (no scaling) in the Digitizer Boards circuit implementation (hybrids temperature) and 1/3 in the Control Boards (box and service box readings). This latter attenuation in the CB circuit has a twofold purpose: ensures that input to the DCU will always be lower than ~1.5 V so the DCU can always be operated in LIR[1] mode and also guaranties that in case of a failure or a voltage drift from the amplifying stage the voltage and the DCU inputs will never reach 2.5 V, which could damage the DCU.

The R_{CABLE} resistance in the picture represents the cables and contacts resistance. This is an unkown parameter that has a non-negligible impact on the measurement since the cables length in some cases reaches 10 m. This effect needs to be compensated basically in the TT detector where the cable resistances become significant.

6.2.2 THE READOUT CALIBRATION

With the aim of having a precise temperature measurement of the box environment, all the CB temperature readout channels were calibrated. A straight line calibration curve that represents ADC readings versus temperature was generated for every channel that is readout by the CB.

All sensors are of DIN 60751 tolerance class B with a permissible deviation of $\pm\ (0.3 + 0.005 \cdot |T|)$, where T is the temperature in °C. PT1000 sensors offer great stability, accuracy and repeatability and for that reason calibrating the PT1000 is not required.

To obtain the calibration curves of the CB temperature channels, a custom made setup was built. It consists of: a small calibration board (Figure 6.2) with some reference resistances; a 5 m cable similar to those used in

---

1 LIR stands for Low Input Range and means that the DCU performs the measurement with reference to ground, while in the High Input Range mode (HIR) the power supply (2.5 V) is taken as the signal reference.

the experiment for the detector box temperatures readout, which connects the CB under calibration with the calibration board; a short cable (30 cm) which also connects the CB to the calibration board to calibrate the backplane temperatures readout channels; and finally, a Readout Supervisor that provides a LHC ~40 MHz like clock. This latter point is quite important as the clock frequency influences the DCU reading.

In the small calibration board six sets of five high precision and low tolerance resistors (10 ppm/°C and 0.1%) were assembled. Four of these sets are used to calibrate the channels corresponding to the PT1000 sensors in the box and the other two for the sensors in the backplane. The chosen resistor values $R1 = 976\,\Omega$, $R2 = 1000\,\Omega$, $R3 = 1020\,\Omega$, $R4 = 1050\,\Omega$ and $R5 = 1100\,\Omega$ correspond in a PT1000 to temperatures of -6.23 ºC, 0 ºC, 5.19 ºC, 12.98 ºC and 25.97 ºC. A jumper permits selecting the resistor that is going to be used for the calibration run.
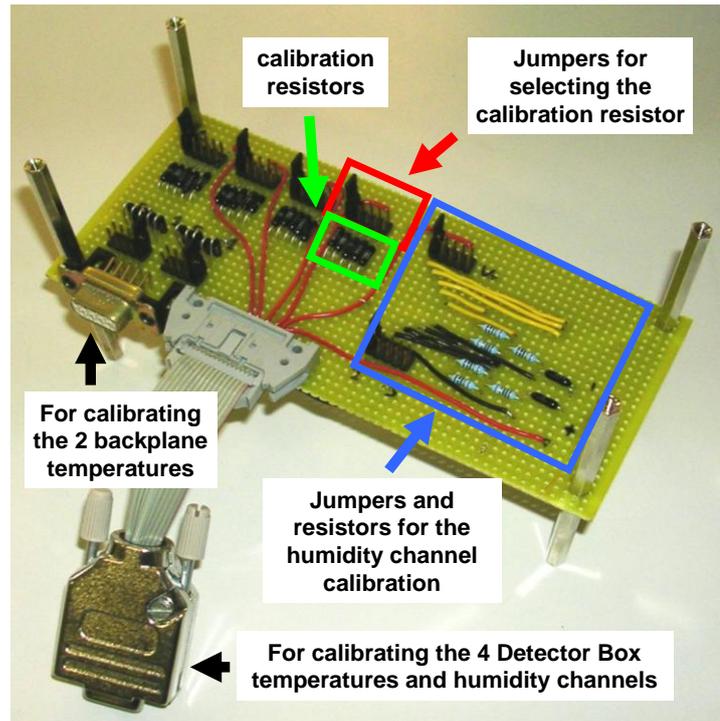
Figure 6.2: Calibration board.

The calibration procedure in each CB consisted in reading all the channels a hundred times for each of the five resistance values, and finally calculating the average and the *rms*. The Figure 6.3 shows a sample of the data obtained for the box temperature in the jumper position correspoding to -6.23 ºC and also a calibration curve after doing the five readings. As expected all data samples for all the calibration runs fitted perfectly to straight line. The calibration functions can be found in appendix G.

The *rms* error in the temperature reading inherent to the system (sensor biasing, signal conditioning and sampling) can be roughly estimated from the ADC *rms*. As shown in the table the ADC *rms* estimation is ~1.5 ADC counts (which is repeatable for all CBs). In the temperature calibration range, from $T_1 = -6.23$ ºC to $T_2 = 25.97$ ºC the ADC values range from ~1400 ADC to ~1200 ADC, which involves a ratio of ~0.161 ºC/ADC and thus the temperature *rms* is ~0.241 ºC.

This result confirms that there is no need to calibrate the PT1000 as they are manufactured with a $\pm(0.3+0.005\cdot |T|)$ ºC deviation, in the same range as the readout *rms* error.

The temperature calibration results have also been used to get an estimation of the ADC gain [V/ADC], which is valuable for the DCU channels that are not used for temperature monitoring but for other purposes. No need to say that of course the uncertainty of this gain estimation is affected by many known and unknown parameters. The experiences in the laboratory and later on in the real detector have shown that the estimation is completely compatible with measurements performed with a handheld multimeter and therefore good enough to be used in the non-calibrated DCU channels (basically used for voltage readings).

DCU CHANNEL	AVERAGE (ADC)	RMS (ADC)	AVERAGE (mV)	RMS (mV)
CH1	1401,99	1,4178	0,7736	0,0008
CH2	1400,4301	1,4984	0,7727	0,0009
CH3	1404,22	1,4737	0,7748	0,0009
CH4	1400,3001	1,4867	0,7726	0,0009
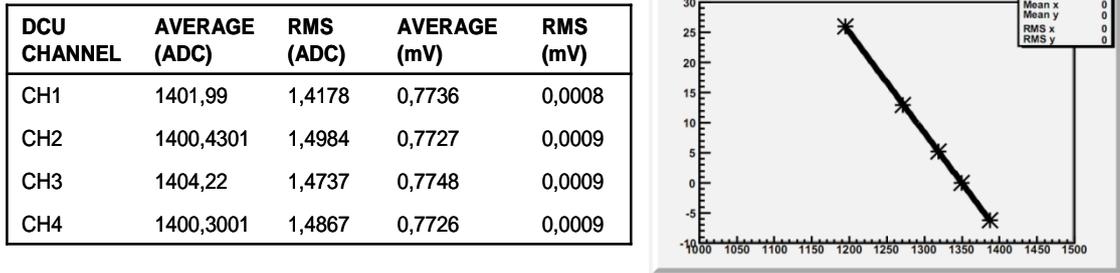
Figure 6.3: Calibration data sample of the temperature channels on the left and a calibration curve in the right.

6.2.3   IT DETECTOR BOX TEMPERATURES

Four sensors are placed in each IT box to perform the temperature monitoring (Figure 6.4). A 15-pin HD-subD connector in the detector box feed-through PCB provides the interface for the PT1000 sensors. The pin-out matching between detector box and CB can be found in the same figure.

The temperatures of each box are read out by only one Control Board as highlighted in Figure 6.5 in grey and black colors. The temperatures of boxes cryo and bottom are read by the first service box and the access and top by the third, numbering outwards the beam pipe. The figure also shows the DCU channel assignment.

The G_A of the amplifying circuit is set to 11 and the G_D to 1/3 by a voltage divider for these sensors.

The cables' length varies from 2.8 m the shortest (bottom box) to 5.5 m the longest (top box), being the cable resistance 0.156 Ω/m according to the manufacturer specifications. As the calibration was performed with a 5 m cable, the effect of cable resistance in the worst case scenario is for 2.8 m cable, which means that path difference is 2×2.2 m that results in 0.686 Ω. As a result a change in the PT1000 resistance of 3.85 Ω means a 1 ℃ difference, so 0.842 Ω implies a deviation of ~0.18 ℃, which can be considered as not appreciable in terms of temperature shift.

The temperature value is calculated as follows:

$\Rightarrow$ $T = ChDCU[adc] \times a + b$ [℃],

where *ChDCU* is the ADC reading of the DCU channel, and *a* and *b* the parameters of the calibration curve associated to the specified channel. The parameter *a* corresponds to the curve slope and *b* to the offset.
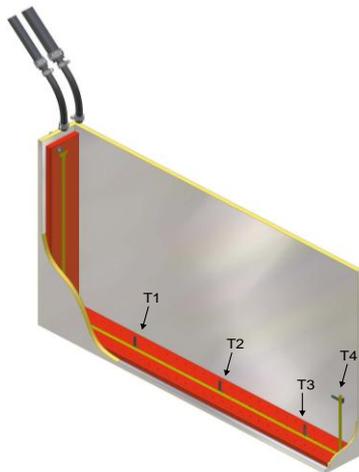
CB SubD15 pin	Device pin	Det. Box SubD15 pin
[14]	PT1000_1+	[6]
[15]	PT1000_1–	[1]
[7]	PT1000_2+	[7]
[1]	PT1000_2–	[2]
[13]	PT1000_3+	[8]
[12]	PT1000_3–	[3]
[11]	PT1000_4+	[9]
[6]	PT1000_4–	[4]
[4]	HMX2000_VEXC+	[14]
[3]	HMX2000_VEXC –	[10]
[5]	HMX2000_SIG+	[5]
[10]	HMX2000_SIG –	[15]

Figure 6.4: PT1000 in the IT detector box (left) and connector pin-out (right).

Box Top → CB in Service Box 3, Cryo side
Box Cryo → CB in Service Box 1, Cryo side
Box Bottom → CB in Service Box 1 Access side
Box Access → CB in Service Box 3, Access side
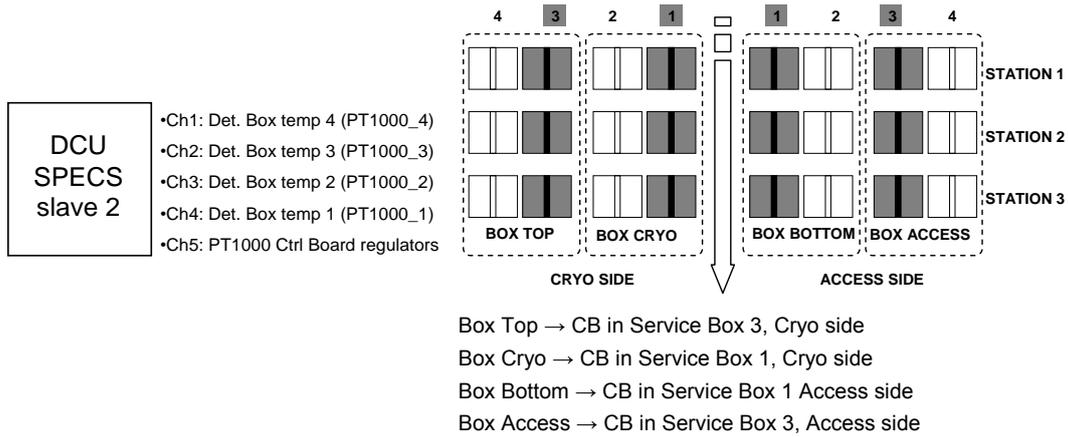
Figure 6.5: DCU temperature channel assignment (left) and correspondence between Detector Box and Control Board used form ambient monitoring.

6.2.4   TT DETECTOR BOX TEMPERATURES

The TT temperatures distribution is fairly more complex than for the IT. The document [52] describes the thermal studies performed and the sensors distribution. The TT is populated with sensors distributed through the cooling plates, side walls, near the beam pipe and in the cooling supply lines. The sensors in the supply lines are integrated in a tube sitting directly in the fluid. For an exact overview of the experimental setup and the positioning of the temperature sensors see in Figure 6.6 and Figure 6.7.

A passive patch-panel placed next to the service boxes redistributes the signals lines coming from the sensors to the control boards. The readout mapping is shown in Figure 6.8.

As a remark, the temperature sensors *C1* to *C8* are all placed next to each of the HMX2000 humidity sensors. This temperature measurement will be used to compensate the humidity sensor temperature dependence.
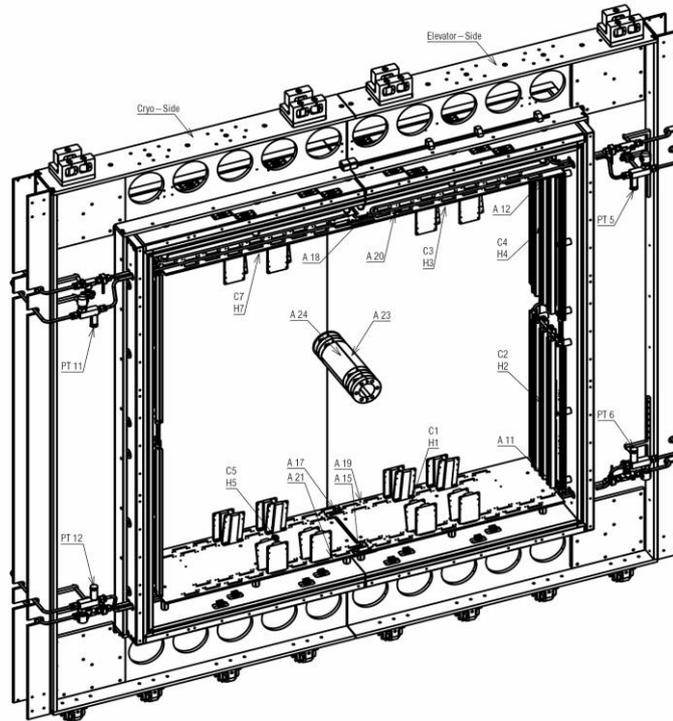
Figure 6.6: Temperature and humidity sensor locations in the TT detector magnet view. Picture taken from [52].
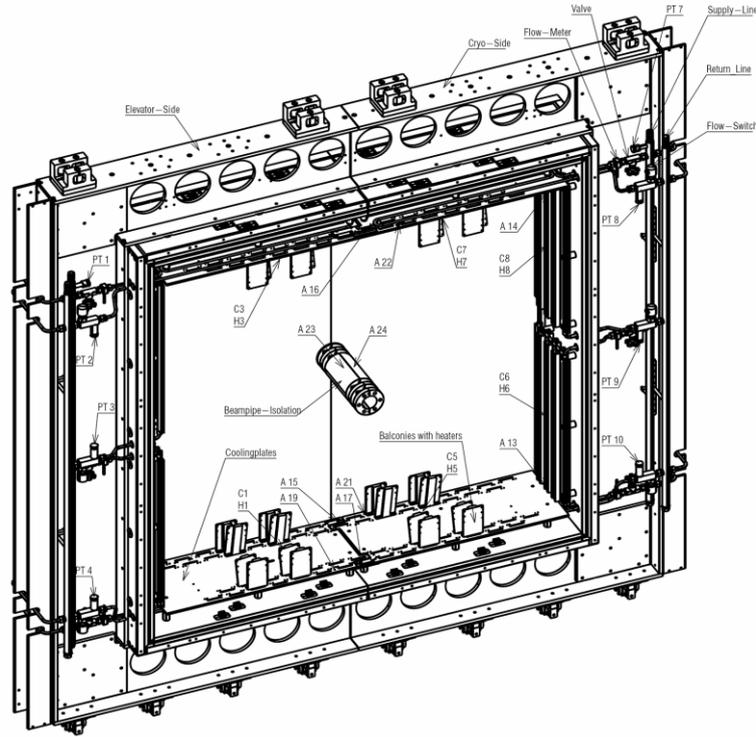
Figure 6.7: Temperature and humidity sensor locations in the TT detector magnet view. Picture taken from [52].

The length of the readout cables is around 10 m, which means that the deviation with respect to the calibration cable length is 2x5 m and hence ~1.56 Ω. The most significant contribution comes from the detector kapton patch-cables which connect the detector patch panel to the hybrids. The resistance was measured to be 2x3.5 Ω.

⇨ The temperature value is calculated in the same way as for the IT:

$$\Rightarrow \quad T = ChDCU[adc] \times a + b + C_F \ [\text{°C}], \ C_F = \frac{-2 \times 3.5 + 1.56}{3.85} = -2{,}22\,°C$$

where *ChDCU* is the ADC reading, and *a* and *b* the parameters of the calibration curve associated to the specified channel, being *a* the curve slope and *b* the offset. C_F is the correction factor that compensates the voltage drop in the measurement due to the cable resistance.

		Temperature senors mapping				
		DCU slave 2				DCU CB
		Ch1	Ch2	Ch3	Ch4	Ch3
Service boxes	Access Top 6	C3	A23	–	–	H3
	Access Top 5	C4	A20	A16	A12	H4
	Access Bottom 6	C1	–	–	–	H1
	Access Bottom 5	C2	A19	A15	A11	H2
	Cryo Top 6	C7	A24	–	–	H7
	Cryo Top 5	C8	A22	A18	A14	H8
	Cryo Bottom 6	C5	–	–	–	H5
	Cryo Bottom 5	C6	A21	A17	A13	H6

Figure 6.8: Detector box to Control Board temperature readout mapping for the TT.

6.2.5 THE BACKPLANE TEMPERATURES

Two SMD PT1000 sensors are placed in the backplane (see Figure 6.10) to monitor the backplane temperature. The backplane regulators slug is contact with a copper water cooled block. These sensors in the backplane PCB will warn us about any problem arising from a lack of water cooling or a bad thermal contact between regulators and cooling block.

The G_A of the amplifying circuit is set to 11 and the G_D to 1/3 by a voltage divider.

The temperature is calculated in the same way as in 6.2.3.

6.2.6   THE CONTROL BOARD TEMPERATURE

In the rear side of the CB a place to assemble a temperature sensor was foreseen. The location of this sensor is in the opposite side of the CB regulators, the hottest spot in the PCB. This sensor has finally been assembled in the CB as this has been shown to be one of the cooling critical points in the system.

As this sensor was foreseen as a backup option, the biasing and readout circuit was kept very simple (see Figure 6.11). As a consequence, the measurement is not very accurate (in the order of ± 4 ºC) but it is fine enough since it is just used for triggering an alert signal in case of overheating due to a cooling failure.

This circuit is a simple voltage divider, so the PT1000 resistance can be obtained as follows:

$$\Rightarrow \quad R_{PT1000} = \frac{R_1 \times V_{PT1000}}{V_{CC}\text{-}V_{PT1000}} [\Omega],$$

with $R1 = 1000\ \Omega$, $V_{cc} = 2.5$ V and V_{PT1000} is the voltage measured by the DCU, which can inferred from the ADC reading times the DCU gain estimation (indirectly calculated from the DCU temperature channels calibration). The temperature is calculated as follows:

$$\Rightarrow \quad T_{PT1000} = \frac{R_{PT1000}\text{-}R_0}{3.85} \ [\text{ºC}], \text{ with } R_0 = 1000\ \Omega.$$
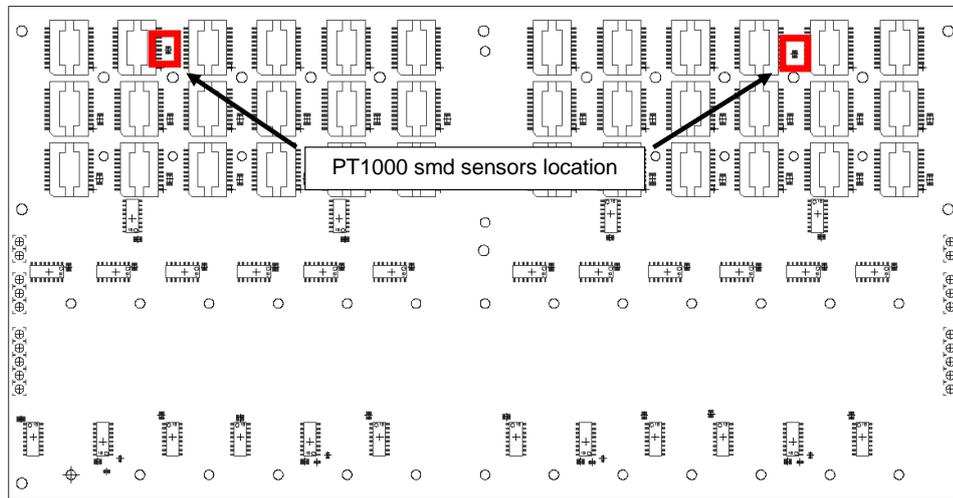
Figure 6.9: Backplane temperature sensors location.
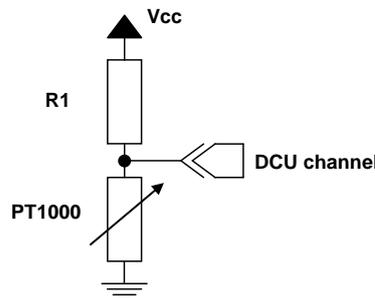
Figure 6.10: Control Board temperature monitoring circuit.

6.2.7   THE HYBRID TEMPERATURES

Each readout hybrid houses a PT1000 sensor. The purpose of this sensor is to check that the Beetles on the hybrid are being properly cooled down. This sensor is readout by the channel 1 of the DCU implemented in the corresponding Digitizer Board.

The G_A of the amplifing circuit is set to 11 and the G_D to 1, so there is not voltage scaling in the amplifier output and thus the DCU will have to work in HIR mode. As a consequence, this measurement can shift a bit with time since working in HIR mode involves using the DCU supply voltage as reference for the sampling

electronics. The supply voltage is provided by a voltage regulator, so at first glance the reference is not as well defined as it would be GND, although it is well set around 2.5 V. Additionally, the voltage regulators output tend to drift a bit with the working temperature, which would imply that the temperature measurement could slightly change with the warming up of the devices.

The temperature can be obtained using the nominal ADC gain of the DCU of 2.22 ADC/mV. The formulas for calculating the input voltage at the DCU channel, the PT1000 resistance and finally the temperature solving the circuit in Figure 6.1 and using the mentioned gains are the following:

$$\Rightarrow \quad V_{IN} = 2.5 - \frac{ChDCU[adc] \times 0.001}{2.22} \ [V],$$

$$\Rightarrow \quad R_{PT1000} = \frac{11 \times 2.5 \times 8250}{V_{IN} - 2.5 + \left(\frac{2.5 \times 8250}{9159} \right)} - 8250 \ [\Omega], \text{ and finally}$$

$$\Rightarrow \quad T_{PT1000} = \frac{R_{PT1000} - R0}{3.85} + C_F \ [^\circ C],$$

with $R_0 = 1000 \ \Omega$ and the correction factor $C_F = 0$ for the IT hybrid temperatures measurement and $C_F = -2.22 \ ^\circ C$ for the TT (as explained in section 6.2.4).

After long term tests with the detector running in normal working conditions the temperature measurements performed showed a rough variation of $\pm 3 \ ^\circ C$ which is considered to be good enough for their purpose, which is warning in case of a serious cooling problem.
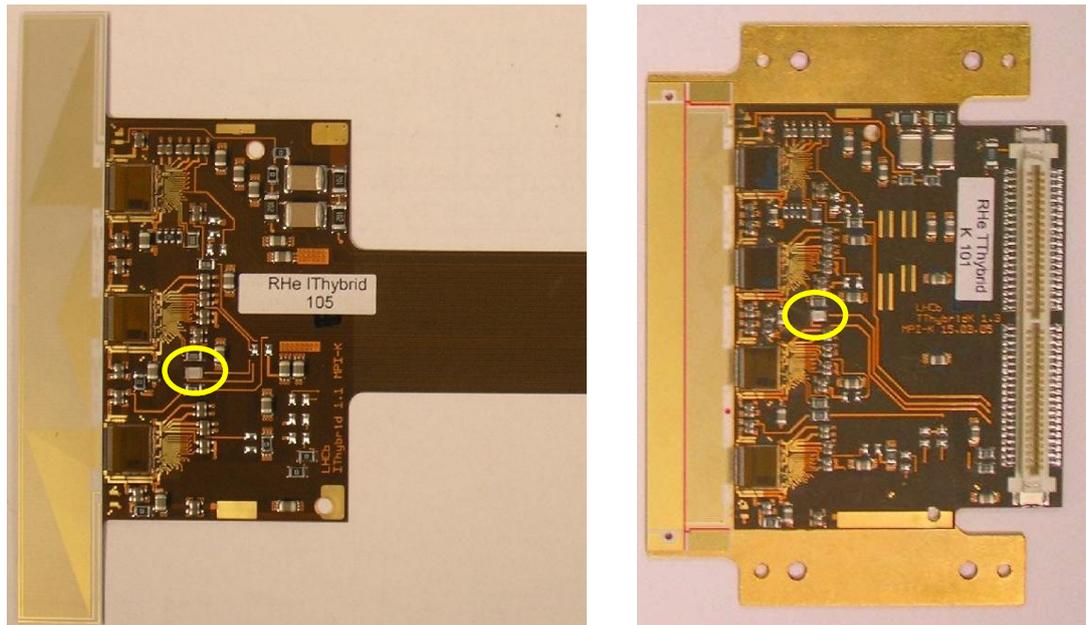
Figure 6.11: PT1000 sensor placement in sample IT (left) and TT (right) hybrids.

6.3   THE ST HUMIDITY MONITORING SYSTEM

Apart from the temperature monitoring, one major concern for the ST is the humidity inside the detector boxes. The detector will have to be cooled down to temperatures below 0 ºC. The detector boxes enclosure provide not just thermal isolation but also against the humidity in the cavern. However, this cover cannot guarantee total isolation of the ST from the atmosphere in the experimental cavern. Water vapor will be present inside the detector boxes being this one of the most critical risks that will affect the ST.

During cooling down water vapor can saturate and revert to the liquid state or even frost, which can potentially destroy either the sensors or the readout electronics. In order to avoid this, one has to permanently monitor the humidity levels and the ambient temperature.

Humidity will be removed by flushing with inert gas (N_2), nevertheless due to the low mass material that had to be used to build the enclosures water vapor will eventually always pass in. The measurement of the

relative humidity inside the boxes is mandatory and must be reliable enough to minimize the risk of condensation.

The critical parameters to monitor before and while the cooling down are temperature and relative humidity. These measurements are essential to compute the dew (or frost) point temperature, which compared to the target cooling temperature of the cooling system will show the margin we have to end up in water vapor saturation.

The calculation of the dew point is done using the Magnus formula [54] expressed as a function of the temperature and the relative humidity. This formula is a commonly used approximation.

$$\Rightarrow \quad Dp(T,RH) = \frac{\lambda \times \left(\ln\left(\frac{RH}{100}\right) + \frac{\beta \times T}{\lambda + T} \right)}{\beta - \left(\ln\left(\frac{RH}{100}\right) + \frac{\beta \times T}{\lambda + T} \right)} \quad [°C]$$

For the range from $-45\,°C$ to $60\,°C$, Magnus parameters are given by $\alpha = 6.112$ hPa, $\beta = 17.62$ and $\lambda = 243.12\,°C$. The formula can be simplified and the dew point can be computed as follows:

```
⇒  H = (log10(RH)-2)/0.4343 + (17.62×T)/(243.12+T)
⇒  Dp = 243.12×H/(17.62-H);  (this is the dew point in Celsius)
```

The election of the humidity sensing device was not a trivial task. The humidity sensors to be used should have a good resolution especially in low ranges (0-20%) and be operational at subzero temperatures. Although this already reduces the scope of valid sensors, the most critical restrictions are given by the environment where they have to be placed: the humidity sensors must be radiation hard; they must be small size; the sensing element should be away from the conditioning electronics to avoid having to develop the radiation hard components.

Conventional capacitive relative humidity sensors were not an option since they often include standard non radiation hard active components for signal conditioning and they cannot be operated far from the conditioning circuit when this is not an integrated part of the sensor. The same applies to dew point sensors based on thin film capacitance or surface resistivity. Another concern about these sensors is that they tend to be non-linear at low relative humidity levels.

Resistive humidity sensors are usually too bulky and the radiation hardness is still a concern. On top of this, the signal conditioning circuitry they need is frequently quite complex involving the generation of a variable-frequency sine wave modulation.

The MEMS (Micro-Electro-Mechanical Systems) strain gauge technology was proven to be quite robust while using semiconductor fabrication techniques for achieving low cost and SMD. The sensor in discussion is the HMX2000 sensor by 'Hygrometrix Inc.' [45]. This type of sensor was shown to be radiation hard and its performance nicely met our needs.

The HMX2000 was finally chosen as the unique possible option. It has to be clearly remarked that the sensor was chosen thanks to all the studies and huge work on testing and characterization of the sensor developed by the CMS tracker group [55].

6.3.1 THE READOUT CIRCUIT

The sensor itself consists of a wheatstone bridge with four sensing elements arranged as shown in the Figure 6.12.

The sensor has eight leads, only four of which are needed: *-Sig*, *+Vexc*, *+Sig*, *-Vexc* (see Appendix H). According to the manufacturer, the maximum allowed excitation voltage is 5 V and the nominal bridge resistance between bias pins is 4K0 Ω - 5K0 Ω.

The gains are set to $G_A = 51$ and $G_B = 1/3$. The value G_A was chosen such we used the maximum dynamic range of the amplifier output (~1.1 V to ~3.9 V). The sensor bias voltage is provided by a voltage reference in the CB and set to $V_{cc} = 2.5$ V. The calibration data provided by the manufacturer is for a bias voltage of 1.25 V, so it has to be scaled. Since the sensor consists of a wheatstone brigde the *+Sig* and *-Sig* values show linear dependence with V_{cc}, as one would expect from a voltage divider (see Figure 6.13). Therefore the sensor output signal V_{out} is linear and hence the output will be scaled by a factor 2 with respect to the manufacturer data. This linear dependence of the output respect to the bias voltage has been confirmed by the manufacturer.
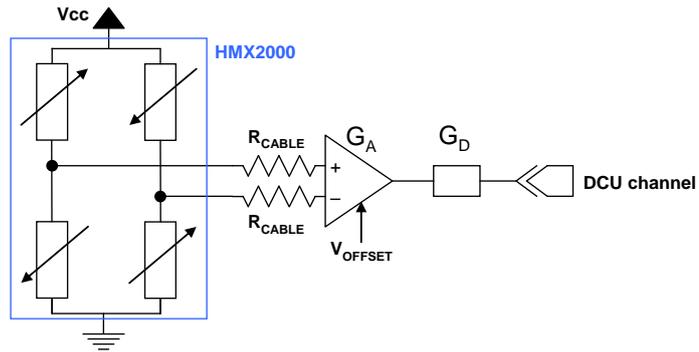
Figure 6.12: Scheme of the HMX2000 biasing circuit and signal conditioning.

$$+Sig = Vcc \times \frac{R - \Delta R}{2R}$$

$$-Sig = Vcc \times \frac{R + \Delta R}{2R}$$

$$Vout = (+Sig) - (-Sig) = Vcc \times \frac{-\Delta R}{R}$$
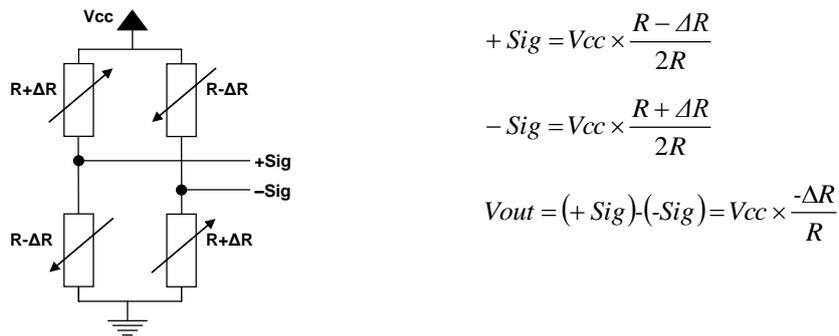
Figure 6.13: HMX2000 linear bias dependence.

The cable resistance has no influence in the measurement since the only current flowing is the amplifier input stage bias current, in the range of few µA.

6.3.2   THE HMX2000 SENSOR CALIBRATION

Sensors have been individually calibrated by the company in a temperature and humidity controlled climate chamber. The sensor was being biased with a voltage of 1.25 V, and the output voltage was recorded during the calibration as well as temperature and humidity values. The calibration data is measured at 15 points: 10.0 ℃ (10.0%, 20.0%, 30.0% RH), 5.0 ℃ (10.0%, 20.0%, 30.0% RH), 0.0 ℃ (10.0%, 20.0%, 30.0% RH), -5.0 ℃ (10.0%, 20.0%, 30.0% RH) and -10.0 ℃ (10.0%, 20.0%, 30.0% RH).
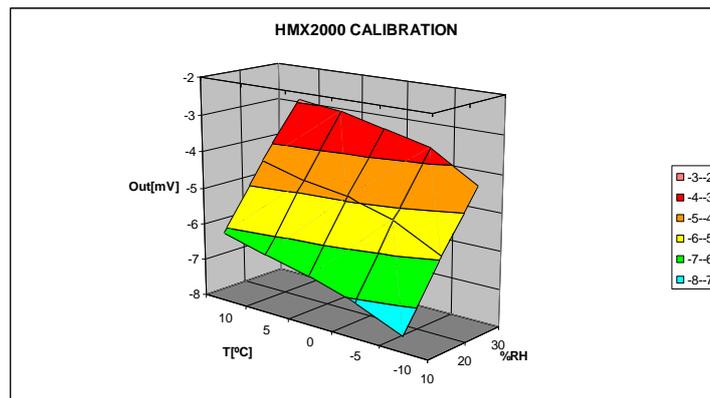
Figure 6.14: Calibration data of one HMX2000 sample.

The sensor is believed to show a linear dependence for temperature humidity changes. Typical calibration data (sensor output voltage as a function of humidity and temperature) as shown in Figure 6.14 reveals a slightly non linear temperature dependent behavior. A plane can be used to fit to these data:

⇨ $Vout = a + b \times (T\text{-}T_0) + c(H\text{-}H_0)$ [V],

where a and b are the fitted parameters. $T_0 = 0\ °C$ and $H_0 = 20\%$ relative humidity are chosen to be the center of the fitted plane in order to minimize correlations between the slope parameters a and b and the offset c. Studies performed by the SCT CMS group have pointed out the need of checking the calibration data provided by the manufacturer. Some batches of sensors clearly showed a large systematic skew in some of the measured calibration values and a way to identity this problem has been by checking the quality of the fit. None of the sensors ordered by the ST have shown a poor fit quality.

In the fit, a constant error on the output voltage of 15 mV is being used. This value has been obtained from the studies described in the CMS note [55].

Calibration equations generally hold over a wide range of temperatures.

$$\Rightarrow \quad H = H_0 + \frac{(V - a - b \times (T - T_0))}{c} \quad [\% \ RH],$$
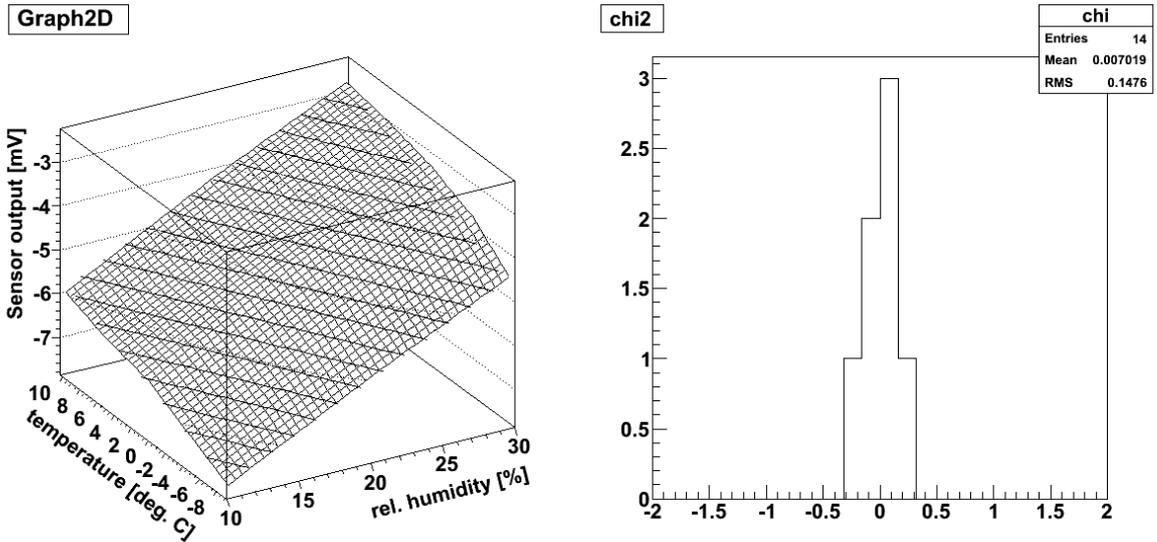
Figure 6.15: On the left the HMX200 manufacturer calibration data and the fitted plane (contour lines). On the right the residuals of the manufacturer calibration data values with respect to the fitted plane.

The calibration data from the manufacturer is fitted to a plane. Since the sensors are biased with twice the voltage (2.5 V) the manufacturer used for their calibration (1.25 V) and the output voltage is proportional to the bias voltage, the resulting parameters from the fit are multiplied by two. The Figure 6.15 shows an example of the calibration data provided by the manufacturer and the fitted plane is shown with contour lines. In the right we can see the residuals of the manufacturer calibration data values with respect to the fitted plane. The *rms* of the residuals is below 0.15 mV for all the sensors. Roughly speaking, for a humidity range from 10 to 30% RH, the output swing of all the humidity sensors is usually around 3 mV. This results in 0.15 mV/%RH and thus indicates that the error introduced by fitting to a plane is equivalent to a ~1% RH. The results from fitting the manufacturer calibration values to a plane can be found in appendix J.

To improve the fit quality, the manufacturer data can be fitted to a second order surface of the form:

$$\Rightarrow \quad Vout = offset + sT \times (T\text{-}T_0) + sT_2 \times (T\text{-}T_0)^2 + sH \times (H\text{-}H_0) + sH_2 \times (H\text{-}H_0)^2 + sTH \times (T\text{-}T_0) \times (H\text{-}H_0)$$

- $T_0 = 0\ °C$
- $H_0 = 20\ \%\ RH$

The size of the residuals is lower in this case, so the error introduced by the fit. However, the experience with these sensors has shown that the most important source of errors in the measurements is due to the non linear behavior of the sensor below 10% RH. The tightness of the detector boxes in conjunction with the flushing of N^2 boxes entails that the sensors are normally being operated below 10% RH. It is due to this fact and for easiness that in the monitoring system just the fit to plane has been implemented. Nevertheless, in appendix J the results from fitting to a second order bi-dimensional function are provided.
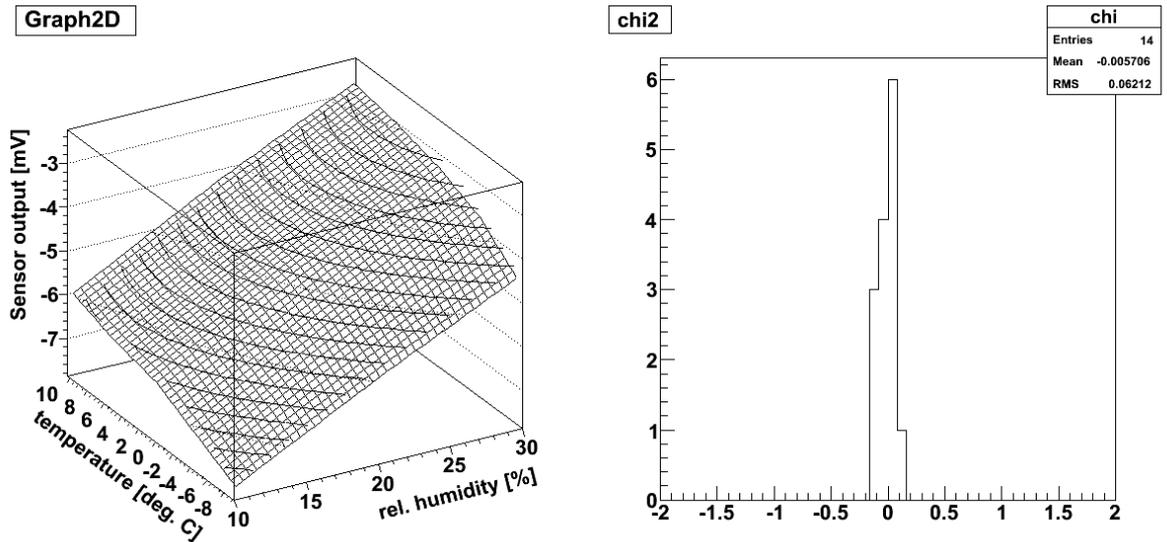
Figure 6.16: On the left the HMX200 manufacturer calibration data and the fitted second order surface (contour lines). On the right the residuals of the manufacturer calibration data values with respect to the fitted plane.

6.3.3   THE READOUT CHANNEL CALIBRATION

In the same way as for the temperature DCU channels, the humidity sensor readout line needs to be calibrated. In this case, as the humidity sensor generates a voltage signal the calibration obtened calibration curve represents ADC readings vs. voltage.

The setup used for the calibration is exactly the same one as for the CB tempetures (see section 6.2.2). The calibration board was populated with some high precision resistor that generated 5 different differential voltage in the range of mV: -16.326 mV, -10.884 mV, -5.442 mV, 0 mV, 5.442 mV. All channels have shown a perfect linear fitting to the acquired calibration data.

DCU CHANNEL	AVERAGE (ADC)	RMS (ADC)	AVERAGE (mV)	RMS (mV)
CH3	2401.7901	1.4717	1.1119	0.0007
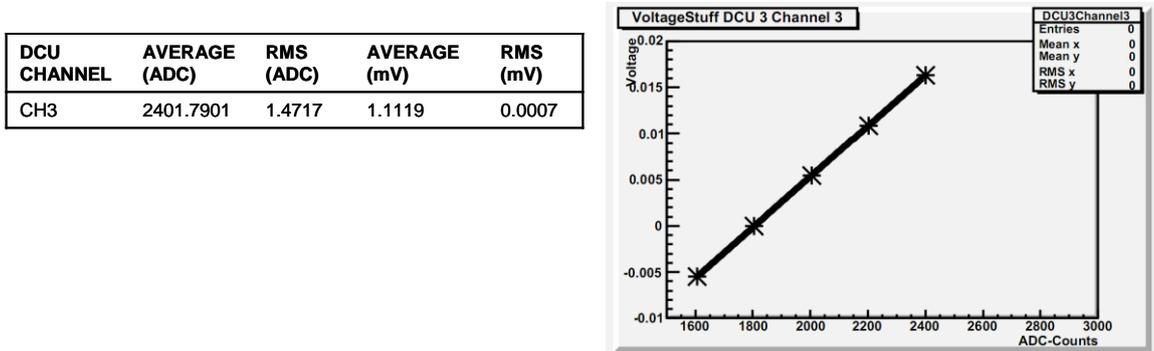
Figure 6.17: Calibration data sample for one resistance value of one DCU humidity channel (left figure) and a calibration curve of one channel (right figure).

The results of the calibration for each CB can be found in appendix G.

6.3.4   THE HUMIDITY MONITORING

One HMX2000 humidity sensor is installed in each IT detector box. The sensor is placed in the feedthrough interface PCB. The associated readout CB is the same as for the temperature sensors. The humidity sensors to detector boxes mapping as well as the connection pinout was already presented in .

The TT detector box contains eight humidity sensors, named *H1* to *H8* as shown in Figure 6.6. Each sensor is read out by a CB, as indicated in the same figure. A PT1000 sensor (*C1* to *C8*) was placed next to each humidity sensor in order to have a good estimation of the temperature, needed for the calculating the humidity from the HMX2000 measurement.

The calculation of the humdity is performed basically in three steps:

1. Read out the HMX2000 voltage output, where *ChDCU* is the ADC reading, a_v the slope and b_v the offset parameters of the calibration curve associated to HMX2000 readout channel.

$$V = ChDCU[adc] \times a_v + b_v \,[\text{V}]$$

2. Calculate the ambient temperature as explained in 6.2.3 and 6.2.4.

3. Calculate the relative humidity using and the HMX2000 corresponding calibration curve parameters. The gain factor 2 of the sensor due the biasing voltage is already taken into account in the calibration curve parameters.

$$H = H_0 + \frac{(V - a - b \times (T - T_0))}{c} \,[\%\text{RH}]$$

Once the relative humidity is known the dew point can be computed using the Magnus formula as explained in 6.3.

6.4 THE DCU CONFIGURATION AND READOUT

See appendix I to K.

CHAPTER 7

THE ST DAQ SYSTEM INFRASTRUCTURE

ABSTRACT

This chapter provides an overall technical description of the Silicon Tracker DAQ (Data Acquisition) front-end electronics infrastructure. An overview of the ST electronics has already been depicted in chapter 3 so in this one we will focus on the specific aspects that are important from the ECS point of view and also to set the system into operation.

"El amor es algo difícil de explicar, fácil de sentir e imposible de olvidar"

Anónimo

7.1   INTRODUCTION

This chapter describes in detail the DAQ system elements of the LHCb Silicon Tracker, composed of the Inner Tracker (IT) and Tracker Turicensis (TT) sub-detectors. The information reported in this note is indispensable for understanding how the DAQ data flow and control is like in the ST detectors and it is thus indispensable for any expert who wants to perform any development of the ST control system. Due to the nature of the information in this note, some hardware descriptions of the system and of parts of it will be necessary in concrete moments.

The description of the DAQ system will be issued such that a global description of the whole readout link will be given first and then each of the different parts will be described in more detail:

- The very front-end electronics in the detector boxes.

- The digitizing electronics in the service boxes.

- The L1 electronics in the counting room.

In addition to the functional description of system elements a map of the whole DAQ infrastructure is also shown. The way the electronics interfaces with the Experiment Control System (ECS) is described as well.

7.2   THE ST DAQ SYSTEM DESCRIPTION

The Silicon Tracker consists of silicon strip detectors with a pitch of around 200 μm. For the TT station upstream of the magnet, a 500 μm thick sensor with 512 strips is used, while the three IT Stations after the magnet feature 320 and 410 μm thick sensors of 384 strips each.

The analogue FE chip Beetle amplifies the weak detector signals with minimum noise and sufficient time resolution, to determine the exact bunch crossing from which the particles originate. The captured data remain stored in the level 0 trigger pipeline buffer until the level 0 trigger decision accepts from a given bunch crossing. The accepted data are extracted from the pipeline buffer, temporarily stored in the L0 derandomizer buffer and transmitted via differential analogue lines to the Service Boxes (SB). Here, the signal is digitized and converted from electrical to optical, and then it is transmitted to the counting room along a signal path about 100 m to the TELL1 optical receiver card ('L1 Electronics').

The event data received on L1 electronics side from the different input links will have some timing skews owing to the fact that they arrive from different detector parts and have variations in cable lengths. The input skew is limited to a few clock cycles and the data alignment between different sources is performed with small data buffers. The correct reception of event data is continuously verified and data tags in the event fragments are used to verify the correct function and synchronization of the L0 FE electronics. After successful verification the data are processed and packed to be sent to the HLT farm.

An overview of the complete data readout link is shown in Figure 7.1 and covered in detail in [22]. It shows the clear split off between the so-called 'L0 Electronics' located in the cavern and the 'L1 Electronics' in the counting house.

7.2.1   THE READOUT HYBRID

The front-end hybrid placed in the detector boxes consists of the Beetle [17] chips mounted together with passive electronic components on a flexible printed circuit board (PCB) and of ceramic pitch adapters to match the silicon sensor pitch to the pitch of the Beetle input pads. This is the very front-end electronic module for the readout of the silicon strip detectors. It is constructed in multilayer flexible printed circuit board technology which allows for a feature size of 50 μm, which simplifies the layout. The two main hybrid types that are employed by the Silicon Tracker carry either three Beetle chips for the IT detector modules or four chips for the TT detector modules. That makes a total of 384 readout channels for the IT hybrid and 512 for the TT. The number of channels is therefore identical to the number of strips of the corresponding silicon strip detector.

The Beetle chip integrates 128 charge signals from the silicon sensors. The Beetle channels consist of a low-noise charge-sensitive preamplifier, an active CR-RC pulse shaper and a buffer. The shaped signal is sampled with the LHC bunch-crossing frequency of 40 MHz into an analogue pipeline which has a programmable latency of maximum 160 sampling intervals and an integrated multi-event buffer of 16 stages, known as derandomizer. For test and calibration purposes a charge injector with adjustable pulse height is implemented on each channel. The bias settings and various other parameters like the trigger latency can be controlled via I^2C and the addressing is as defined in the figure annexed (extracted from [56]). All digital

control and data signals, except those for the I^{2}C ports, are routed via LVDS ports. Appropriate design measures have been taken to ensure the radiation hardness against total dose effects in excess of 100 Mrad and robustness against Single Event Upset is achieved by redundant logic.
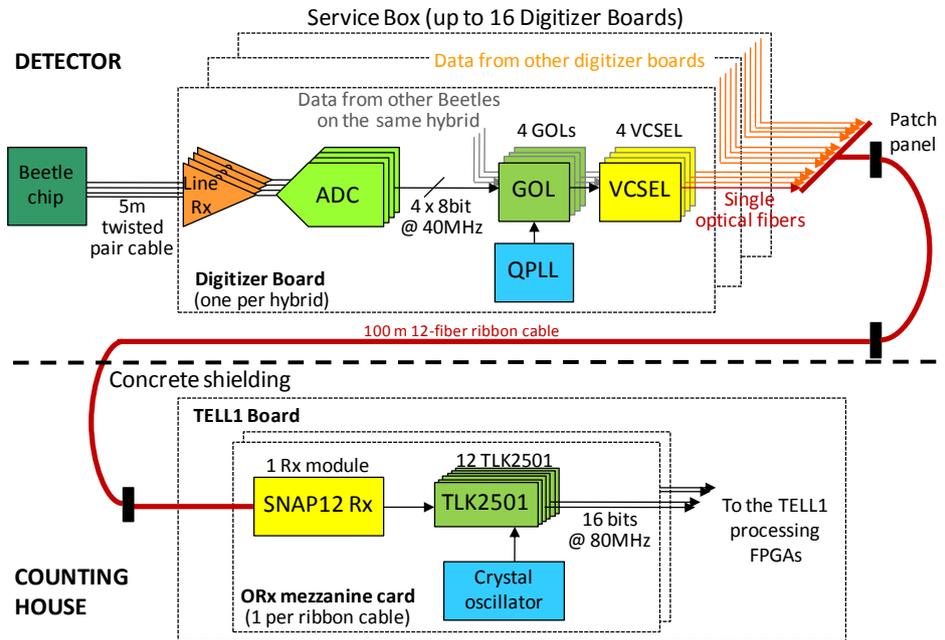
Figure 7.1: The IT readout link.

Upon a L0 accept reception, the Beetle reads the pipeline information and transfers it to the multiplexer via a resettable charge-sensitive amplifier (*pipeamp*). Within a readout time of 900 ns current drivers bring the serialized data off chip through 4 output ports. In each port is sent the data from 32 channel plus additional information such as the pipeline column number. The 900 ns latency is compatible with the maximum Level-0 accept rate of 1.1 MHz, as specified in [57]. A binary signal (*DataValid*) is sent in parallel to facilitate the synchronization of this analogue data frame. The duration of this signal is shortened by one clock cycle to 875 ns to provide a signal gap even for consecutive readouts and starts one clock cycle (25 ns) earlier than the analogue readout to allow a possible setup of subsequent processing stages. The analogue outputs are designed as fully differential current drivers, capable of driving several meters of impedance controlled differential lines. The differential outputs as well as all fast digital inputs are designed as LVDS interfaces to minimize EMI.

Several versions of the Beetle chip were produced, and ST decided to use the version 1.3 as it fulfilled the detector requirements. A basic block diagram is shown in Figure 7.2.

Commercially available I^{2}C devices usually operate at 3.3 V or 5 V. With the Beetle version 1.3 or higher, such external devices can be directly connected to the Beetle I^{2}C interface.

The Beetle contains 24×8-bit registers with the addresses 0-23. See in Appendix N the list of all registers and specifications. Detailed information about the Beetle register description can be found in [56].

7.2.2   THE SERVICE BOX

While the silicon sensors and the hybrids are located in the detector boxes, the digitizing and control electronics is located in the service boxes, out of the detector acceptance. This reduces the amount of dead material and hence multiple scattering and generation of secondary particles. In addition, the location of the SB is exposed to lower radiation levels and allows for easier access for cooling and maintenance.

Each SB houses up to 16 Digitizer Boards, which handle the signals coming from and going to the readout hybrids. As this includes not only the physics data, but also power supply, timing and slow control signals, this allows for a single cable connection between a hybrid and its associated Digitizer Board. For each Service Box, a Control Board provides the interface to the experiment-wide Timing and Fast Control (TFC) system and the Experiment Control System. The distribution of these signals to the Digitizer Boards within a Service Box is done with a custom designed backplane. The backplane also carries the radiation tolerant linear voltage regulators to supply the readout hybrids and the SB itself.
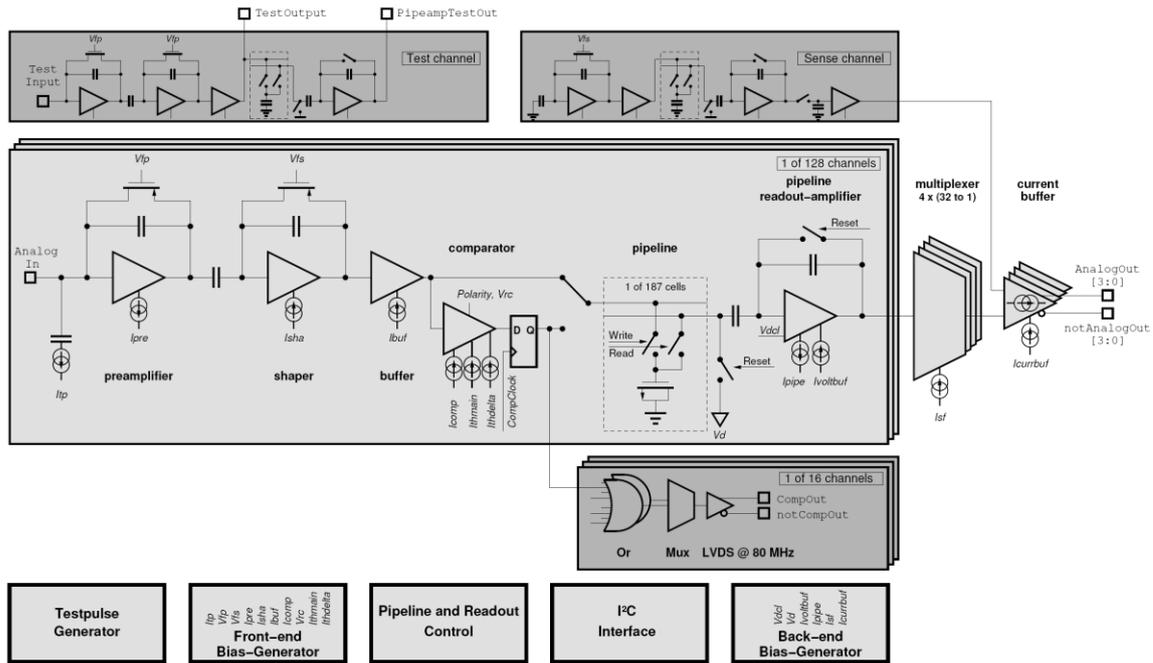
Figure 7.2: Beetle chip architecture.

7.2.3   THE CONTROL BOARD

The CB has thoroughly been described in chapter 3. As a summary, we will mention the most important points for the purpose of this chapter. The SPECS slaves mezzanines [31] provide the connection to the ECS and the TTCrq mezzanine [41] is used to get the clock, trigger and fast commands from the TFC network. The SPECS slave mezzanine provides eight I^{2}C busses for complete control of the devices in one Service Box, plus 64 I/O lines that permit individual power control for each readout hybrid and for a group of four DBs.

The TTCrx chip in the TTCrq is the timing receiver ASIC which delivers the TTC (Timing and Trigger Control) [42] signals to the front-end electronics. The TTCrx can be programmed to compensate for particle times of flight and for propagation delays associated with the detector and the electronics. An I^{2}C interface allows to read and to write (or reset) all internal registers, which are listed in Appendix N. The ST front-end electronics uses the following TTC signals:

- *Clock40Des1*: 40 Mhz clock for the Beetles.
- *Clock40Des2*: 40 Mhz clock for the ADCs and GOLs in the Digitizer Boards.
- *L0accept*: Trigger signal for reading out the Beetle data.
- *L0reset*: Reset signal for the Beetles external reset.
- *calib0*: Test signal for injecting a charge pulse in the Beetle input channels.

These signals are driven along the CB in LVDS mode for signal integrity reasons. The *L0reset* and the *calib0* are decoded from the TTC broadcast command in the SPECS slave channel B decoder. The SPECS slave features control bits to disable either the channel B decoder or the individual signals (Appendix N shows the list and description of SPECS internal registers). The calibration signal can also be coarse delayed in the slave up to 32 clock cycles and fine delayed in the Delay25 chip up to 64 steps of 0.5 ns. The *L0ACCEPT* (*L1accept* in the TTCrx manual naming), *L0reset* and *Clock40Des1* also pass through the delay chip so the relative phase can be tuned in order to avoid setup and hold timing violations in the front-end. The Delay25 device is also interfaced by I^{2}C using the addressing that we will explain in 7.4.1.

The SPECS slave internal state machines are clocked with the experiment clock *Clock40Des1*. The TTC broadcast command is synchronized with respect to *Clock40Des1* so the SPECS decoder needs it as reference clock for the *L0reset* and *calib0* decoding.

Each slave provides 32 I/O lines, some of which are used for the DAQ devices control and monitoring, namely: resetting the TTCrq, the TTTCrq QPLL, the DBs QPLL and the GOLs; monitor the status of the

READY, *LOCKED* and *ERROR* signals of the TTCrq; enable and disable the Control Board clock drivers. The bit assignment is shown in Appendix N. The SPECS addressing will be described in 7.4.2.

As the structure of the IT and TT is very similar but with a different partitioning scheme, the same CB is used for both.

7.2.4   THE DIGITIZER BOARD

Each readout hybrid is connected with a 68-wire twisted-pair round cable to a Digitizer Board. First, a line receiver converts the analogue detector signal from a differential signal into a single-ended signal (the Figure 7.1 depicts the diagram of a single analogue input stage). In addition to the elimination of common mode noise introduced in the cable, the line receiver matches the signal amplitude to the input of the 8-bit analogue-digital converter (ADC). Data of four ADCs, corresponding to the four analogue output ports of one Beetle are encoded by the CERN GOL chip [28] into the Gigabit Ethernet protocol and modulated onto a 850 nm VCSEL diode. The digital data are framed by using the *DataValid* signal of the Beetle readout hybrid to drive the *TX_EN* line of the GOL serializer. An antifuse FPGA from 'Actel Inc.' was used to form a 7-clockcycle triple-redundant shift register. This re-aligns the *DataValid* signal to the digitized data and allows for automatic re-synchronization of the optical link between consecutive physics triggers. Two versions of Digitizer Boards have been designed: a TT version with 16 ADCs corresponding to a 4-chip TT hybrid and an IT version with 12 ADCs only to match a 3-chip IT hybrid.

Figure 7.3 shows the DB constitutive elements. The interface to the readout hybrid is seen on the left, where the long SCSI copper cable is connected. The connector on the right is the interface to the SB backplane, providing power and timing and control signals for the DB and its associated readout hybrid. The figure also shows the channel assignment of a TT Digitizer Board. As a TT hybrid features four Beetle readout chips, 16 analogue channels have to be digitized. The DB for the IT is similar but digitizes only 12 analogue channels. Four circuit blocks are used to process the data from the four Beetles. Each block uses four line receivers with four ADCs, one GOL and one VCSEL. The link synchronization circuit is located at the top left, while a QPLL clock clean-up circuit and a slow-control ADC are located on the bottom right. In the center of the board, the root of the clock distribution tree can be seen.

For the purpose of this chapter we have to remark the GOL serializer provides different operational modes, some of them programmable via I^2C, others to be determined by connecting special pins to either ground or V_{DD}. While some of the hardwired parameters are completely fixed, other values, for example the PLL charge pump current and the mean laser diode current, can be reprogrammed via the I^2C interface. For these parameters, the hardwired value represents the power-up default, which is set before any user intervention. The DB also implements a DCU for slow control monitoring. From the DAQ stand point the only important signal that is read out by this device is the QPLL status of the DB in DCU channel 1.

7.2.5   THE BACKPLANE

The backplane interconnects the Digitizer Boards to the Control Board. The distribution of timing and fast trigger signals is done via impedance controlled differential traces. Equal trace lengths were maintained to keep the time skew of the fast signals among different backplane slots minimal.

The backplane defines the detector partitioning. The TTC signals and I^2C buses distribution together with the power regulators grouping is done in the backplane such that it follows the partitioning into groups of 4 slots for the IT and 3 slots for the TT service box. Therefore, the IT and TT backplane are different with 16 and 12 slots respectively.

The TTC signal distribution consists of the clock trees for the ADC/GOL devices and the Beetles, the *L0accept*, the *L0reset* and the *calib0* signals. The Control Board is the only signal source and the up to 16 Digitizer Boards are the signal receivers, so a strict tree structure was designed to drive the signals to all the boards.

In addition to the TTC signals, the backplane also distributes the I^2C buses to the readout hybrids and DBs, as well as two reset lines: one for the GOL and one for the QPLL.

The I^2C addressing scheme explained in 7.4 contains two bits, labeled 'BOARD-ID', which assign part of the I^2C addresses to the bus devices. These two bits are set by plugging a Digitizer Board into a slot of the backplane, as the corresponding pins of the backplane connector are hardwired to ensure a unique address for each I^2C bus.
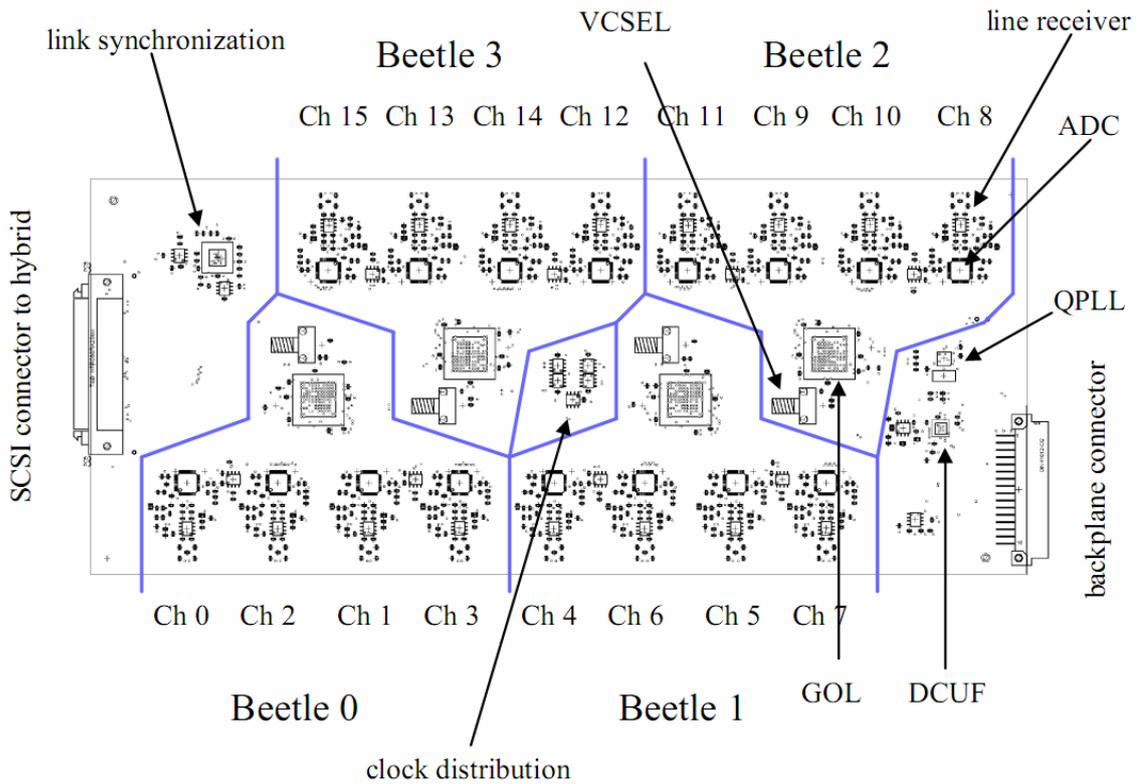
Figure 7.3: Digitizer Board elements and Beetle assignment (TT type). Picture taken from [21].
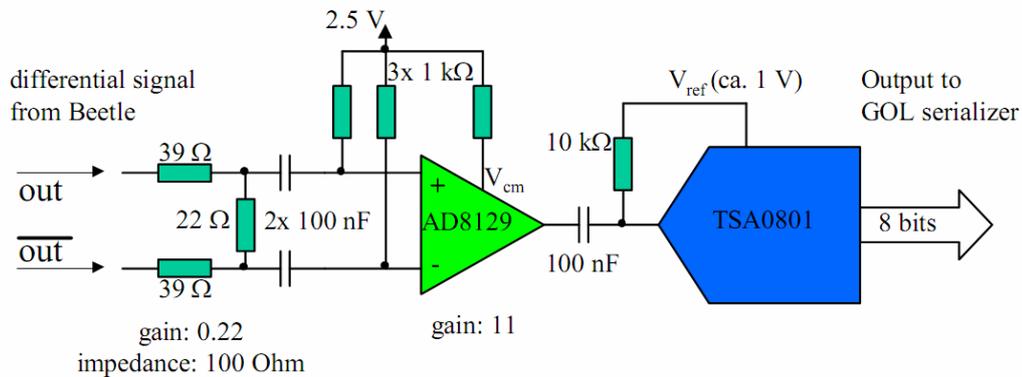
Figure 7.4: Block diagram of a single analogue input stage of the Digitizer Board. Picture taken from [21].

7.2.6 THE TELL1

The 'L1 electronics' is the part of the DAQ system handling L0 accepted data before transmission to the DAQ (see block diagram in Figure 7.5). The implementation of each level is in general specific to each sub-detector. On the other hand, it has been recognized that all the sub-detectors require very similar L1 functionalities. Hence, it was decided to develop a common 'L1 board' with enough flexibility to adjust to the different needs, with evident rationalization in the development effort, construction and maintenance.

The data are transferred by optical fibers from the SBs to the counting house. There, the TELL1 [23] boards perform de-serialization, data processing and final transmission of the filtered and compressed data to the L1 and HLT trigger CPU farm. Each ST TELL1 can handle up to 24 input optical links, which means that it is able to process the data coming from 24 Beetles and hence 8 IT or 6 TT readout hybrids.

The 'L0 electronics' is placed close to the detector and requires radiation hard or radiation-tolerant components. The 'L1 electronics' is placed behind a shielding wall, quite far from the interaction point (more than 60 m), to minimize the amount of electronics prone to radiation effects.
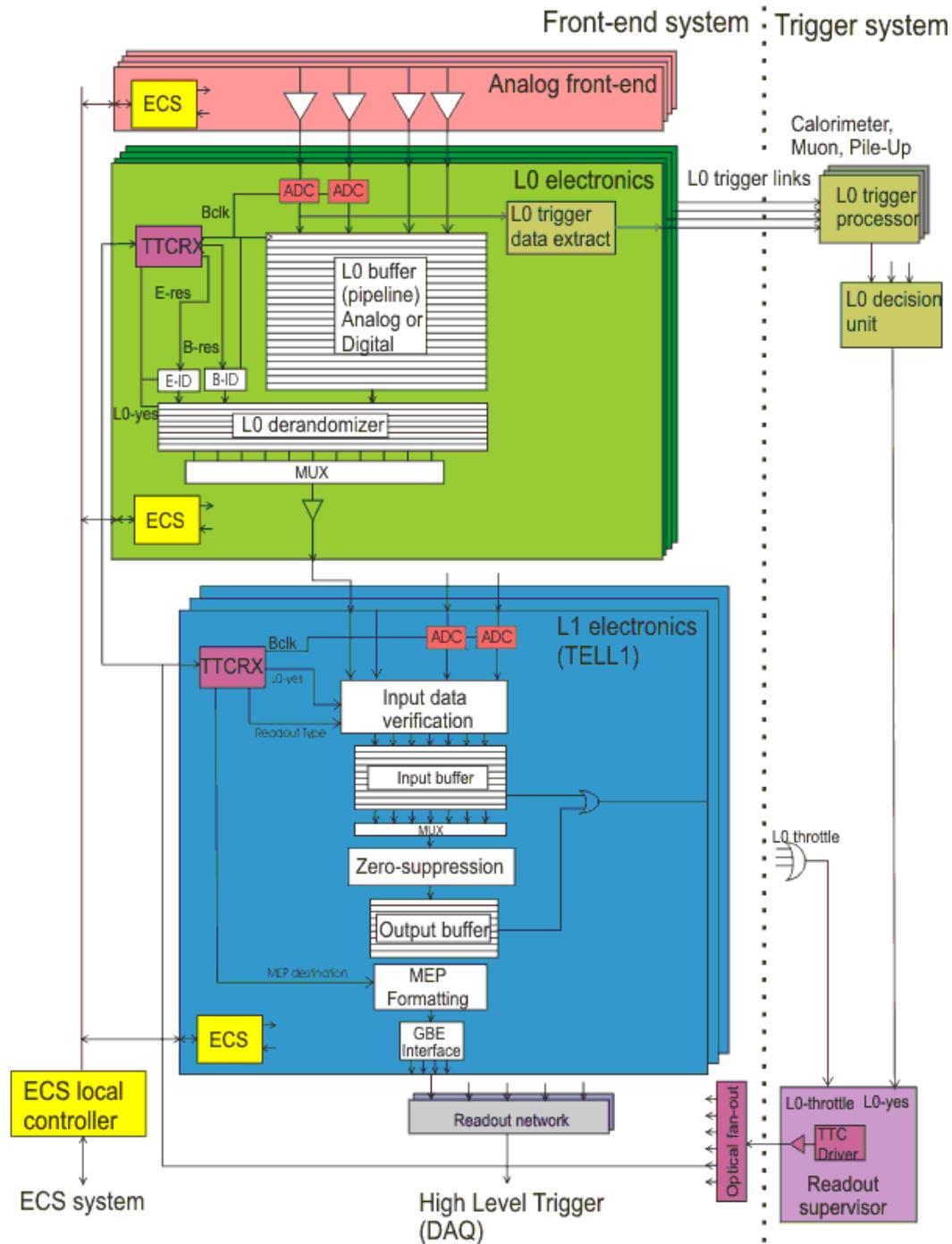
Figure 7.5: Block diagram of front-end electronics and its interface to Trigger, DAQ and Detector Control System.

After the *L0accept* trigger signal, the data are transferred by optical lines to the counting houses. There the TELL1 boards perform deserialization, noise filtering and zero suppression, followed by transmission of the compressed data to the L1 and HLT trigger CPU farm. Physically the TELL1 is composed of a motherboard hosting several mezzanine cards. Its main processing elements are five large FPGAs that are equipped with optical receiver cards (O-Rx cards) for direct connection to the optical fibers coming from the detector:

- The ORx mezzanines are employed for the reception of the optical signals. Each one of the 2 ORx cards uses two connectors and implements a 12-way optical receiver resulting in a total of 24 channels per TELL1 board.

- *PP-FPGA*: four 'PreProcessor' FPGAs receive from the ORx cards L0 accepted data over a parallel link at a maximal event rate of 1.11 MHz. The pre-processing consists of noise filtering, zero suppression and subsequent transfer to the *SyncLink-FPGA*.

- 109 -

- *SyncLink-FPGA*: the "Synchronization and Link" FPGA is connected to the TTCrx and distributes synchronization signals as clock, trigger and event identification to the *PP-FPGAs*. It collects and merges the data fragments from the *PP-FPGAs*, assembles multiple events into so-called multi event packets (MEP), performs Ethernet and IP framing before transmission to the readout network.

- Quad Gigabit Ethernet (GBE): this is a four port Gigabit Etherner card that interfaces to the HLT network. Its output data bandwidth is up to 4 Gbit/s.

- TTCrx: the on-board TTC receiver chip TTCrx receives the following TFC signals: the clock (locked at 40.079 MHz), the *L0reset*, the *L0accept* and the DAQ IP destination address.

- Throttle system: 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 trigger.

- ECS: the TELL1 board slow control is managed by a commercial Credit Card-sized PC (CCPC) running a Linux kernel. A small adapter card (Glue Card), linked to the CCPC via PCI, provides all necessary interfaces to control and monitor all the other devices. The FPGAs and the quad Gigabit Medium Access Controller (GBE MAC) are on a shared microprocessor bus driven with a PCI bridge. I^2C and JTAG interfaces are also provided for convenient programming of the FPGA firmware EEPROM and monitoring the devices. The disk-less CCPC is attached over the network to a boot server.
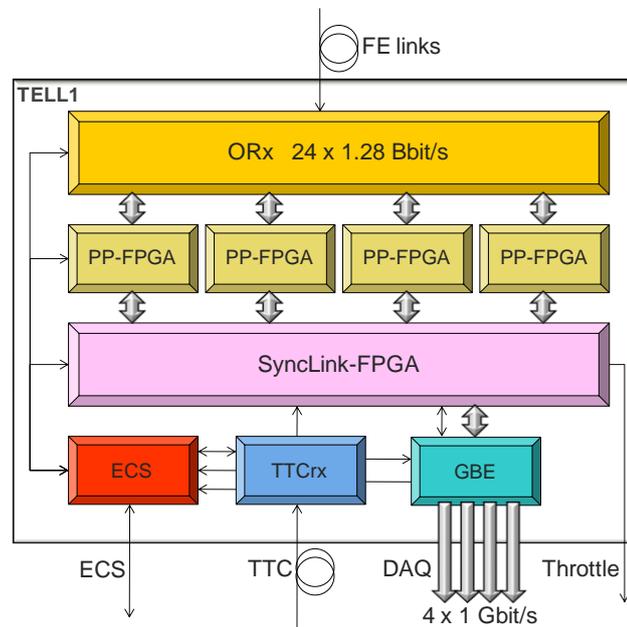
Figure 7.6: Simplified block diagram of TELL1 equipped with ORx receiver cards.

Next we explain the several processing stages that take place in the TELL1.

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*' (or 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). An idle cycle is ensured always in between of every consecutive event so the event synchronization is preserved at the TELL1 side. Due to optical transmission failures, caused by faulty low power VCSEL laser transmitters in the DB, it was necessary to improve the performance of the data synchronization by using a simple but robust algorithm. This algorithm consists in inspecting the status of the *DataValid* (DV) signal from the deserializers (Figure 7.7) and filtering the spurious glitches:

1. Two counters count the high and low period of the DV signal. The DV is in the 80 MHz domain so it should stay high for 70 cycles for a complete event. 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

- 110 -

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 the 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.
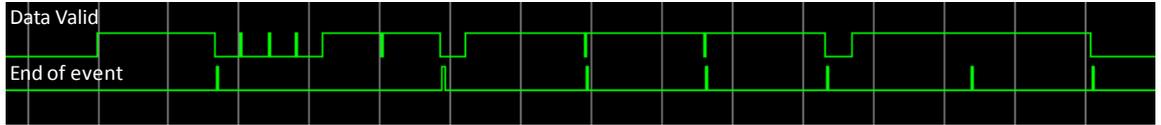
Figure 7.7: TELL1's input synchronization simulation of "Data Valid" and "End of event" signals. The simulation shows the 'Data Valid' and the 'End of event' signals for different scenarios: starting from the left, a normal event, few noise glitches, a "split" event, few consecutive events and a conglutinated event.

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 ALGORITHM

After pedestal correction, the ADC count of the first of each 32 read-out strips is also corrected for header crosstalk, as it is adjacent to a header strip with a value much deviant to the read-out strips. Here, linear common mode noise suppression (LCMS) is performed per set of 32 read-out strips, as is shown in the schematic (Figure 7.8). 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 LCMS 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 LCMS corrected data using detector specific hit finders and clusterization procedures. In a first step, the seeding strips are found using strip individual S/N 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.

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 multi-event packets prior to transmission to the network.

7.3 THE DAQ ELECTRONICS PARTITIONING

The grouping and distribution of the TT and IT readout of the different SBs has been a non trivial challenging task. Due to the large number of channels and devices consisting of the detector, it becomes necessary a physical and logical grouping of the DAQ electronics. This grouping must not only respond to the needs of readout structuring, but also to needs in terms of system control, reconstruction performance in case of hardware failure and TELL1 load. The readout partitioning is described in detail in [48] .

Each SB contains up to 16 (12 in the TT) DBs, which means that handles signals (control and data readout) of 16 (12) readout hybrids and thus 48 Beetles. The TELL1s were designed to receive and handle the data

from 24 Beetles, arranged in 2 fiber bundles of 12 links, and the front-end hybrids were designed such they house 3 and 4 Beetles respectively for IT and TT. Thereby, it makes sense to create basic partitions of 4 hybrids in the IT and 3 in the TT, which correspond to 12 Beetles.

The CB and SB backplane have been designed and built such that the hybrid/DB partitions are independent from each other in terms of slow and fast control[1]. The only common point is that they are all controlled by the same CB for each service box. The partitions of 4 hybrids/DBs (3 in the TT) are defined by plugging into specific slots of the backplane, as shown in the Figure 7.9. As each hybrid is associated to a DB, it implies that the grouping at the hybrid level is defined by the DB grouping in the service box, so it is the hybrid to DB cabling which will define the hybrid partitioning.
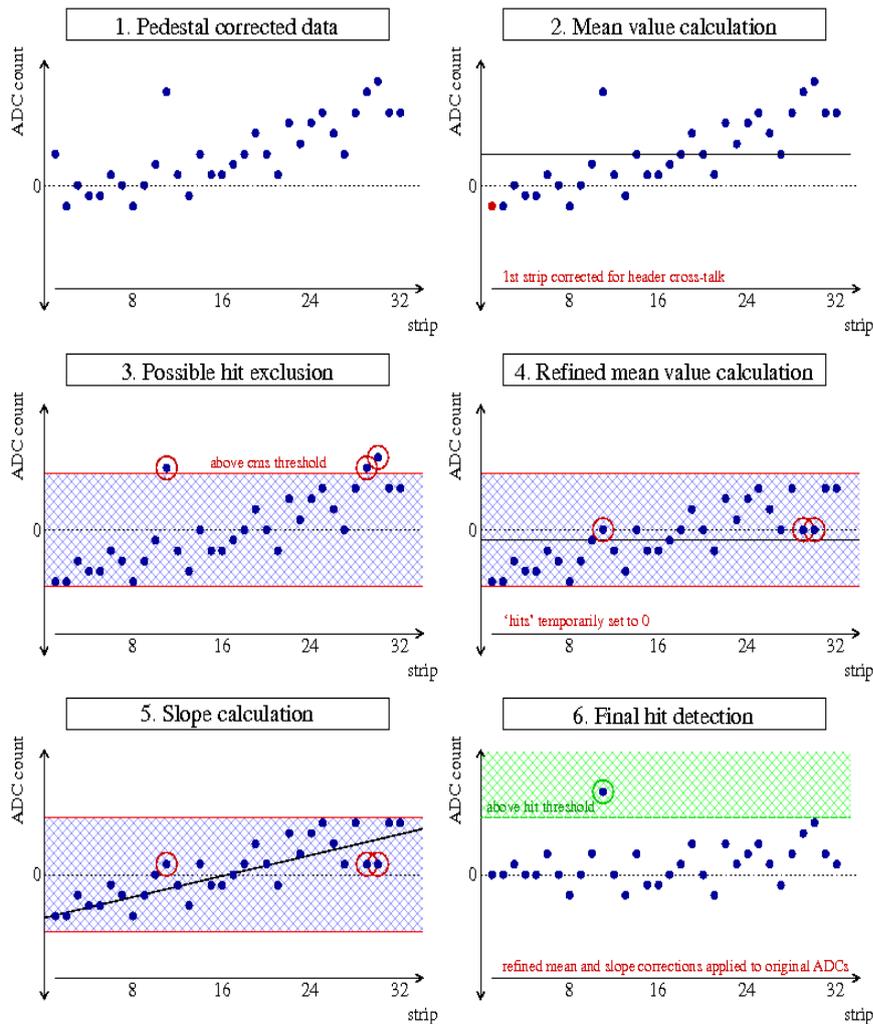
Figure 7.8: LCMS algorithm. In the step one is shown an example of pedestal corrected data for one Beetle port. In the second step the header cross-talk of first strip has already been corrected and the mean valued calculation is performed. In the step 3 the hits that are above the CMS threshold are identified and will be set to temporarily to "0". In the step 4 a refined mean value calculation is carried out and in the fifth the slope of the data set is calculated. Finally, in the last step the resulting data from applying the refined mean and slope corrections is shown. The data is now noise filtered so the final hit detection can proceed.

---

1 The slow control tasks of temperature, humidity and LV monitoring as well as the FE devices programming is carried out at the service box level by the CB, but the HV detectors biasing is performed by a detached system which for affinity also follows the same hardware partitioning.
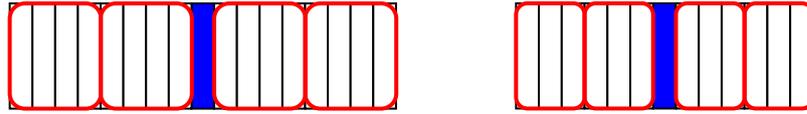
Figure 7.9: Scheme of the SB partitioning for the IT (left figure) and TT (right) looking from the front. The slot for the CB is shown in the center in blue color, the slots for the DBs in white and the partitions are outlined in red.

As the optical links leaving the SBs and reaching the TELL1 are bundled into 12 channel fibers, each service box delivers up to 4 fiber bundles, one per group of 4 (3) DBs. Each TELL1 handles up to 24 optical links coming from the two 12-link input connectors, so it is reasonable that from the standpoint of the readout, the TELL1s are allocated to areas covered by a single service box to the possible extent.

7.3.1 THE IT PARTITIONING AND MAPPING

Each of the three Inner Tracker stations corresponds to 112 readout sectors and 14 TELL1 boards, which make a total 42 to read out the entire detector. The readout partitioning is shown in Figure 7.10. Since an Inner Tracker box contains 28 ladders it is necessary to share a TELL1 with two different detector boxes. The IT stations are constructed as two retractable frames, one per half-station, so the readout follows the same grouping: top and cryo-side boxes into the "cryo" half-station; access and bottom box into the "access" half-station.

In the IT readout grouping sketched in the figure each black or hashed rectangle represents one readout sector. The black and grey shading on the sectors correspond to different optical fiber bundles and the red outline corresponds to a TELL1.

Figure 7.11 shows the mapping between the readout partitions and the service boxes. Each color in a box represents a partition of 4 hybrids that maps with 4 corresponding DBs in the service boxes. As each box houses 28 hybrids, 2 services boxes are needed to read it out, one of them instrumented with 16 DBs and the other one with just 12. Therefore, 4 SBs are needed per half-station and 24 for the whole IT. The service boxes are placed just below the detector at the bottom of the half-station frames in groups of four.

The items representing the SBs, arranged in groups according to the station and box to which they belong to, show the matching with the box depending on the color. A white color in the SB means that the slots are not filled in with DBs, and the dark blue color indicates the CB slot. The service boxes are numbered from 1 to 4 outwards the beam pipe. It is important to note the asymmetry in the SB mapping for the top box in the half-station 2 due to a cable length constraint with the data cables from the DBs to the hybrids. Detailed pictures showing the hybrid to DB assignment can be found in Appendix L.
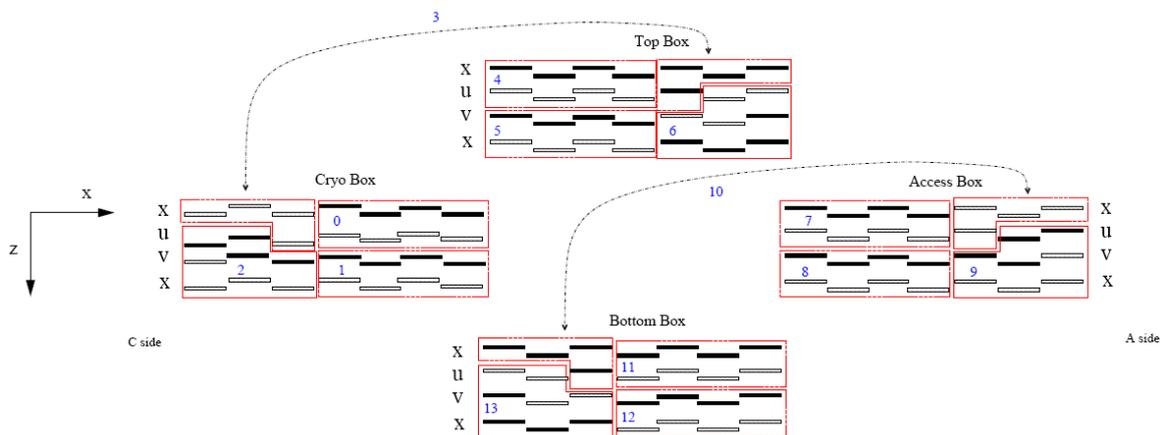
Figure 7.10: IT partitioning. Picture taken from [48].
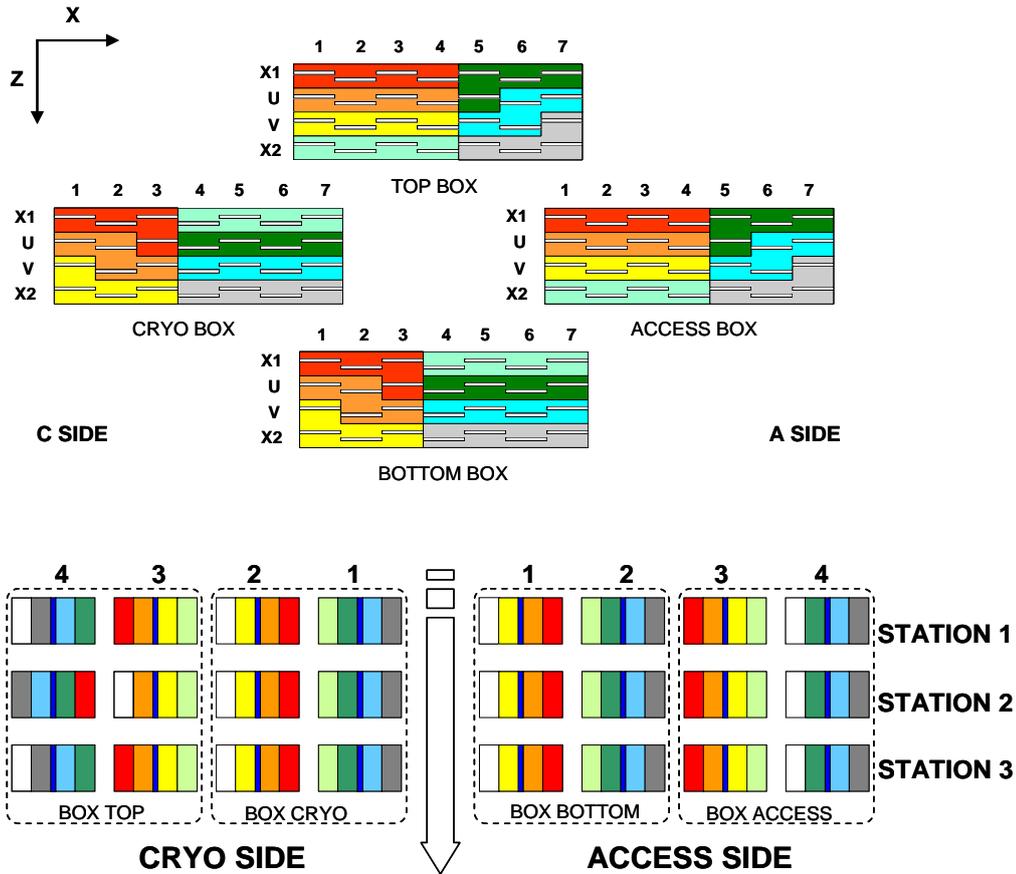
Figure 7.11: IT detector box to SB assignment. In each SB the four partitions are highlighted with a different color with the CB in the middle in dark blue. The white boxes indicate that the partition is empty. The SB 3 and 4 of station 2 have a slightly different mapping due to cable length constraints.

7.3.2   THE TT PARTITIONING AND MAPPING

The TT partitioning is shown in Figure 7.12 for the two stations 'TTa' and 'TTb'. A total of 48 TELL1 boards are needed to read out the TT detector.

As in the IT picture, each black or hashed rectangle represents one readout sector (Figure 7.12). The black and grey shading on the sectors corresponds to separate optical fiber bundles which correspond to the readout of three TT hybrids. The partitioning of the sectors into quadrants corresponding to TELL1s is also shown. Positive 'z' is out of the page.
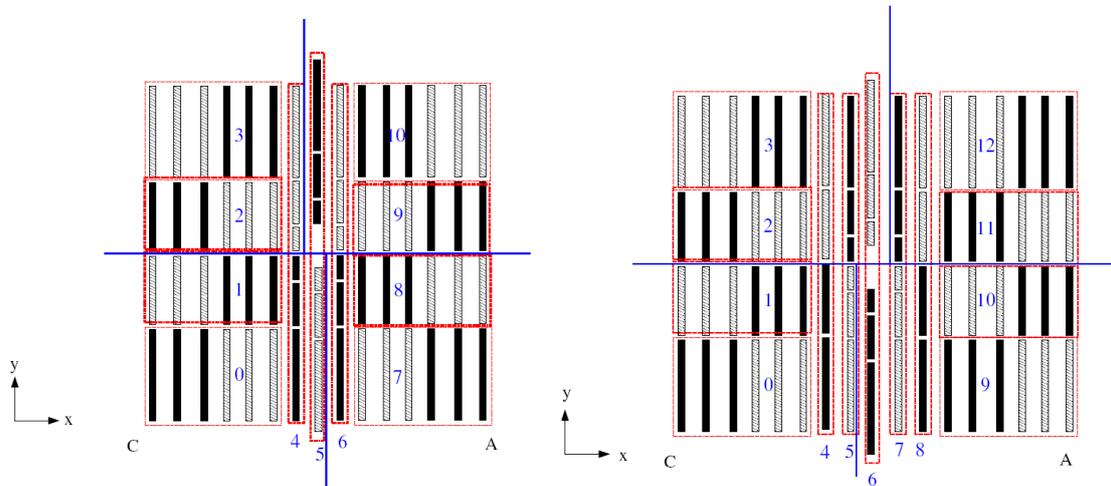
Figure 7.12: TT partitioning. TTa in the left figure and TTb in the right. Picture taken from [48].

Six service boxes for each of the four detector quadrants are fixed to the TT support structure on the left and right side of the station. The service boxes are arranged in four stacks located out of the acceptance and next to the magnet. The Figure 7.13 and Figure 7.14 show the mapping for the four quadrants. The white boxes in the SB drawing mean that there is an empty slot in the SB since no hybrid connects to it and this is because some partitions in the 'X2' and 'V' layers are made of 2 readout sectors.

Detailed pictures showing the hybrid to DB assignment can be found in Appendix M.
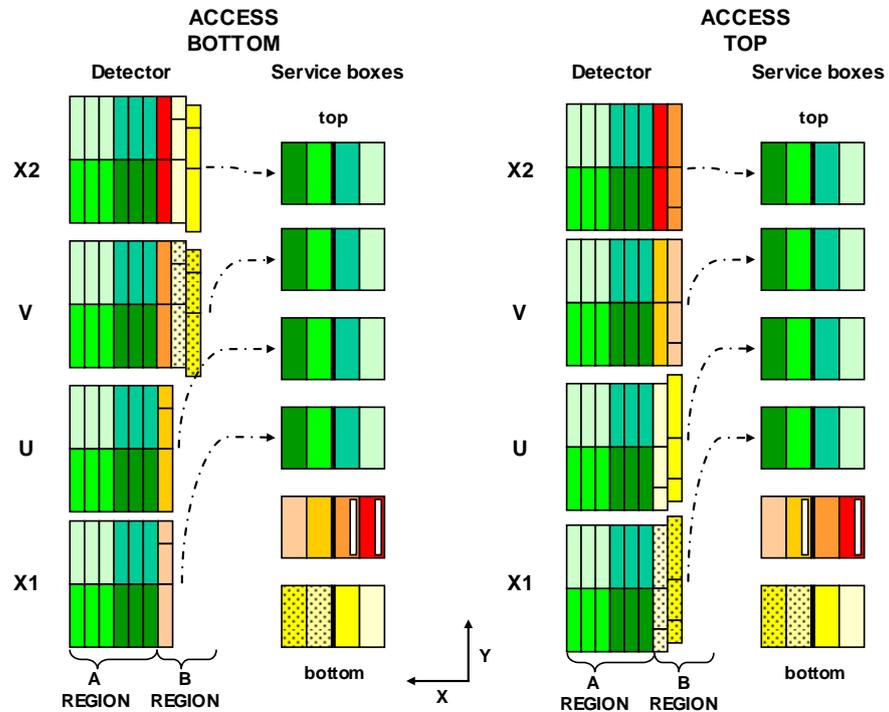
Figure 7.13: TT detector box to SB assignment for Access side.
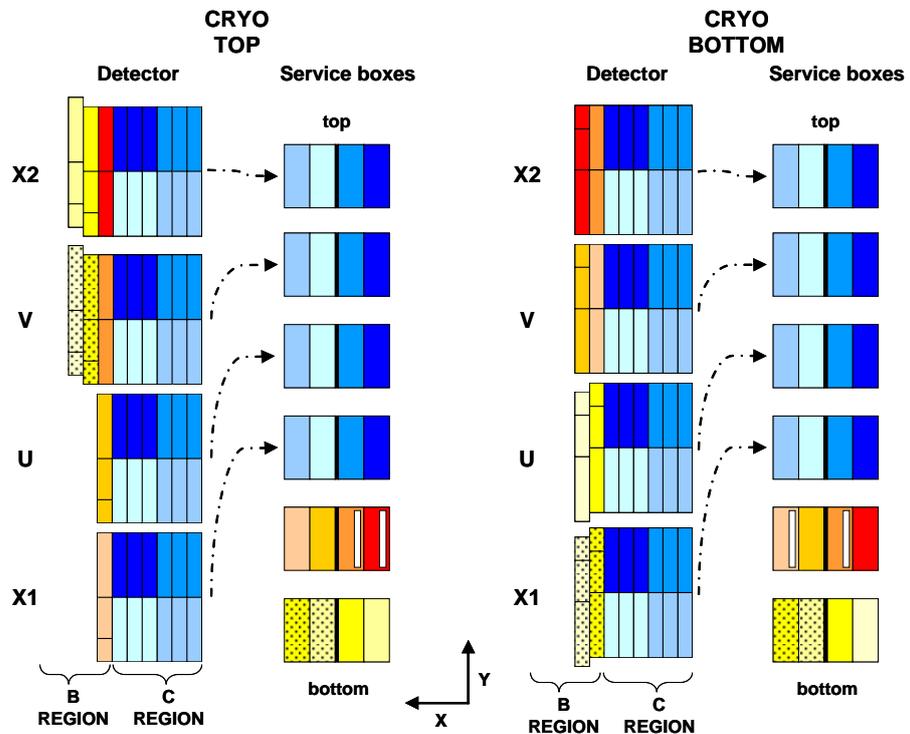
Figure 7.14: TT detector box to Svce box assignment for Cryo side.

7.4 DEVICE ADDRESSING

7.4.1 I^{2}C ADDRESSING

Due to the large amount of channels that the ST consists of, the number of readout chips in the Silicon Tracker reaches several thousands. To make feasible the control of these, the system electronics has been designed in a modular fashion taking into account the addressing needs.

All the FE devices are controlled using the I^{2}C protocol, which is driven by the SPECS system. The I^{2}C address identifier must be unique in the I^{2}C bus and one has to keep in mind the limitation on the chips space address (see Figure 7.15):

- The Beetle: all seven I^{2}C address bits can be used. Its programming scheme only occupies a single address.

- The GOL: occupies two addresses on the I^{2}C bus, being the first address associated to the register pointer and the second to the selected register data content [58].

- The TTCrx: uses two addresses in the same way as the GOL.

- The DCU: needs the three least significant bits to distinguish between its eight internal registers, which leaves only the address bits 'A6' to 'A3' for the user address space.

- The Delay25 chip: occupies eight addresses as the DCU.

Figure 7.15: I^{2}C devices space addressing.

In addition, there is a limitation on the I^{2}C space address given by the standard: two groups of eight addresses ("b'0000XXX" and "b'1111XXX") are reserved by the I^{2}C standard (see [59]) and must not be used. As a result, only the address range from '8' to '119' is free for use.

Keeping in mind these restrictions and the SPECS multi-bus I^{2}C capabilities an address space for the front-end electronics was defined and the hardware built accordingly.

On one hand, the CB, which is the same for IT and TT, delivers up to eigth external and long distance I^{2}C buses and an internal one. The external buses are dedicated for the hybrids and DB devices (Beetles, GOLs and DCU) while the internal one is for the CB inboard chips (TTCrx, Delay25 and DCU). The pictures 7.16 and 7.17 show the I^{2}C bus distribution.

On the other hand, each readout hybrid is associated to a DB, and there is a one to one matching between Beetle and GOL. The number of devices per hybrid/DB and the boards per partition are different for IT and TT: the IT hybrids and DBs house 3 Beetles/GOLs while in the TT are 4; and the number of boards comprising a partition is 4 in the IT and 3 in the TT.

From the previous, an I^{2}C addressing scheme for the long distance I^{2}C was defined, and the bit assignment in the addressing space reflects the chip and board grouping, as it is shown in Figure 7.18: 'BOARD-ID' identifies the board within a partition and ranges from 0 to 3; 'CHIP-ID' ranges from 0 to 3 and identifies the chip within the board.

The 'CHIP-ID' bits ('A2' and 'A1'), which are hardwired "in-board", correspond to the consecutive numbering of the Beetle chips on a given hybrid, or GOLs in a DB.

The 'BOARD-ID', ranging from 0 to 3, is set by the I^2C address bits 'A6' and 'A5'. All devices (Beetle, GOL, DCU) linked to a specific board will be "programmed" with the same 'BOARD-ID'. This is done with two common lines coming from the SB backplane, which are connected to all I^2C devices. The logic state of these lines is determined for each DB and its associated readout hybrid by the backplane slot, in which the DB has been placed. With this scheme, all DBs and hybrids could be identically manufactured without the need for manual addressing before installation and double address associations are avoided.

For the GOL/DCU I^2C bus, address bit 'A4' determines the device type, where '1' is associated with a GOL chip and '0' with a DCU. The DCU address bit 'A3' is hardwired to '1' for all DCU devices to avoid that any chip has the non permitted address '0000000' which is defined as the broadcast address in the I^2C "b'111111X", not allowed in the I^2C specification, and hence there is not any addressing conflict.

In addition, the Beetle address is a replica of the GOL address: associating bits 'A2' and 'A1' on the readout hybrid to the four Beetle readout chips creates identical addresses for each Beetle/GOL pair. No address conflict occurs since the I^2C buses are held separate. As the bit 'A0' is used for programming internal GOL registers, the matching of addresses cannot be extended to the full I^2C address. The remaining bit 'A0' of every Beetle is connected to '0' as no further addresses are needed for the Beetle I^2C bus.

The addressing of the CB internal devices poses not much complication since the number of chips in the bus is just three, and the assignment is shown in the Figure 7.18.

The I^2C buses distribution is depicted in the Figure 7.16 (IT) and Figure 7.17 (TT), which show a sketched frontal view of the SB. Every DB and associated hybrid plugs into a specific slot of the service box, and the partitioning of the slots is done as shown in the picture. The 'BOARD-ID' binary value assigned to all the devices plugged into a specific slot can immediately inferred from the slot position. The assignment of the eight external I^2C buses delivered by the CB is also shown, and the numbering used is the same as the internally used by the SPECS system. For a given partition, the I^2C bus that drives the communication with the hybrid is different from the one going to the associated DB. One of the reasons for separating the Beetles (hybrids) bus from the rest is limiting the number of devices attached to a certain bus, but there are other ones such as: having the same I^2C address for the pair Beetle/GOL, which makes the system more intuitive; it would allow for instant programming of all Beetles on a given I^2C bus by an I^2C general call; the I^2C bus speed is dominated by the bus capacitance, which in turn is given by the bus length (up to 10 m) for the hybrids, so having the DCU apart will permit for a higher communication rate with the GOL/DCU bus.

Figure 7.16: IT Service Box schematic view (frontal view) showing the slot numbering, the partition grouping, the I^2C bus assignment and the 'BOARD-ID'.

Figure 7.17: TT Service Box schematic view (frontal view) showing the slot numbering, the partition grouping, the I^2C bus assignment and the 'BOARD-ID'.

Long Multiplexed I2C bus addressing

A6	A5	A4	A3	A2	A1	A0
BOARD-ID	1	0	CHIP-ID	0

Beetle Chip I2C addressing

A6	A5	A4	A3	A2	A1	A0
BOARD-ID	1	0	CHIP-ID	X

GOL I2C addressing. 'X' bit is used internally

A6	A5	A4	A3	A2	A1	A0
BOARD-ID	0	1	X	X	X

DCU I2C addressing. 'X' bits are used internally

Control Board internal I2C bus addressing

A6	A5	A4	A3	A2	A1	A0
0	1	0	0	X	X	X

D25 chip I2C addressing. 'X' bits are used internally

A6	A5	A4	A3	A2	A1	A0
1	0	1	1	1	1	X

TTCrq I2C addressing. 'X' bits is used internally

A6	A5	A4	A3	A2	A1	A0
1	1	0	0	X	X	X

DCU chip I2C addressing. 'X' bits are used internally

Figure 7.18: Front-end addressing scheme.

7.4.2 SPECS ADDRESSING

The SPECS system is a master multi-slave protocol designed for the ECS of LHCb. It consists of a long distance connection between the master in the counting room and the slaves in the detector area. The slaves provide among other features an I^{2}C interface for the front-end electronics and five ADC channels for monitoring. The SPECS master is implemented in a PCI card which houses four masters and the slaves are implemented as mezzanine cards which plug into a mother board that in our case is the Control Board. Each SPECS master has its own ID which depends on the PCI slot where the card is plugged, having assigned each slot four IDs. The slave ID is defined by 5 interrupters of a micro-switch in the mezzanine. The SPECS slaves which assigned to a specific bus, that is, to a specific SPECS master, must all have different IDs so they can be unambiguously identified.

Each SPECS master provides a SPECS bus link between counting room and the SB location. The buses and hence master distribution has been done such that each group of neighboring service boxes are attached to the same bus. As plotted in the Figure 7.19, each group of four service boxes attached to an IT half-station are linked to the specs master shown in the picture. Figure 7.20 shows the assignment done in the TT, which just needs 4 four masters since the groups are made of the six SB for each of the TT quadrants.

The pictures also show the SPECS slaves IDs in the Control Boards for each SB. A CB houses two SPECS slaves, each with a different address. The SPECS slave identified in the CB as SPECS1 (the SPECS mezzanine plugged in the top) gets the lowest number (e.g., the ID 5) and the slave SPECS2 (the one in the bottom) gets the highest (e.g., ID 6).

CRYO SIDE   **ACCESS SIDE**

	4	3	2	1		1	2	3	4	
IT1C	7,8	5,6	3,4	1,2		1,2	3,4	5,6	7,8	IT1A
	SPECS master: 09					SPECS master: 13				
IT2C	7,8	5,6	3,4	1,2		1,2	3,4	5,6	7,8	IT2A
	SPECS master: 10					SPECS master: 14				
IT3C	7,8	5,6	3,4	1,2		1,2	3,4	5,6	7,8	IT3A
	SPECS master: 11					SPECS master: 15				
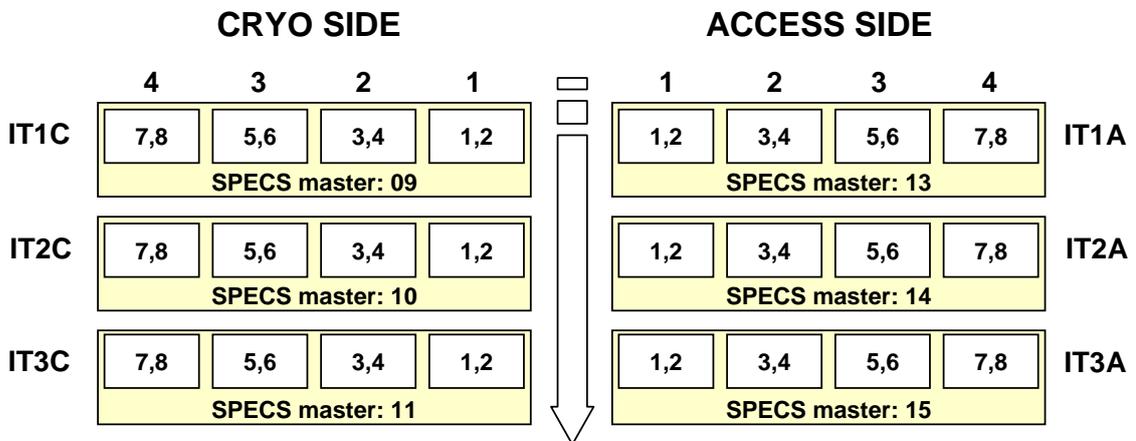
Figure 7.19: SPECS addresses assignment for IT service boxes.

	ACCESS TOP	ACCESS BOTTOM	CRYO TOP	CRYO BOTTOM
6	11,12	11,12	11,12	11,12
5	9,10	9,10	9,10	9,10
4	7,8	7,8	7,8	7,8
3	5,6	5,6	5,6	5,6
2	3,4	3,4	3,4	3,4
1	1,2	1,2	1,2	1,2
	SPECS master: 13	SPECS master: 14	SPECS master: 09	SPECS master: 10
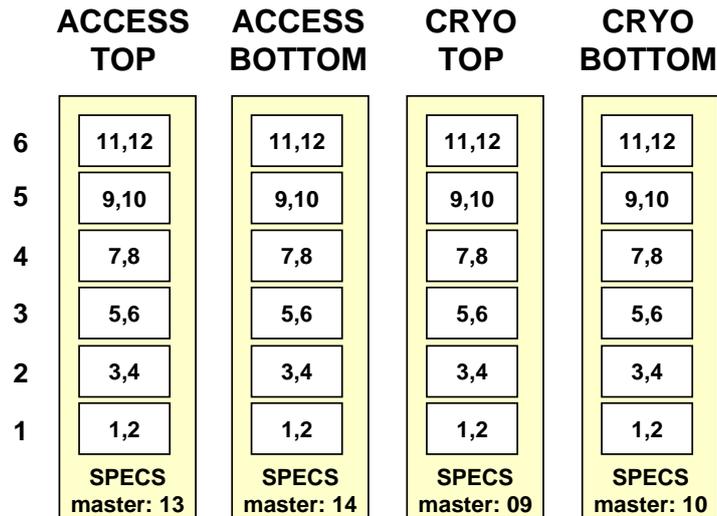
Figure 7.20: SPECS addresses for TT service boxes.

7.5   THE TTC SYNCHRONIZATION

The detector signals from the silicon sensors must be readout by the front-end electronics and further processed in the TELL1 such that the final data that arrives at the high level software processing units is consistent.

All FE Beetles must be perfectly synchronized to guarantee the coherent capture of the detector signals. The arrival of the *L0accept* to the FE must also be in perfect synchronization, to assure the extraction of correct event data from the pipeline. The *L0accept* is a constant 160 cycles latency trigger (~4 μs), so the detector signal is stored in an analogue pipeline buffer of the same length. Due to the relative high level 0 trigger accept rate, the data from the Level 0 pipeline is temporarily stored in the Beetle L0 derandomizer buffer before being transferred to the digitizing and L1 electronics. The size of this derandomizer buffer has been chosen to be 16 for performance reasons [57]. The L0 trigger-accept rate is limited centrally by the Readout Supervisor (RS), based on an emulation of the L0 derandomizer buffers. For such a scheme to work reliable for a large system, the L0 pipelines and the L0 derandomizers work in perfect synchronization.

The LHCb TFC system [60] provides a common Timing and Trigger Control to all front ends. The TFC system network distributes all time critical signals (the clock and fast control signaling) to the L0 and L1 DAQ electronics. The Readout Supervisor that receives trigger decisions from the trigger systems, drives the TTC system. The RS is the responsible for preventing buffer overflows in the FE and DAQ systems. During testing, debugging and special calibrations each sub-detector is driven from a separate RS via the partitioning system of the TFC system. During normal physics running all sub-detectors are driven from one common central RS.

The TFC system controls the entire readout of the LHCb detector from the FE electronics up to the distribution of events to the processing farm. The ECS, DAQ, the TFC and other network and hardware resources such as the data storage and data online monitoring systems compose what is known as the LHCb Online system.

7.5.1   PHASE ALIGNMENT TO BUNCH COLLISIONS

The phase of the clock used to sample the detector signal at the Beetles input must be precisely aligned with the bunch collisions in the interaction point taking into account the particle time of flight and the delay in the detector and the analogue front-end. The sampling of the detector signal must be performed at the peak of the signals generated by the traversing particles. Because of the high bunch crossing rate of the LHC, and the requirement to limit signal pile-up and spill-over to a minimum, the signals originated in the front-end normally should have peaking times (and pulse width) not much higher than the 25 ns bunch crossing interval (Figure 7.21). This condition can be relaxed a bit by the fact that the detector occupancy is below (1%) and the probability to get to consecutive particles in the same readout strip is very low.

The sampling time must therefore be controlled with high precision and stability (~ns). This is done using the *Clock40Des1* from the TTCrx which has a resolution of ~104 ps. The correct phase alignment can be

obtained during special calibration runs where the clock phase is adjusted until the correct capture of real detector signals are obtained.

The clock phase alignment is performed for the group of modules that plug to the same SB. The TFC signals for each SB are provided by one CB and redistributed to the different modules (hybrids and DB) via backplane. A strict matching of delays of all elements was taken in the design of the backplane to keep the signals skew to a minimum. In addition, all the cables that plug to the same SB have identical length and so the signals propagation delay.

The source clock for all the hybrids in the ST is the *Clock40Des1*: this guarantees that all the Beetles (and thus the L0 pipelines) run synchronously and that the L0 trigger arrives at the Beetle front end in perfect phase. The latter comes from the fact that the TTCrx generates the signal *L0accept* synchronized to the *Clock40Des1* and the relative phase between both can be configured in the Delay25 chip.
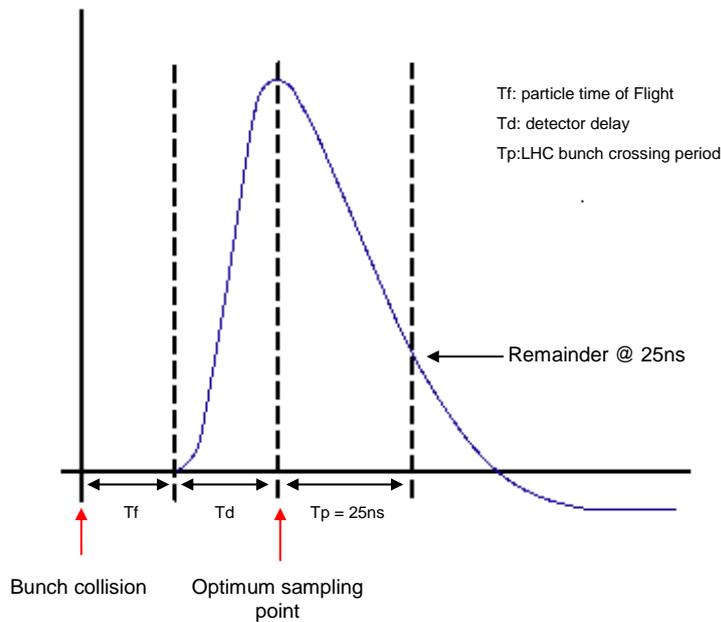
Figure 7.21: Detector signal phase alignment.

7.5.2   SYNCHRONIZATION TO LHC BUNCH STRUCTURE AND L0 TRIGGER ACCEPT

After phase alignment of the individual detector signals and synchronization to a common clock all data in the FE are available in a sampled format at the rate of the LHC bunch crossings. The alignment with the LHC bunch structure must be performed to assign correct bunch crossing identification to the acquired data. The absolute value of the bunch identification (B-ID) is not of any real significance in the ST as most B-ID use is to perform relative comparisons between different B-ID counters to verify that the local systems or local front-end units are synchronized. Nevertheless it is crucial to have the detector time aligned with respect to the right bunch to get synchronous data with respect to the other detectors.

A coarse alignment, in number of clock cycles, is performed by the TTCrx, introducing a delay of up to 15 clock cycles on the TTC signals (L0 trigger, resets, short broadcast, etc.). The limited range of this alignment requires the complete FE system to have a maximum skew of 400 ns including: particle flight time, detector response, analogue front-end, input to L0 pipeline, TTC distribution and the final extraction of data from L0 buffer. A worst case scenario in LHCb would imply that the detector delay (from the sensor to the input of the L0 pipeline) should be kept below 200 ns [57], requirement that the ST fulfills by far since the detector response is well kept within the range of one-two clock cycles.

The assignment of correct bunch crossing ID to event data is realized after an event is accepted at the TELL1 level. Although the TTCrx implements a bunch ID counter which is only available at its output when a *L0accept* has been generated, the TELL1 implements a bunch ID counter which is available all the time. The correct bunch ID of data at the input to the L0 pipeline can be calculated from this by adding the L0 latency (160) taking into account that the Bunch ID does not return to zero after 4095 (normal binary roll-over) but at 3563 (LHC machine cycle length = 3564).

7.5.3 SYNCHRONIZATION OF BEETLE SIGNALS

The TTC signals delivered by the CB to the front-end hybrid pass through the Delay25 chip. This chip allows the phase alignment of the *L0accept*, the *L0reset* and the *calib0* with respect to the 40 MHz clock (*Clock40Des1*) in order to avoid setup and hold timing violations at the Beetle inputs due to the skew of signal distribution in the whole chain from the TTCrx to the Beetle.

7.5.4 SYNCHRONIZATION OF BEETLE DATA DIGITIZATION

The data read out from the Beetles arrive at the DBs with a minimal skew for all the links belonging to the same service box. The *Clock40Des2* is used by the ADCs as the sampling clock, so its phase with respect to the incoming data from the Beetles must be precisely tuned in order to sample with the optimum timing. The optimal sampling point should usually be in the peak of the signal, but a precise tuning should imply searching not for the peak of the signal but for the maximum of the S/N, just in case any systematic noise exists somewhere in the FE or in the electrical path from the Beetle output to the ADC input.

The synchronization between Beetle digitized data and GOL transmission is done by means of the digital signal called *DataValid* that the Beetle generates to indicate the presence of analogue readout data. This signal is manipulated in the DB for accommodating its phase to the delays of the ADCs and for reducing the data length from 36 to 35 clock cycles (875 ns) and therefore guaranty always an empty clock cycle for consecutive readout. This empty clock cycle is of crucial importance since the links synchronization at the TELL1s relies on the fact that the incoming FE event data size is fixed to 35 cycles.
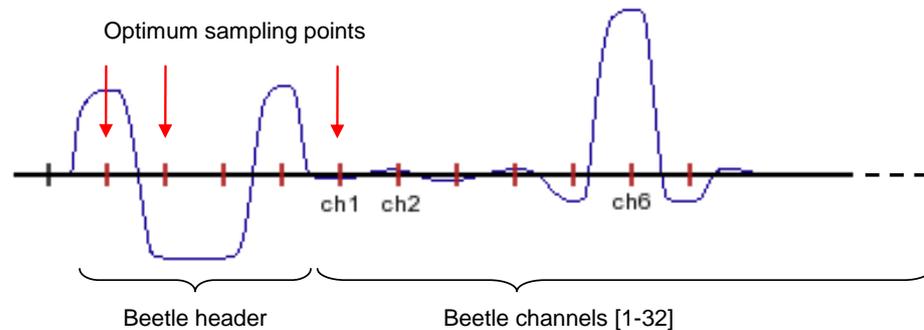
Figure 7.22: ADC optimum sampling points marked in red in the ordinate axis.

7.5.5 SYNCHRONIZATION VERIFICATION ON THE TELL1 ELECTRONICS

The TELL1 checks that the event data fragments sent by the GOL to the optical link receiver always contain 35 data words. This permits maintaining correct event synchronization in case of a corrupted event (e.g. bit error on optical link) since the link idle character is always inserted between any two consecutive events. Failure to disentangle two consecutive events causes a desynchronization at the TELL1 level that may end up in corrupted events and consequently an unpredictable behavior at higher levels in the DAQ chain.

The 35 word event fragment consists of 32 detector channel samples and three data tags. One of the data tags contains the ADC value of one bit of the Beetle Pipeline Column Number which is used to check the correct synchronization of all Beetles read-out by the same TELL1 board.

In order to assure a robust synchronization between the FE and the TELL1 an algorithm was develop at the first processing stage of *PP-FPGA*. This has already been reported in the "Synchronization" paragraph of section 7.2.6.

7.5.6 *L0RESET* SYNCHRONIZATION

After a reset of the L0 front-end electronics (*L0reset* via short TTC broadcast) the data synchronization stages must re-synchronize themselves with new data streams. When the *L0reset* is issued the RS will normally ensure that event data from correctly working L0 front-ends are not truncated (but this may not be the case for L0 front-ends which have suffered a SEU or other mal-function). No valid event data will arrive before 160 clock cycles after the *L0reset* and the first new valid event fragment will have the L0-Event-ID set to zero.

7.5.7   CALIBRATION PULSE INJECTION WITH TIMING ADJUST

To make the calibration pulse injection useful for real system tests it is possible to adjust the timing of the pulse injection in addition to the general time adjustment made in the TTCrx affecting all TTC signals. The timing adjustment of the calibration pulse (*calib0*), generated after decoding the broadcast command, is external to the TTCrx: a coarse delay covering at least 16 cycles can be performed in the SPECS decoder and a fine phase delay in steps of 0.5 ns is possible in the Delay25 chip.

The timing from the received calibration pulse injection command (seen at the output of the TTCrx) to the corresponding trigger accept (seen at the TTCrx output) has been fixed at 160 +16 = 176 clock cycles [57].

The nominal latency of the L0 pipeline in LHCb has been chosen to 160 cycles and that is what we configure in the Beetle chips. If for any reason this latency is modified there is a side effect in the calibration signal timings. This has to be handled locally since timing between calibration pulse injection command and corresponding trigger will stay fixed at 160+16 clock cycles.

The Figure 7.23 shows a use of the calibration signal for testing the IT ladder sensors. It consists of pulsing the 384 sensor channels looking for possible open (not connected to the Beetle inputs) or shorted (neighboring strips connected) strips. The picture shows highlighted in red the strips 94 and 95 as potentially shorted. These strips show a smaller peaking height compared to the others due to the higher combined capacitance of both strips. The left hand of the picture shows the superimposed pulse-shape timing scan for the 384 channels of a ladder, and the highlighted area corresponds to the low peaking of the shorted channels. The right side of the picture shows the channel peaking of all channels when sampling in the optimum point.
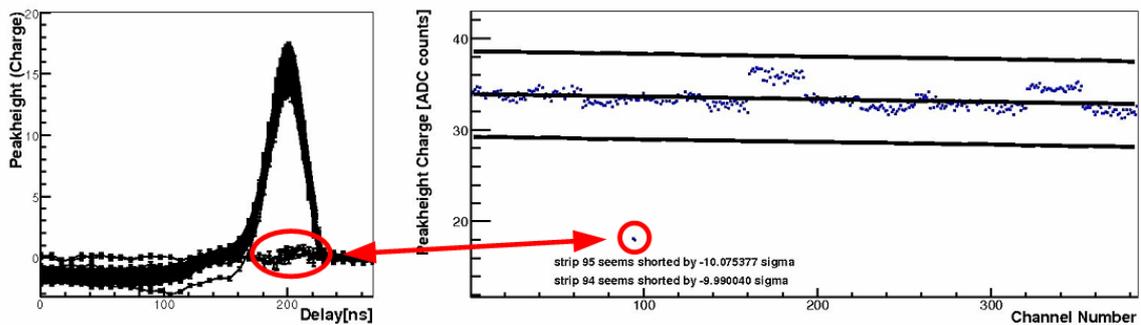
Figure 7.23: Calibration Pulse injection picture test results in an IT ladder. Highlighted in red are two channels that are shorted together and have been spotted with the test-pulse signal.

7.6   THE DAQ ELECTRONICS CONFIGURATION

See appendix P.

CHAPTER 8

INTRODUCTION TO THE ECS IN LHCB

ABSTRACT

This chapter provides an overall technical description of the ECS in the LHCb experiment. The aim of the ECS is presented first. Follows a brief explanation of how common approach to develop the control systems of all LHC experiment was taken and the tools that were evaluated. PVSS and the framework built around it are also described and finally a specific description of the most relevant Online System elements is given.

"La vida es aquello que te va sucediendo mientras estás ocupado haciendo otros planes"

John Lennon

8.1 OVERVIEW

The LHCb's Experiment Control System (ECS) is one of the three key parts of the LHCb Online system (see section 8.8).

LHCb's ECS is in charge of the configuration, control and monitoring of all the components of the online system. This includes all the devices in the areas of: data acquisition, detector control (ex slow controls), trigger, timing and the interaction with the outside world. The ECS is a highly hierarchical system with a large set of local controllers performing the control and monitoring of specific sub-systems.
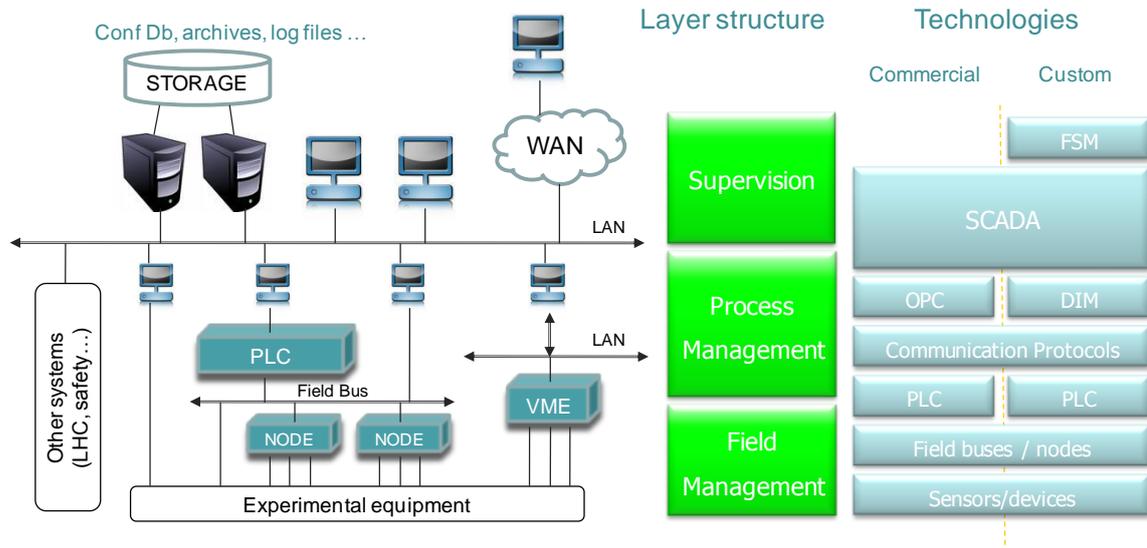
Figure 8.1: LHCb ECS hardware and software layered view architecture

From the ST point of view, the main task of the ECS is to enable the proper and safe operation of the LHCb ST detector and provide a link to the general LHCb ECS. The tasks that the ECS has to fulfill can be summarized as follows: it has to be able to control, configure and monitor the status of the detector electronics, temperatures, humidity…; it has to implement enough functionalities to permit performing calibration runs on the detector; it has to be capable of performing exhaustive tests of individual modules; it will have to be able to interface to external systems such as databases, the DSS system, the LHCb higher level control units, the TFC system…

The ECS has the ability to control the system in different modes and perform automated procedures in case needed. For example, it will automatically stop the data acquisition when the high voltages report an error. Although the ECS will operate the experiment as a whole during physics data taking, it provides the functionality of operating each detector as an individual unit. This will usually be the case while testing, calibration, commissioning periods.

The ECS has to be as automated as possible in order to avoid potential operator mistakes. It is foreseen that the grade of automation in the system will be increasingly growing as the detectors behavior is better known and understood. If automation is not possible, the system has to be intuitive and easy to handle.

The way to build a common, consistent and integrated control system for the whole experiment has been based on the use of the same tools and devices by all the ECS development teams [61] [62]:

- The software tools and protocols used by the development teams are the same for everybody: LHCb makes extensive use of the JCOP tools [63] apart from some specific LHCb specific ones, and everybody uses the same physical and software protocols for communication with devices and between software components. One of the major agreement points is the use of the PVSS SCADA [64] by all LHC experiments.

- A common architecture and set of guidelines for building the hierarchical distributed control system of each sub-detector were defined since the starting stages. A common 'look and feel' for the user graphical user interfaces has been adopted, which enormously helps the operator. Also the way of operating the system presents common features to everybody, such as the partitioning modes and naming conventions.

- Common hardware developments such as the SPECS and systems such as the Power Supplies have commonly been adopted by all sub-detectors. In addition, the network infrastructure is common to every sub-detector and sub-system and isolated from the outside world.

The physical connection to electronic modules in HEP experiments has traditionally been the backplane bus of the crates (CAMAC, Fastbus, VME) using simple register mapping or message passing. This approach was withdrawn and COTS (Commercial Of The Shelf) components were introduced as much as possible. At the lowest level of the system the connection to the electronic modules had been standardized in LHCb to three different types:

- Credit card PC (CCPC): a miniature PC on a commercial credit card sized hybrid is mounted on the front-end (FE) module and interfaces to the higher levels of the ECS system over 10/100 Mbit Ethernet.
- SPECS: SPECS is a simple custom protocol made specifically for applications in high energy physics. It consists of a single master located in a standard PC communicating with multiple slaves place at distances of up to 100 m at a data rate of 10Mbits/s.
- CAN (ELMB): A CAN slave interface based on a self checking master slave configuration.

In the ST the SPECS system was chosen for interfacing the FE electronics as it was found more flexible and easier to adapt to the system needs both in terms of radiation tolerance and I/O control lines. The CCPC are using for controlling complex electronic boards such as the L1 and TFC boards.

8.2 CONTROL SYSTEMS AT CERN

8.2.1 BRIEF HISTORY OF CONTROL SYSTEMS

Automatic control is a discipline that is approximately eighty years old. During this period both technological development and the formulation of the modern general control theory resulted in an impressive evolution of control systems. Developments in various fields of engineering entailed to the development of very sophisticated machines, devices and manufacturing processes. Successful operation of these machines, devices and processes requires very short response time, large amount of complex, repetitive analytical and mechanical operations, and low tolerance to errors that are well beyond human abilities. Automation became the only alternative to continue the technical progress. The design of a particular automatic control system constitutes on one hand an electrical and/or mechanical engineering problem and in the other hand a control process engineering challenge.

Introduction of computers implied the beginning of a new era in control engineering: powerful mathematical tools could be developed; computers allowed for numerical simulation of control and dynamic systems; the processors became a part of a control system, implementing in software the most sophisticated control laws. This latter point is the most exploited aspect in the LHCb experiment, since the level of automation required can be considered to be very low no models of the system where developed and hence no automation control theory applied.

The application of computers in the process control meant an enormous technological step forward involving the introduction of new systems of control in the industry and made possible the development of things such as the space navigation. The use of microprocessors soon went beyond using them just for simple regulation as their continuous incrementing calculus power allowed the implantation of much more complex advanced algorithms of control. The final aim was scaling the use of the computers to the level of realizing an integral control of the manufacturing plants, including also the production management.

Nowadays, the main uses of computers in the process control are:

- Acquisition of information: It covers gathering, treatment and storage of the information.
- Supervision: The computers provide summarized information such as alarms, failure treatment and recovery procedures. It is not the computer itself who controls the process but other controllers (PLCs, regulators PID …), and the computer connects to them by means of any kind of communications bus or network. The principal function is to help the operator of the plant (floor).
- Control: The computer is used to directly control the process by simple sequential control, direct digital control, distributed control…
- Analysis of data: The information of production is analyzed using office or custom developed tools.

The advantages of using the computers in the process control are multiple, but we can remark the major efficiency of the operations, major safety and a drastic reduction of the manual operations.

However, the introduction of computers and microprocessors also brought some drawbacks. At the beginning, the specific nature of the tasks to be controlled called for instruments and control methods to be custom designed, and later on, in the 1980s transductors began to integrate digital control. These smart instruments had to be integrated into the field networks and, consequently, a heterogeneous variety of privative and open field bus standards and protocols were developed. This variety of hardware and software interfaces to connect to hardware supposed a problem in high scale systems since the control computers had to handle many types of communication protocols using dedicated interface boards and software.

During the 1990s the Supervisory Control And Data Acquisition (SCADA) systems emerged following the introduction of the personal computer. These systems generally implement software interfaces or drivers that provide support to many types of field bus protocol. SCADASs have evolved allowing full distributed control facilities using the Internet more as a communication tool.

8.2.2   CONTROL SYSTEMS EVOLUTION AT CERN

Traditionally, at CERN, teams on each experiment (in some cases each sub-detector) have independently developed their Experiment Control System (ECS), also known as "slow controls". At the LEP experiment, this was still the case. The new generation of detectors for the LHC experiments brought new challenges on terms of ECS due to their higher complexity and size: the number and geographical distribution of development teams (~50 sub-detectors in LHC), the size and complexity of the control systems, limited resources, the long lifetime (20 years) and the perceived similarity between the required systems. A change in philosophy was needed and CERN and the experiment's management jointly decided to develop, as much as possible, a common ECS for the LHC experiments. In 1997 the Joint Controls Project (JCOP) was set up as a collaboration between the controls teams on the LHC experiments and the support groups in CERN's information technology and physics departments.

A lot of experience was gained in the LEP experiments and an important lesson was learnt [65]: the use of industrial component technology reduces the development and system maintenance workload and time. In LEP experiments there was a lack of standardization in many areas and custom hardware and software was developed for most of the systems. Development and maintenance was hard, and upgrades or plant extensions were expensive in terms of money and manpower.

The JCOP team undertook evaluations to assess the suitability of a number of technologies, primarily commercial ones, such as OLE for Process Control (OPC) [66], the field buses 'CANBus' and 'ProfiBus', commercial programmable logic controllers (PLCs), as well as supervisory control and data acquisition (SCADA) products. The PVSS (Prozessvisualisierungs und Steuerungs System) SCADA product was eventually selected as the basic building tool for the LHC experiment control systems. In addition, and where suitable commercial solutions were not available, products developed at CERN were also evaluated. This led to JCOP's adoption and support of CERN's Distributed Information Manager (DIM) [67] middleware system and the SMI++ finite-state machine (FSM) toolkit [68]. Furthermore, developments made in one experiment were also adopted by other experiments.

PLCs are effective at performing autonomous and secure local process control, and fieldbuses are an ideal solution in a geographically dispersed environment. The advantage of using a standard bus relates to its ease of use having no drivers to write and maintain. If the fieldbus is then connected with a SCADA product or to a PLC component no communications software have to be written. OLE (Object Linking and Embedding) for Process Control (OPC) is an emerging non-proprietary software standard used to interface components from different vendors which offers several advantages: First, it acts as glue between independent layers of the system allowing to change an individual layer without breaking the entire system; Second, it permits separately developed components to be readily incorporated into the overall DCS. Regardless the advantages offered by OPC, initially it was developed as a platform-dependent protocol running on Windows. Thus, in order to complement OPC, the Data Interchange Protocol (DIP) was adopted as the standard solution for the DCS exchange information with external systems (e.g. LHC machine, TDAQ, CERN Technical Services). DIP is based on the Distributed Information Management (DIM) protocol already used at the DELPHI experiment during the nineties. This is a suited solution for exchanging information between heterogeneous systems running in different platforms.

One major challenge has been the development of the so-called JCOP framework [63]. It provides a customized layer on top of the technologies chosen, such as PVSS, SMI++ and DIM. It offers many ready-to-use components for the control and monitoring of standard devices in the experiments (e.g. CAEN high

voltage, Wiener and CAEN low voltage…). The FW (framework) also extends the functionality of the underlying tools, such as the configuration database tool and installation tool.

As well as selecting, developing and supporting tools to ease the development of the DCSs, there have been two areas where complete applications have been developed. These are the detector safety systems (DSS) and the gas control systems (GCS). The DSS, which is based on redundant 'Siemens' PLCs and on PVSS, has been developed in a data-driven manner that allows all four LHC experiments to configure it to their individual needs.

Some issues of scaling and performance have emerged as the systems have increased in size, with a bigger fraction of the detectors being commissioned. However, thanks to the JCOP collaboration, it has been possible to address these issues in common for all experiments.

Due to the scale of the LHC systems (in the order of several millions channels) and to guarantee the operating independence of the experiment sub-detectors, the ECS of the LHC experiments needs to be distributed across many machines. It has been decided that the best approach to perform the control of such systems is to build a supervised distributed hierarchical control system in each of the experiments.

8.3   SCADA System: PVSS

SCADA systems are commercial software packages extensively used in industry for automated and real-time process control. A SCADA is not a full control system but rather focuses in the supervisory level. A SCADA provides a way to connect both to hardware (or software) devices and to other users (local and business networks) in a completely transparent manner for the user. PVSS, as any other SCADA, is able to acquire information from the devices that can be used for supervision, i.e. to monitor their behavior and to initialize, to configure and operate them. To do so, it provides:

- A run time database where the data coming from the devices is stored, and can be accessed for processing, visualization, etc. purposes.
- Archiving: Data in the run-time database can be archived for long term storage, and retrieved later by user interfaces or other processes.
- Alarm Generation & Handling: Alarms can be generated by defining conditions applying to new data arriving to PVSS. The alarms are stored in an alarm database and can be selectively displayed by an Alarm display. Alarms can also be filtered, summarized, etc.
- A Graphical Editor: Allowing users to design and implement their own user interfaces.
- A Scripting Language: Allows users to interact with the data stored in the database, either from a user interface or from a "background" process. PVSS scripts are called CTRL (read control) scripts.
- A Graphical Parameterization tool, allowing users to: define the structure of the database, define which data should be archived, define which data, if any, coming from a device should generate alarms, etc.

8.3.1   PVSS Architecture

PVSS presents an open flexible distributed architecture that eases the development of custom designs for specific application areas. The PVSS internal architecture is highly modular and it is implemented as a set of different functional modules that perform different specific tasks. These modules are called managers and run as separate software processes. Figure 8.2 shows a typical configuration example of a PVSS system. The communication between the individual managers is based on TCP.

The most remarkable managers are:

- Drivers (D): it is the PVSS system lowest level. They provide the interface to the devices to be controlled. These can be PVSS provided drivers like Profibus, OPC, CANbus, etc. or user-made drivers.
- The Event Manager (EV): it is responsible for all internal communications. It receives data from drivers (D) and sends it to the Database Manager (DB) in the database. The EV maintains the process image of all current data values in memory and ensures that the data is distributed to all the managers subscribed to it. It is also the responsible for the alarm generation and handling.
- The Database Manager (DB): provides the interface to the (run-time) database.
- User Interface Managers (UI): they correspond to the highest level of abstraction and they provide the interface with the user. These include a graphical editor (GEDI), a database editor (PARA

module) and general application user interface. In the UI, values are displayed, commands issued and alerts tracked. Trends and reports are also usually included in the UI. In PVSS, the UI interaction software runs concurrently to the rest of the processing running in "background", the UI just provides a window to the process image and to the historical archived data.

- Ctrl Managers (CTRL): provide for any data processing as "background" processes, by running a scripting language. This language is like "C" with extensions. This is a powerful tool as it permits multi-thread programming as well as provides a large library plenty of useful functions for data and process handling.

- API Managers (API): allow users to write their own programs in C++ using a PVSS API (Application Programming Interface) to access the data in the database. This is the most powerful and flexible way to add extra functionalities to PVSS such as adding self-contained managers.

- Distribution Manager (DIST): used to build PVSS distributed systems. They provide the interface between systems connecting them together. Hundreds of systems can be connected in this way.
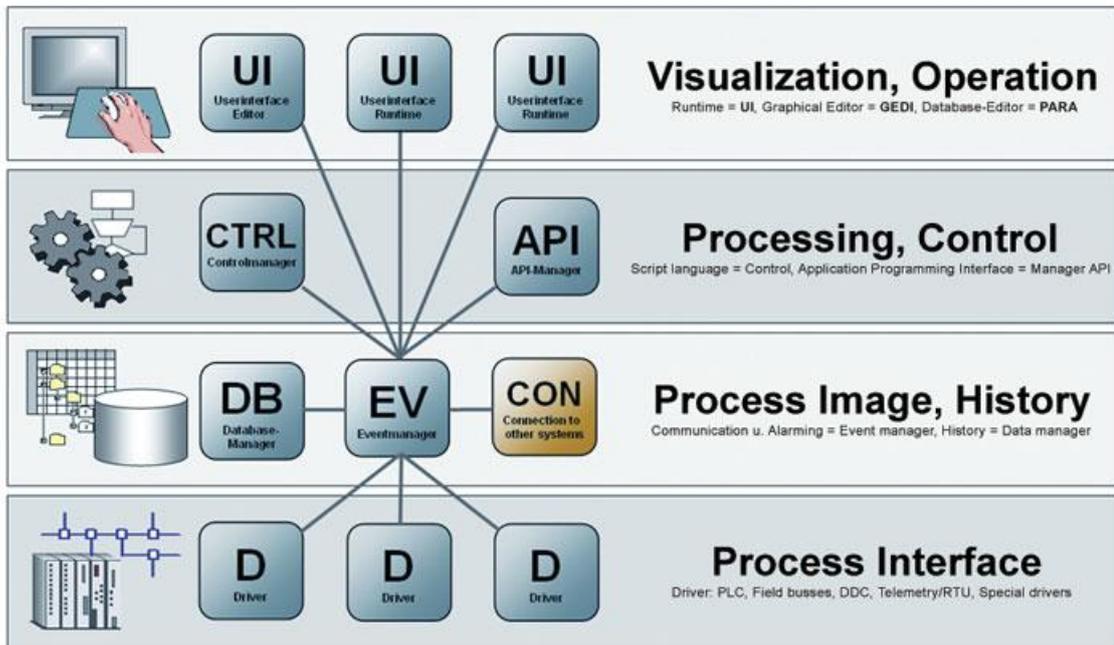
Figure 8.2: PVSS modular architecture. Picture taken from [64].

8.3.2 SYSTEMS AND DISTRIBUTION

The definition of a PVSS System is an application containing one database manager and one event manager and any number of drivers, user interfaces, etc. but as a minimum an event and a database manager must exist.

PVSS is based in the client-server architecture that allows practically unlimited scaling of the system. PVSS is used from small single-site systems to distributed, redundant multi-site systems in a wide range of configurations. Thanks to this, functions and loads can be distributed across multiple computers.

PVSS Managers can run on 'Windows' or 'Linux' and they can all run in the same machine or be distributed across different machines (including mixed 'Windows' and 'Linux' environments). When the managers of a single system run distributed across different machines this is called a PVSS Scattered System (Figure 8.3).

PVSS can provide for very large applications. In the case one PVSS system is not enough, a PVSS Distributed System can be used (Figure 8.4). A distribution manager must be added to each of the systems and then they must be configured to connect to each other. In a hierarchical structure, the best solution to set up a distributed system is that the top-level system acts as a client for all other systems, the bottom-level systems act as servers to all other systems and the medium-level systems act as clients for all systems below and servers to all system above. This clarification about hierarchical structures will make sense when the approach for the LHCb ECS architecture is presented.
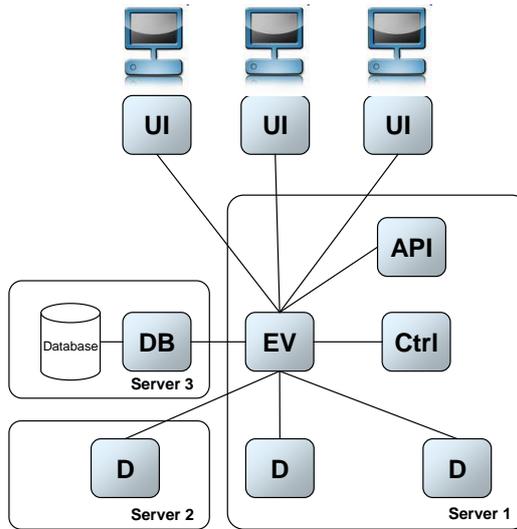
Figure 8.3: PVSS scattered system.
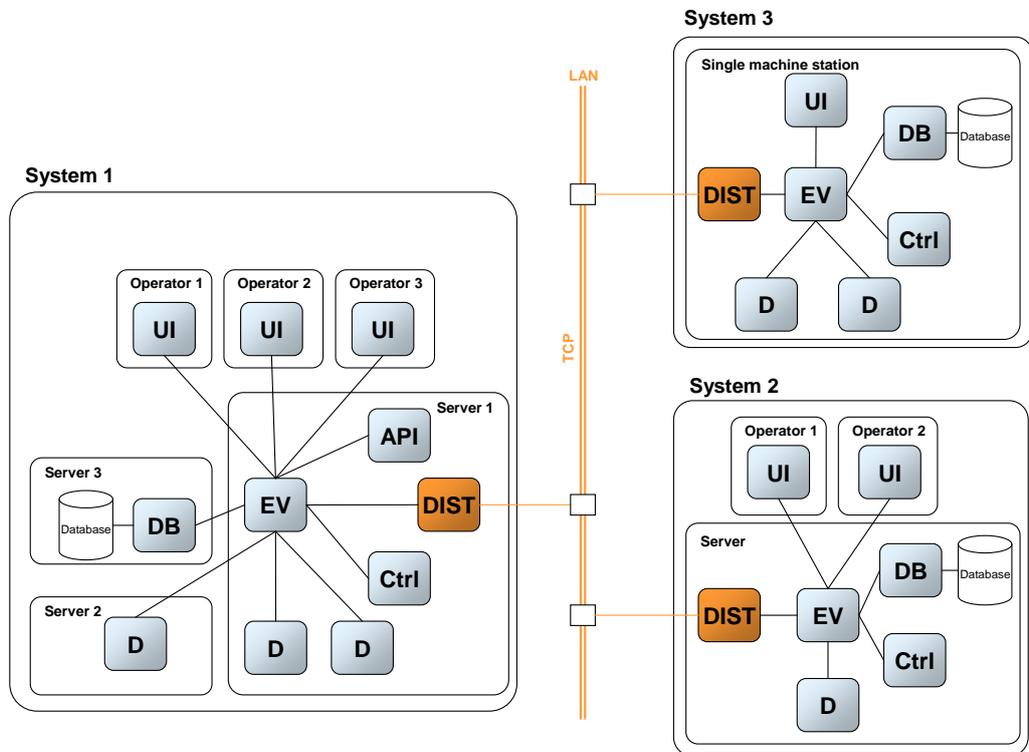
Figure 8.4: PVSS distributed system.

8.3.3   THE *DATAPOINT* CONCEPT - DATABASE STRUCTURE

The variables of the process to be controlled and monitored must also find their way into the software at the control desk. Every logic state, every measured value or set-point value must correspond to a sort of variable that represents this value within the system. These variables of the process image are called *datapoints* (DP) in PVSS II. Many different names are used for these information carriers depending on the product or region (tag, process variable, item, point, I/O point, etc.). The advantage of PVSS with respect to other SCADAs is that instead of just allowing the instantiation of single primitive type *datapoints* it allows to create complex tree structured *datapoints* types. This is a clear advantage as this permits creating user definable types that represent complex devices with many functional parameters.

A *DataPoint Type* (DPT) describes the data structure of the device and a DP (*Datapoint*) is a particular instance of the DPT that contains the information related to a device. In the example Figure 8.5 we see a DPT representing a simple high voltage channel. The DPT has a set of read (readings) and write (settings)

parameters, called *DataPoint Elements* (DPEs). After the DPT is defined, the user can then create datapoints of that type which will hold the data of each particular device. The user can define as many DPTs as desired to model all the types of devices to control.

As already mentioned, PVSS provides a scripting language and libraries that allows the users to interact with the data, more precisely, the *datapoints* and *datapoint elements*. This is of great importance as it will permit having "background" scripts operating and performing the needed tasks on the DPEs, and thus will permit extending the PVSS capabilities to a great extent.
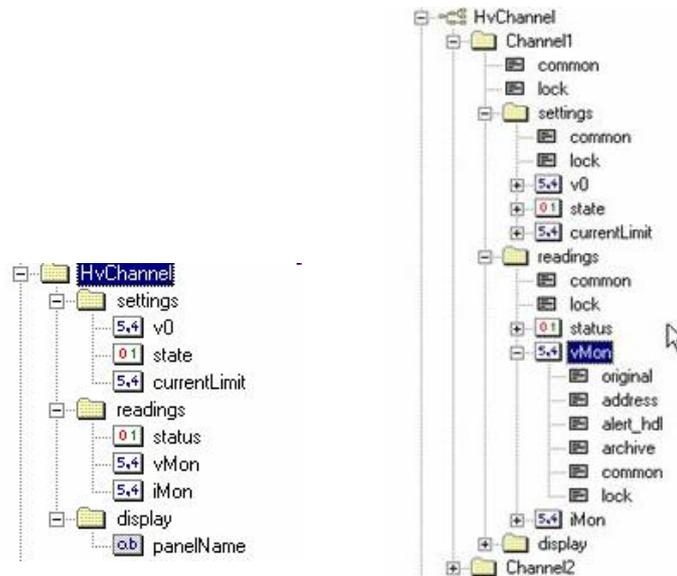
Figure 8.5: Datapoint *type and* datapoint.

8.3.4 THE CONTROL SCRIPTS

PVSS Ctrl Scripts can be used in panels or as stand-alone processes. They provide the interface to the PVSS database. These scripts comply with the "C" syntax but they contain extensions, in particular new variable types, like *string*, and the provision for dynamic arrays.

PVSS provides a very large library of functions giving access to all PVSS functionality and more. There are functions for *datapoint* manipulation, for graphics, for file access, etc. Also any user functions that are repeatedly needed can be stored in libraries to be used by panels and scripts.

8.3.5 USER INTERFACES

PVSS allows users to design their own user interfaces, in a "drag and drop" fashion. By using the Graphic Editor (Figure 8.6), the user can design panels, by placing widgets like buttons, tables, plots (called trends in PVSS), etc. Actions can then be attached to each widget. Depending on the widget type actions can be triggered on Initialization, user click or double click, text input, etc.

Actions are programmed using PVSS's scripting language, i.e. by means of CTRL scripts.

8.4 THE LHCB FRAMEWORK

The LHCb framework is meant to provide all the necessary components to build the control software of the LHCb experiment. The control framework is based on the SCADA PVSSII and the JCOP framework [63] as cornerstones. In addition some other specific tools have been built for the LHCb experiment.

The LHCb framework is not only composed of a set of tools but also by a set of guidelines to help in the development and integration of the several sub-detector control systems of the LHCb experiment.

Although the JCOP framework primarily consists of a set of components developed based on PVSS such as libraries, custom PVSS framework managers, graphical user interface panels and other things, it also includes external components as relevant as SMI++ and DIM. SMI++ is a framework for designing and implementing complex distributed control systems based on the concept of objects behaving as Finite State Machines (FSM). SMI++ objects can run in a variety of platforms since all communications are handled transparently

by the underlying communication system known as DIM. DIM is a communication system for distributed/mixed environments that provides a network transparent inter-process communication layer. The JCOP FW provides panels and libraries for building hierarchical control systems based on the SMI++ and DIM in completely transparent way and this will described aside in section 8.5.

Other specific framework components have been built trying to fulfill the LHCb needs related to the LCHb specific and widely used hardware interfaces SPECS and CCPC. On top of these two, a specific tool called 'Hardware tool' was built. The 'Hardware tool' requires a specific section in this chapter as it is of special relevance in this theses. The framework also consists of a set of components that were constructed to integrate into the ECS sub-systems such as the TFC (Timing and Fast Control) and the HLT (High Level Trigger) farms, and to ease the implementation of the controls hierarchy by the sub-detectors.

Hereafter the JCOP framework, LHCb specific tools and the most relevant components will be described in more detail.
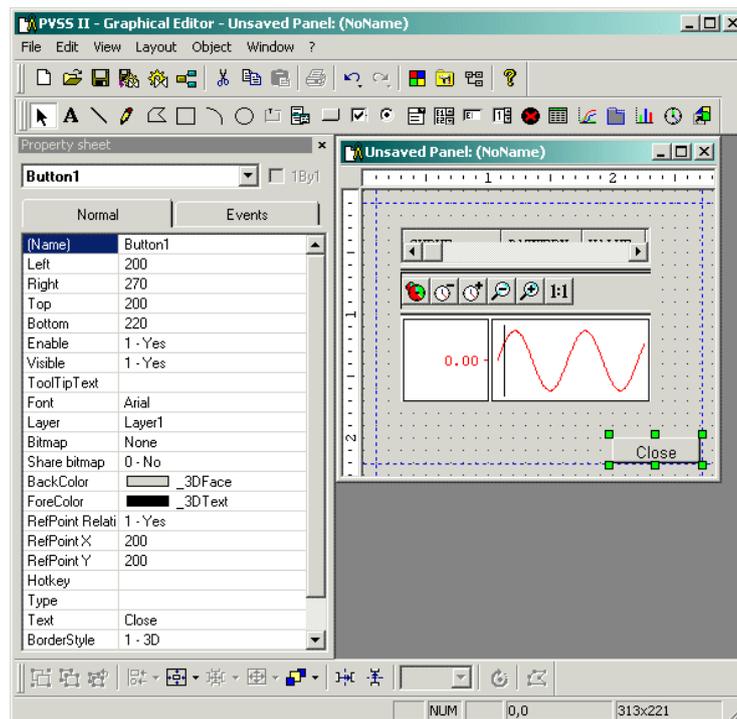
Figure 8.6: PVSS graphical editor.

8.4.1   THE JCOP FRAMEWORK OVERVIEW

The JCOP Framework provides a set of common components ready to be used by all LHC experiments. The project aim is to reduce effort in development and maintenance by re-using common components. These components try to hide the complexity of the underlying tools and equipment giving to the possible extent a high-level of abstraction and an interface for non-experts through PVSS-II.

Figure 8.7 gives an overview of the JCOP FW (Framework). As can be seen, the framework in principle covers all levels down to the connection to the hardware. However, as there is only agreement on the use of certain hardware elements and front-ends, the majority of the framework is provided at the supervisory level. Nevertheless, connection to other front-ends, and integration with the framework, is possible via one of a number of middleware communications interfaces (OPC, PVSS II Communications, DIM or DIP).

The Framework is divided into components. It provides an 'Installation Tool' to manage the components and install what is needed and ignore what is not needed. This tool increases the flexibility & reusability of the framework. Apart from the 'Installation tool', the components provided by the framework can be divided into three types: a core that contains general purpose functionality; devices to monitor and control hardware; tools to handle, display and store data.
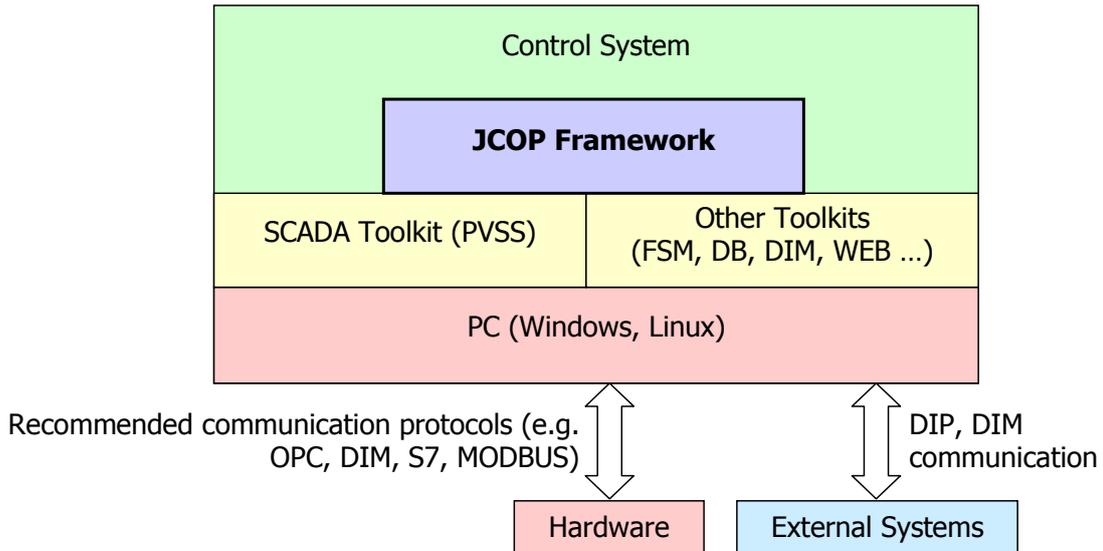
Figure 8.7: The layered structure and use of the Joint Controls Project framework, indicating its basis on both commercial (e.g. PVSS) and CERN-developed products (e.g. the distributed information manager (DIM) and finite-state machine (FSM) toolkit).

Following there is a list of the most important components and tool elements of the framework:

- To model the devices: the framework can model a front-end device directly as a piece of equipment (e.g. a high voltage crate) or as a logical group of devices (e.g. a group of channels). In the supervision layer, a device consists of one or more PVSS-II DPT, a library that encapsulates the knowledge to manage the device instances, and a set of displays to provide an interface for the user. Therefore, the developer can make the front-end layer to be specific for each application and, the only constraint at this layer, is to have an interface to one of the protocols supported within the framework (preference is given to protocols coming directly from the maker of the hardware, e.g. the OPC server of CAEN). The devices that are at disposal to the users are:

  1. Generic Analog-Digital Devices: this provides basic functionality to deal with inputs and outputs that can reside in a PLC, a fieldbus module, an I/O card, etc.

  2. CAEN HV Power Supplies: the power systems of CAEN are interfaced through OPC into PVSS II. The framework component provides full functionality to model and control the equipment of this manufacturer.

  3. Wiener LV Power Supplies: similarly to CAEN, this equipment is supported. In this case, the crates have either a CANbus or an OPC interface. All the models that support the standard Wiener protocol are included.

  4. Wiener Fan Tray: they follow a similar interface to the Wiener power supply.

  5. ELMB: through CANbus and CANopen the ELMB is interfaced into PVSS-II via OPC. This framework component allows operating the device easily.

  6. Logical Node: this device is a container for other device instances. Once grouped together, the set of instances offers the same interface as a single instance.

- The 'Device Editor & Navigator': entry tool for the framework that allows to manipulate the devices. It is the main user interface of the FW. The DEN permits to build simple device hierarchies with a logical structure, providing high level view of the system. In addition, it offers a user interface for the FSM modeling, so one can model the devices and system behavior by building a hierarchy of Finite State Machines.

- 'Controls Hierarchy': this tools permits building hierarchical control systems including Finite State Machines based on SMI++, and a hierarchy based on pointers. Due to the relevance of this component in the ECS design it will be thoroughly presented in the dedicated section 8.5.

- 'External applications': there are several systems that have to be integrated into the ECS. Due to their character they can be classified as 'External Applications' and they have been put together in one component since they are of common interest to all LHC experiments. Among these systems one can find:

1. 'Cooling and Ventilation' system: connects via MODBUS to the PLC that controls the cooling plant.

2. 'Magnet data exchange': connects via DIP to the information from the Magnet Control System.

3. 'Access Control System' data exchange: connects via DIP to the information from the LHC Access Control System (LACS).

- The 'communication protocols': the devices (hardware or software) have to communicate with PVSS and hence a kind of software interface is needed. Communication over a network using common middleware protocols is essential for inter-process communication between the different ECS elements. The DIM and the OPC (OLE for Process Control) are used as the main protocols for the middleware that interfaces the supervision and front-end layers. For the exchange of information with other CERN systems like the LHC machine or the technical services, the Data Interchange Protocol (DIP), will also be used. Whenever possible, an OPC server delivered by the equipment manufacturer should be used. As this is not always possible, the equipment can still be interfaced to PVSS by writing a DIM server. These three protocols are analyzed in more depth in the section 8.6.2.

- The 'Databases': the storage of data is an essential point in any control system. The databases are not explicitly a framework component, but the FW provides interface to them. We explain them next.

8.4.2 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 somewhere: configuration data of the devices, historical of monitoring measurements, etc. Three different databases for different purposes are used in the LHCb experiment:

- The PVSS archive: it contains all the data gathered from the devices used for online monitoring and error handling (i.e. HV value readings, temperatures values…). There are two different implementation choices for this data archiving, either use the PVSS data archiving system, based on proprietary technology and which the data distributed across different files and systems, or use the ORACLE database connection which is a more generic and scalable option.

- The Configuration Database: all data needed to configure the HW (or SW) for the various running modes (i.e. HV V0 Settings, pedestal settings, trigger settings, etc.). The database has been implemented using Oracle. See 8.5.5.3 for a closer description of its use.

- The Conditions Database: contains a subset of the monitoring data read from the hardware if it is needed for Event processing (e.g. HV readings if changed by more than n Volts) or some configuration data that has been used (e.g. FE settings used by a particular run).

The framework provides interfaces for accessing the mentioned Dbs. The Figure 8.8 shows a scheme of the three databases connected to different PVSS systems. The scheme in figure Figure 8.9 shows the databases dataflow. A PVSS system is connected on one hand to the experimental equipment and in the other hand to the three previously mentioned databases. The data interchange between the databases to equipment and vice versa, through the PVSS system, responds to the next flow path:

1. The PVSS system retrieves from the Configuration Db the data needed for configuring the equipment in a specific running mode. This data is the so-called recipe and it is applied to the hardware. As the behavior of the electronics can change in time, better configuration parameters can be found or more updated configuration values can be needed, a way to perform an update of the configuration data for following runs is needed, as sketched in Figure 8.9.

2. The monitoring data from the equipment is gathered by the PVSS system. This data can be further stored in archiving Db if required. The data from the archiving Db can be recovered at any time by the control system for analysis and displaying purposes of the historical data.

3. In case any monitoring or configuration data is needed for future offline data analysis, the relevant information shall be stored in the conditions database. This information corresponds to a condition change during the normal experiment running which will have an impact in the data analysis.
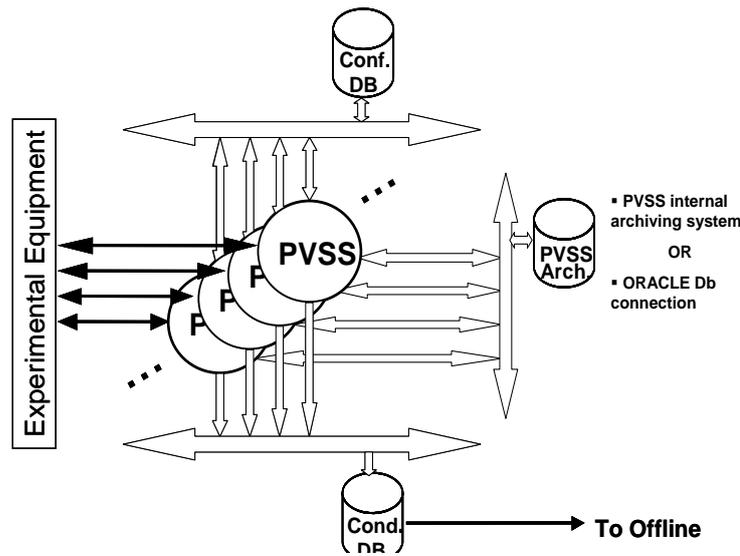
Figure 8.8: Online Databases.
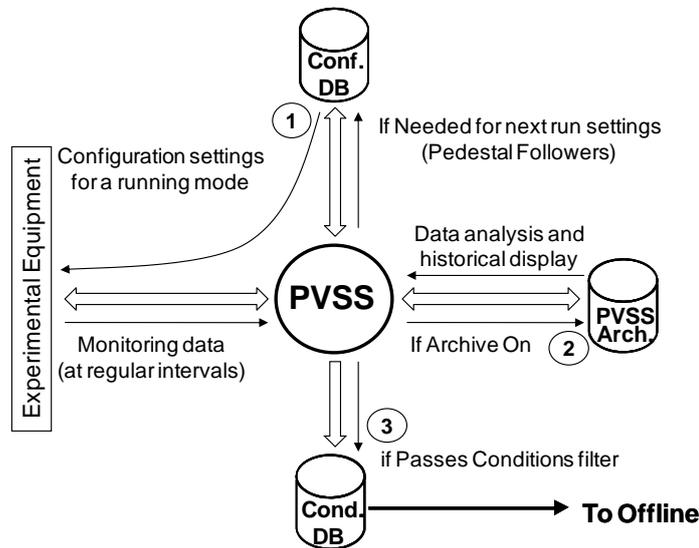
Figure 8.9: Databases dataflow.

8.4.3   THE LHCB SPECIFIC COMPONENTS

Apart from the described above, some relevant LHCb specific tools were developed, namely:

- 'SPECS framework': it is a framework to operate the SPECS related hardware. It permits handling the I^{2}C, JTAG, PBUS, REG (Register) & DCU functions of the SPECS slaves hiding the complexity of the hardware. This component relies on a generic DIM Server to be running ('SpecsServer') on each PC with a SPECS Master.

- 'CCPC framework': similar to the SPECS tool but for operation the CCPCs. The tool allows users to develop their own PVSS applications communicating with a generic ccpc server ('ccserv') running in the credit card PC.

- 'Hw Tool': generic tool to model the SPECS and CCPC interfaced hardware as a PVSS DPT. It permits to define registers for SPECS and CCPC hardware types (each register is represented by a DP with elements and settings). It is a hardware abstraction just to write to a register (specific name) or monitor it without having to know all the details of the hardware behind, hiding the DIM communication layer.

- 'TELL1 framework': component including DPT representations, recipes and user panels for the TELL1 boards control. It also includes the finite state machines for the TELL1 operation.

8.5   The FSM and ECS Software Architecture

The JCOP Finite State Machine (FSM) is the main component for the organization and automation of the ECS. The component provides a high-level interface to model the different parts of the experiment as objects that follow a FSM behavior, which in turn can be organized in a hierarchical structure across different systems using a distributed architecture. These objects, as shown in Figure 8.10, are then deployed vertically at the different levels of abstraction of the experiment, i.e., from the front-end equipment to the overall control).

Computing technology is permanently evolving providing components that ease the implementation of large distributed control systems. In that context, the JCOP FSM tool is the central component for the integration, sequencing and automation of control processes at the LHC experiments. This tool uses the functionalities provided by the PVSS-II and the State Manager Interface (SMI++). It permits modeling large control systems by means of many co-operative software objects that, hierarchically controlled, follow a finite state machine logic.

8.5.1   The Finite State Machine Concept

The concept of Finite State Machines (FSM) is used to model the behavior of a system by means of a limited number of states, transitions between these states, actions and events. A current state is determined by past states of the system, i.e. it reflects the input changes from the system start to the present moment. A transition indicates a state change and is described by a condition or rule that would need to be fulfilled to enable the change. An action is a description of an activity that is to be performed at a given moment. Finally, events, which are either externally or internally generated, trigger actions and can lead to state transitions.

The concept of FSM can be applied to a wide range of applications in both hardware and software. In addition to their use in modeling reactive systems as presented here, finite state automata are significant in many different areas, including electrical engineering, linguistics, computer science, philosophy, biology, mathematics, and logic. Finite state machines are a class of automata studied in automata theory and the theory of computation. In computer science, finite state machines are widely used in modeling of application behavior, design of hardware digital systems, software engineering, compilers, network protocols, and the study of computation and languages.

An FSM can be represented using a state diagram (Figure 8.10). There are different kinds of state diagrams that differ slightly and have different semantic. There are two classical methods for modeling state machines: Moore and Mealy machines. The main difference between both mechanisms is that whilst in Mealy an output is based on its current state and the inputs, in Moore the output is determined by only the current state not depending on the inputs.

In the world of the object-oriented software, state diagrams are represented by means of standardized notations, from which, the most relevant today is the Unified Modeling Language (UML) [69]. The UML state diagrams represent objects of a single class and track the different states of its objects through the system.

The magnitude of LHCb experiment, in terms of system complexity and collaboration effort, suggests that a hierarchical structure is the natural way to organize the detector. Thus, the ECS is organized in a tree like structure, segmented into sub-detectors, systems, sub-systems, etc. The hierarchical control and operation of the different sub-detectors is performed by means of FSMs, which handle the states and transitions of the different parts of the detector. Each FSM defines a particular task and like that the whole can remain understandable. A large amount of state machines are arranged forming hierarchical FSM with interconnections that carefully designed ensure the coherence of the entire system. The specific segmentation and hierarchical distribution of the LHCb control system is explained with more detail in section 8.8.
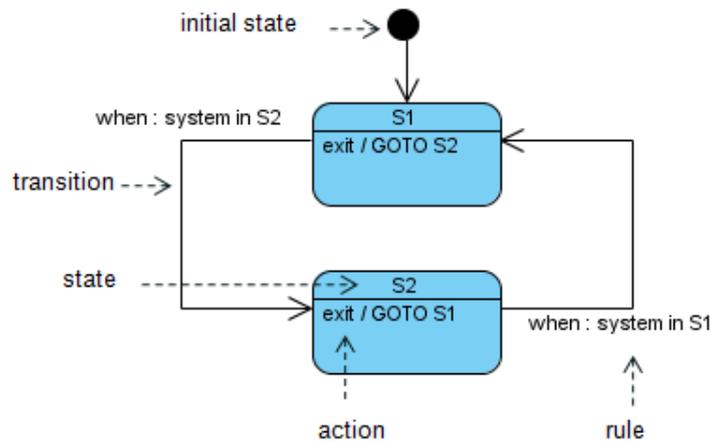
Figure 8.10: FSM concept in UML.

8.5.2   STATE MANAGEMENT INTERFACE (SMI++)

SMI++ is based on the original State Manager [70], which was developed by the DELPHI experiment [71] in collaboration with the CERN Computing Division.

With the SMI++ framework a control system is described as a collection of objects behaving as finite state machines which are associated with an actual piece of hardware or a real software task. Each of these objects interacts with the concrete entity it represents, through a so-called proxy process. The proxy process provides a bridge between the real and the SMI++ worlds: the SMI++ process has access to the real process data and at the same time, is able to command the processes.

The main attribute of an SMI++ object is its state. In each state, it can accept commands that trigger actions. An abstract object, while executing an action, can send commands to other objects, interrogate the states of other objects, and eventually change its own state. It can also spontaneously respond to state changes of other objects.

In order to reduce complexity of large systems, logically related objects are grouped into SMI++ domains. In each domain, the objects are organized in a hierarchical structure, and form a sub-system control. These basic concepts are graphically summarized in Figure 8.11.

All issues related to distribution and heterogeneity of platforms are transparently handled by the underlying communication system DIM on top of TCP/IP (see section 8.6.2.2).

The SMI++ framework consists of a set of tools. A special language called State Manager Language (SML) is used for the object description. The SML description is then interpreted by a logic engine called State Manager (SM) coded in C++ that drives the control system. Figure 8.12 shows an example of SML code.

8.5.3   STATE MANAGER LANGUAGE (SML)

This language allows for detailed specification of the objects, such as their states, actions, and associated conditions allowing the definition of classes of objects. The main characteristics of this language are the following:

- Finite-State Logic: objects are described as FSMs. The main attribute of an object is its state. Commands sent to an object trigger object actions that can change its state.

- Sequencing: an action performed by an abstract object is specified as a sequence of instructions which mainly consist of commands sent to other objects.

- Asynchronous Behavior: in principle, all actions proceed in parallel. A command sent by object A to object B does not suspend the instruction sequence of object A (i.e. object A does not wait for completion of the command sent to object B before it continues with its instruction sequence). Only a test by object A on the state of object B suspends the instruction sequence of object A, until object B reaches a stable state.

- Rule Based System: each object can specify logical conditions based on states of other objects. These, when satisfied, will trigger an execution of the action determined in the condition. This

provides the mechanism for an object to respond to unsolicited state changes of other objects in the system.
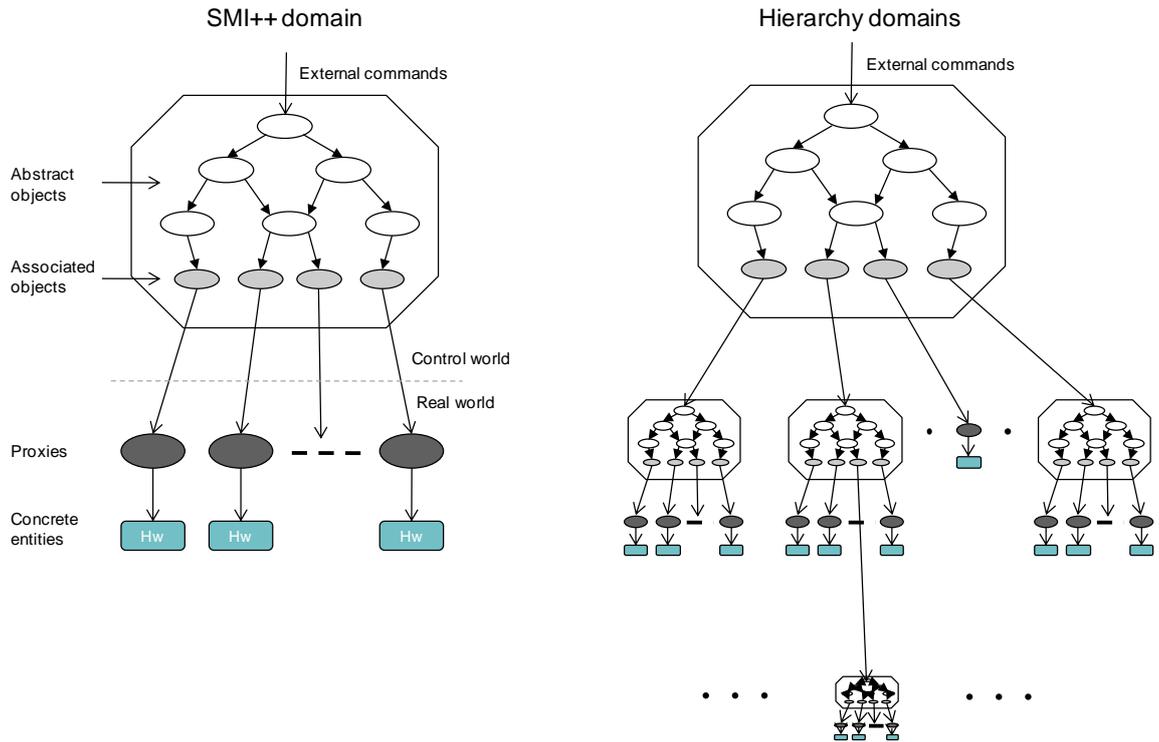
Figure 8.11: SMI++ domain(left) and hierarchy of domains (right).

An example of SML code is shown in Figure 8.12. Two different types of objects are declared, RUN_CONTROL (logical entity) and CONTROLLER (real entity). For both objects the list of possible states, actions and transition rules of each state are specified. For instance, in the object RUN_CONTROL the action *START_RUN* is only possible when it is in state *READY*, this action consists of sending command *MOUNT* to object TAPE and checking if the TAPE object reaches state *MOUNTED*. The action *START_RUN* will eventually terminate by setting the state of RUN_CONTROL to either *RUN_IN_PROGRESS* or *ERROR*. In the state *RUN_IN_PROGRESS* a set of transition rules (*WHEN* clauses) are defined. These rules are valid during the time the object is in this state and indicate that whenever one of them becomes true the corresponding action will be executed.

```
object: RUN_CONTROL                            object : READOUT_CONTROLER/associated
    state: READY                                   state : READY
        action : START_RUN                             action : START
            do MOUNT TAPE                              ...
            if (TAPE not_in_state MOUNTED) then    state : RUNNING
                do MOUNT_ERROR ERROR_OBJ               action : PAUSE
                move_to ERROR                          action : ABORT
            endif                                      ...
            do START READOUT_CONTROLER
            if (READOUT_CONTROLER in_state RUNNING)
                move_to RUN_IN_PROGRESS
            ...
    state : RUN_IN_PROGRESS
        when (TAPE in_state FILE_FULL)
            do PAUSE_RUN
        when (READOUT_CONTROLER in_state ERROR)
            do ABORT_RUN
        action : ABORT_RUN
            ...
```

Figure 8.12: SML code example.

8.5.4 STATE MANAGER (SM)

This is the key tool of the SMI++ framework. It is a program which, at its start-up, uses the SML code for a particular domain, and becomes the SM of that domain. In the complete operating control system there is, therefore, one such process per domain (called 'smiSM' process).

When the 'smiSM' process is running, it takes full control of the hardware components assigned to the domain, sequences and synchronizes their activities, and responds to spontaneous changes in their behavior. It does it by following the instructions in the SML code, and by sending the necessary commands to proxies through their associated objects. In a given domain, it is possible to reference objects in other domains. These are then locally treated as associated objects, with their relevant proxies being the other SMs. In this way, full cooperation among SMs in the control system is achieved.

The SM implements a DIM interface that is used for 'smiSM' inter-process communication and communication with other external objects or user interfaces.

8.5.5   JCOP FSM, RESULT OF THE SCADA - SMI++ INTEGRATION

Due to the large amount of devices to control the decision of developing a supervised, distributed hierarchical control system was chosen for the four LHC experiments. The JCOP provides a framework to build, amongst other things, hierarchical tree-like structures based on Finite State Machines (FSM) to model large control systems.

The idea behind this is building the ECS as a highly hierarchical system that: allows a high level of independence between different parts of the system, for example, between different sub-detectors when running in stand-alone mode (during commissioning, calibration phases…); allows integrated global and central control of the whole experiment when running in physics data taking mode.

It is considered a supervised control system because in some of the experiment parts there is no aim on having a completely autonomous system and just some of them will be automated. The reason for this is that it would be a colossal work to completely automate the operation of the tens of thousands of devices that comprise the detector, which in addition are really prone to exhibit errors due to the hard conditions (high radiation and magnetic fields) where they are being used.

For that reason SMI++ has been interfaced to the PVSS SCADA system, in order to provide a JCOP framework component for 'Hierarchical Control' based on FSM. This has been accomplished thanks to the openness that the PVSS API offers for implementing new functionalities such as a DIM client interface.

It is the combination of functionalities of both PVSS and SMI++ that provides a fully featured tool to build hierarchical control systems:

- SCADA provides a database to store the SMI++ description and configuration, i.e. the same database will contain device description and behavior.
- SCADA provides device access (OPC, Profibus, drivers).
- SCADA provides alarm handling (Generation, Filtering, Masking, etc)
- SCADA provides archiving, logging, trending.
- SCADA provides user interface builder to be used by the SMI++ developer.
- SCADA provides alarm display, access control, etc.
- SCADA provides scripting language to derive a state out of monitored data and to implement actions on the devices.
- SMI++ provides abstract behavior modeling (Finite State Machines).
- SMI++ provides automation & error recovery (rule based system).

8.5.5.1   TYPES OF JCOP FSM OBJECTS

JCOP defined a hierarchical control composed of two types of different nodes: "logical" and "real" entities. Whilst real entities normally model and control directly hardware devices (e.g. a pump), the logical entities group these devices and control them from a more abstract hierarchical level (e.g. a sub-detector).

The real entities are called 'Device Units' (DU), whereas the logical ones can either be 'Control Units' (CU) or 'Logical Units' (LU):

- 'Control Unit'. It is an abstract object (e.g. whole LHCb, TT sub-detector, IT HV system, etc.) corresponding to one 'smiSM' process capable of containing children of any type (i.e. CUs, LUs or DUs). These objects are written in SML, so they implement a logic behavior in terms of FSM. Therefore they can take decisions and act on their children (i.e. send them 'Commands') based on their 'States'. Any 'Control Unit' and the associated sub-tree can be a self-contained entity. State transitions can be triggered by: command reception (either from its parent or from an operator) or

state changes of its children. State transitions cause the evaluation of logical conditions and possibly 'Commands' to be sent to the children. This mechanism can be used to propagate actions down the tree, to automate operations and to recover from error situations.

- • 'Logical Unit'. It also represents an abstract object, but they are embedded in an overlying CU, so they are part of a 'smiSM' process. It can contain children, but not of type CU. LUs have restricted functionality compared with CUs, but by using LUs the number of 'smiSM' processes is reduced, and thus the performance improved by the use of computing resources. Hence, by using LUs hierarchies with larger number of nodes can be created. These objects are also written in SML.

- • 'Device Unit'. It corresponds to a concrete object in PVSS, they implement the interface with the lower level components (Hardware or Software) and, in that case, they cannot contain children of any type (they are always a tree "leaf"). The behavior of these objects is defined in the PVSS scripting language instead of SML code. Consequently, since these objects can be inserted at any level, they can easily add functionality to the control tree by using the power of the SCADA system. They can receive commands and act on the device, as well as they can gather device data and translate it into a state.

With these different kinds of objects we will describe the behavior of LHCb in terms of a concurrent distributed hierarchical FSM. All the FSMs working together will permit the operation of the LHCb experiment as well as the single independent operation of every sub-detector. Figure 8.13 illustrates an example of hierarchical structure architecture with CUs, LUs and DUs. It also depicts the flow direction of commands, states and alarms: the alarms and states generated in the devices flow from the bottom upwards to the top, and the commands descend from the top control units to the devices at the bottom of the hierarchy.
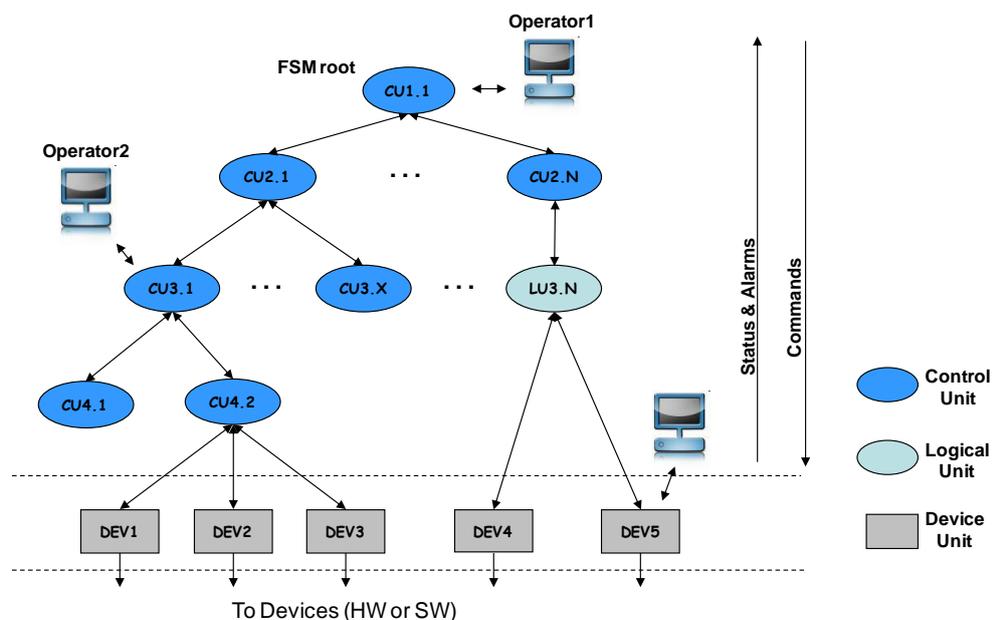
Figure 8.13: ECS software architecture.

8.5.5.2 PARTITIONING

The global ECS, built up of many different specific sub-detectors, requires the capability of operating either the whole experiment or sub-parts of the system independently. This capability is accomplished with the partitioning feature provided by the JCOP FSM tool [72]. This characteristic is very appreciated in the HEP (High Energy Physics) world where large number of independent teams and very different operation modes coexist.

The partitioning concept entails the introduction of the notion of ownership. In order to be able to send commands to the different components an operator can reserve the whole tree or a sub-tree in which case he/she becomes the "owner". Each component has one and only one owner at any time. All components of a sub-tree have the same owner.

The components can receive commands from only one or from more operators depending on their exclusivity mode:

- Exclusive mode: only the owner can send commands.
- Shared mode: any other operator with the correct access rights can also send commands.

Only the owner can change from one mode to the other.

Concerning the different partitioning modes, Figure 8.14 shows graphically the partitioning modes supported only by the CUs. Each CU knows how to partition "out" or "in" its children. Excluding a child from the hierarchy implies that its state is not taken into account any more by the parent in its deciding process, that the parent will not send commands to it and that the owner operator releases the ownership so that another operator can work with it. Only the owner can exclude a component from the hierarchy.
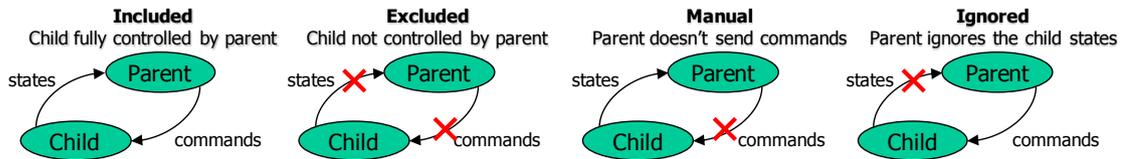
Figure 8.14: partitioning modes supported.

8.5.5.3 DEVICES CONFIGURATION – THE RECIPES AND THE CONFIGURATION DB

During the lifetime period of the LHC detectors, the configuration parameters of the detector control systems will vary as a result of changes in the mode the detectors are operated, sensor re-calibrations due to equipment ageing, etc. The amount of parameters that need to be configured before the detector is ready for operation is considerably huge, so in order to ensure a consistent management and safe permanent storage of these parameters, an approach based on a Configuration Database was envisaged [73].

The database contains the data that the ECS needs to configure the experiment's equipment [74]:

- Static properties: properties that rarely or never change. Changes in this data involve a hardware intervention.
- Dynamic properties: properties that change frequently e.g. alarm settings, hardware settings such as ramping speed, calibration constants, pedestals…

The static data can for instance be the hardware address, the geographical location or device structure of the equipment. Also the connectivity of devices (e.g. a switch can be connected through one of its input ports to a supervisor and through one of its output ports to a card somewhere in the detector) and hierarchical devices allocation describing how components are nested (e.g. a chip sits on a board that can be found in a crate which is placed in a rack) shall be stored.

The dynamic configuration data is grouped into the so called recipes. A recipe is a collection of settings (values or/and alert parameters) corresponding to a number of DPs in PVSS, that is referred to as a whole, and identified by a unique name. This collection of data could vary depending upon the partitioning and running mode of the experiment. Applying a recipe to a control system may be seen as executing a complex command that involves setting a large number of related parameters and reconfiguring alert thresholds. Evolving versions of recipes may be saved to the configuration database, where the changes are tracked as subsequent versions of dynamic configuration data.

As the architecture of the Control System is based on hierarchical FSMs, a FSM driven tool for automatically downloading from the Db and applying the recipes has been created [73]. The 'FSM-Configuration DB Tool' automates the handling of configuration data required at run-time by the detector control systems. The tool bridges the FSM of the control systems and the configuration database together to ensure the availability of all configuration data for a given type of run, during the FSM transitions.

The 'FSM-Configuration DB Tool' integrates in the FSM tree specific device units called, in the terminology of the tool, *configurators*, which act as links between the FSM and the Configuration DB, as shown in Figure 8.15. The *configurators* make use of the functionality of the tool to access the database and preload the configuration data into the recipe caches. Following a FSM command, these caches can be accessed by the *configurators* to apply the recipes to the devices in the hierarchy tree. In order to cope with the eventual partitioning of the tree, the tool adds one *configurator* per FSM control domain to manage the configuration of all devices in the sub-tree. This guarantees the availability of the configuration data during stand-alone operation of a subsystem or throughout the parallel operation of multiple parts of the detector according to different run modes.
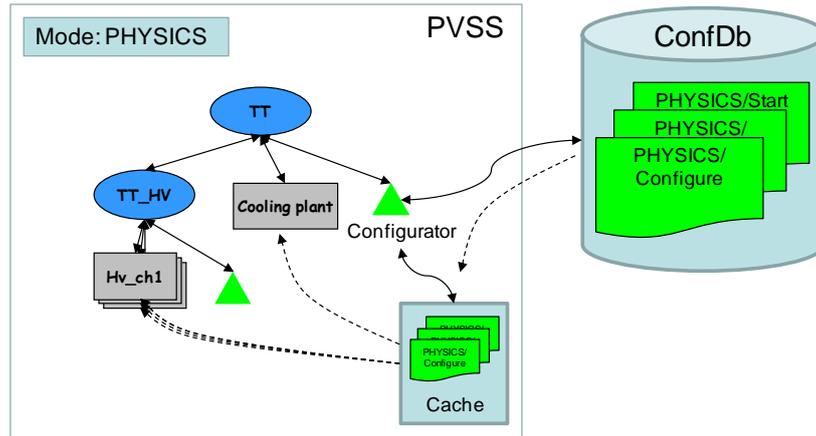
Figure 8.15: Configurator loading recipes and applying the recipe to the hardware.

8.6   COMMUNICATION PROTOCOLS IN THE ECS

Figure 8.1 showed a general view of the LHCb ECS hardware and software view: on the left side the elements that buil up the system and on the right the corresponding logical layers, and used technologies, sub-divided into commercial and custom developed.

A clear distinction between fieldbuses and Local Area Networks (LANs) has to be made at this point. The two of them are widely used in the LHCb experiment and although they are sometimes used for the same purpose (control data transmission) they should not be confused.

Fieldbuses are industrial network systems for real-time distributed control. It is a way to connect instruments in a plant. Fieldbus works on a network structure which typically allows daisy-chain, star, ring, branch, and tree network topologies. While fieldbuses are the links used to communicate with field devices (HV power supplies, PLC, etc.), LANs are links for inter-computer communications.

In order to facilitate interoperability between equipment and software and have a common way to access data from any source, the OPC communication protocol is used in commercial standard equipment while DIM and DIP are used for custom LHCb hardware developments. These different middleware protocols are also discussed in this section.

Figure 8.16 shows a scheme of the OSI model of communications in LHCb: the CAN fieldbus is used to connect with the ELMB devices and the Wiener crates and the cooling system; The SPECS communication system is a serial communications fieldbus based on a single master multi-slave bus that allows simple and fast means of communication between with FE electronics; OPC and DIM are application layer protocols used to interface the equipment of the different field buses with external systems.

8.6.1   INTERFACE TO ELECTRONICS

As emphasized earlier, all the LHCb electronic boards and chips must be under the control of the ECS, so a physical hardware interface that carries the communication between the ECS running in dedicated computers and the experiment electronics must exist. The electronics used in LHCb can be clearly divided into two groups:

- The boards and devices placed inside the experiment area hence exposed to a radiation and magnetic field hostile environment. This electronics has been designed to use simple communication protocols and require only the I^2C, JTAG and parallel bus protocols.

- Boards and systems in the counting rooms, behind the concrete shielding wall, and not exposed to radiation. These boards make use of large memory chips and/or processors and require I^2C, JTAG, parallel bus, PCI interfaces to access the board components, TCP...

The hardware architecture approach followed in the experiment was designing the electronics such that each piece of equipment (a board, an assembly of boards or even a part of a detector) contains an interface that provides the necessary protocols: I^2C, JTAG or parallel bus. This interface can either be a slave device that connects to a master controller placed in a PC by a fieldbus or an "autonomous" interface that can be directly accessed via the network. In the case of the on-detector electronics, where the radiation level is high, the

interface is able to drive I^2C and JTAG up to 15 m where the FE chips are located. This avoids the need for radiation hard technology for the controller devices and they can just be qualified for radiation tolerance.

Three different solutions have been agreed by the LHCb collaboration, the SPECS and ELMB for the radiation areas and the CCPC for non radiation environments.

Middleware Domain	DIM	OPC	CANOpen	SPECS libs	Application
		DCOM			Presentation
	DIP				Session
Data Transport Services	TCP				Transport
	IP				Network
	Ethernet	CAN Controller	SPECS M/S		Data Link
		CAN Tx/Rx	SPECS Tx/Rx		Physical
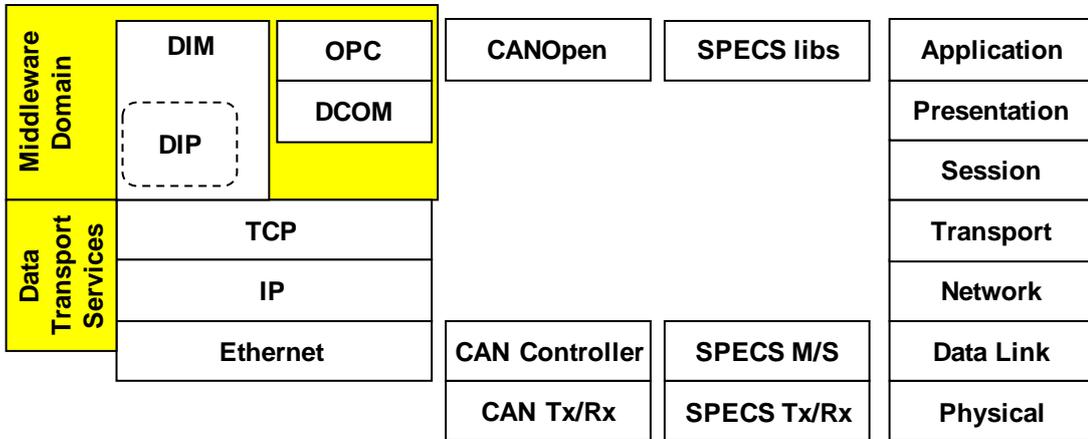
Figure 8.16: OSI model.

8.6.1.1 CREDIT CARD PC (CCPC)

A miniature PC on a commercial credit card sized hybrid is mounted on the front-end module and interfaces to the higher levels of the ECS system over 10/100 Mbit Ethernet. It is used to control the electronics in the counting rooms, especially VME sized boards (for example the TELL1s) in which a CCPC is installed. The CCPC contains an Intel Pentium compatible CPU and 64 MB of RAM. An additional adapter card needs to be used to provide I^2C, JTAG and parallel bus. Two kinds of dedicated glue-cards providing the interfaces needed for controlling on-board electronics devices have been designed. The CCPCs run Linux, are booted remotely via the network and are intended to be used in non radiation environments.
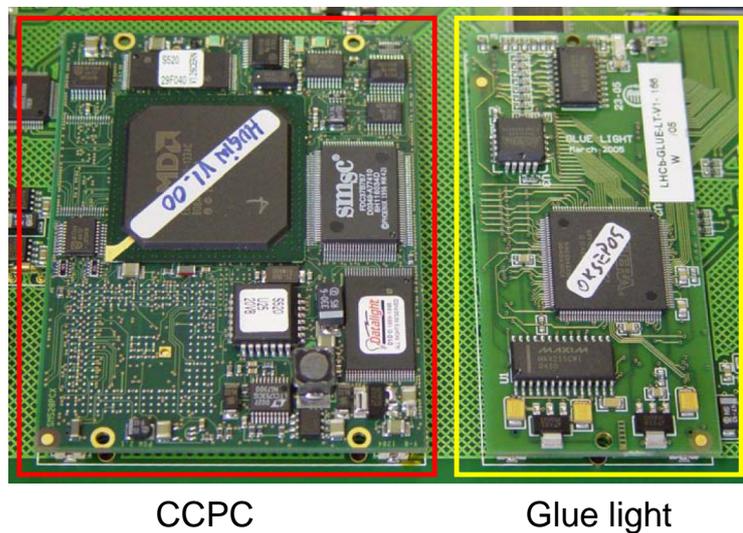
CCPC   Glue light

Figure 8.17: CCPC and glue light cards.

8.6.1.2 SPECS

SPECS is a simple custom protocol similar to I^2C, made specifically for applications in high energy physics. It is an evolution of the ATLAS SPAC (Serial Protocol for the Atlas Calorimeter) with improved radiation tolerance and increased functionalities. It consists of a single master located in a standard PC communicating with multiple slaves place at distances of up to 100 m at a data rate of up to 10 Mbits/s thanks to the use of LVDS signaling.

The SPECS masters are implemented as PCI boards that house four masters. The SPECS slave is implemented in radiation tolerant FPGA and triple voting logic which makes it suitable to work in environments with limited radiation levels.

The slaves provide the interface to I^2C, JTAG and parallel buses, in addition to several multipurpose I/O and lines and 5 ADC channels. For this reason, the natural way of using this system is placing a SPECS slave in a mother board close to the electronics to be interfaced.

Figure 8.18: Specs slave mezzanine card.

8.6.1.3 ELMB

The Embedded Local Monitoring Board is general purpose plug-in board developed for the monitoring and control of ATLAS sub-detectors front-end equipment. It is based on an ATMEL microcontroller and the industry standard CANbus. CANopen has been implemented as high-level communication protocol. It contains 64 multiplexed ADC channels and multiple general purpose digital I/O lines that could eventually be used for I^2C and JTAG communication. The CAN bus runs at a speed of 500 Kbits/s for a bus length of 100 m. The ELMB is foreseen to be used in environments with modest radiation levels since the microcontrollers are not SEU safe. Any commercial CAN master card can be used to control the CAN bus to which the ELMB are tied. They are basically used for the racks environmental monitoring in the counting room.

Figure 8.19: ELMB plug-in board.

8.6.2 MIDDLEWARE PROTOCOLS

Middleware is computer software that interconnects software components or applications. It basically consists of a set of enabling services that allow multiple processes running on one or more machines to interact across a network. This kind of technology evolved to provide for interoperability in support of the move to coherent distributed architectures, which are used most often to support and simplify complex, distributed applications. Middleware sits "in the middle" between distributed application software working on different operating systems. It is similar to the middle layer of a three-tier single system architecture,

except that it is stretched across multiple systems or applications. Examples include database systems, telecommunications software, transaction monitors, and messaging-and-queueing software.

The reason for using middleware in process control comes from the fact that in the past different manufacturers of equipment developed their own technologies for remote device control. One had to develop custom dedicated software if wanted to interconnect several devices from different vendor and perform an online control and supervision over them. Although this approach is usually fine for simple small systems, it becomes expensive in terms of effort, manpower development and maintenance costs as soon as the system grows in size and complexity.

OPC technology was developed in the industry world with the aim of overcoming the mentioned problems by defining a standard protocol interface for publishing and accessing configuration data of the equipment. DIM and DIP are protocols defined and developed at CERN which are a powerful tool for the development of middleware control process applications for custom developed hardware.

Middleware fits in the OSI model as shown in the following example (Figure 8.20). Middleware supports the application, presentation and session layers, and may involve in some cases some portion of the transport layer. The middleware domain is always between the application and the operating system and network services, but the exact area covered is defined slightly differently in the OSI model and in the TCP/IP model.

8.6.2.1 OPC: OLE FOR PROCESS CONTROL

OLE for Process Control (OPC) is the original name for an open standard specification developed in 1996 by an industrial automation industry task force. The OPC standards specify the communication of industrial process data, alarms and events, historical data and batch process data between sensors, instruments, controllers, software systems and notification devices.

The standard specifies the communication of real-time data between control devices. The specification defines a standard set of objects, interfaces and methods for use in process control and manufacturing automation applications to facilitate interoperability. Then, by the usage of OPC, it is easy to integrate different vendors' systems such as power supplies or PLCs in large-scale facilities and, as so, OPC has naturally become a standard at CERN to interface the equipment with the supervisory level.
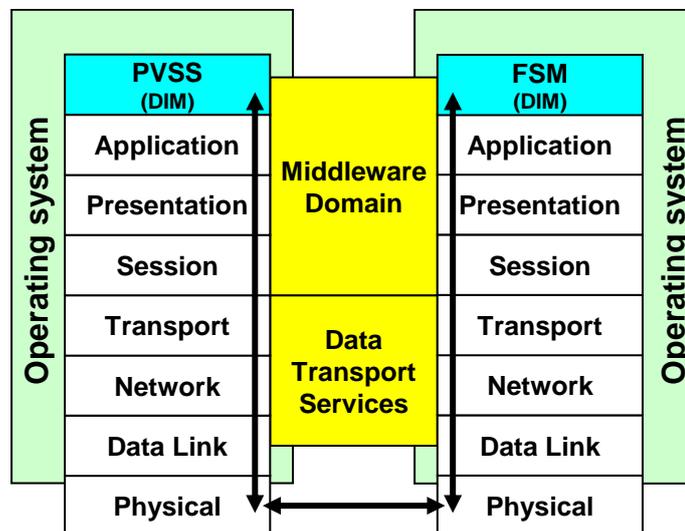
Figure 8.20: OSI model and middleware case in LHCb.

OPC was originally based on the Microsoft OLE technology (hence its name), but it was soon replaced by the Component Object Model (COM) and Distribute COM (DCOM). These technologies provided multi-client capability (access to the same an OPC server by several OPC clients) in distributed systems. OPC was designed to bridge 'Windows' based applications with process control hardware and software applications. It is an open standard that permits a consistent method of accessing field data from plant floor devices. This method remains the same regardless of the type and source of data. OPC servers provide a method for many different software packages to access data from a process control device, such as a PLC or DCS. Traditionally, any time a package needed access to data from a device, a custom interface, or driver, had to be written. The purpose of OPC is to define a common interface that is written once and then reused

by any business, SCADA, HMI, or custom software packages. Once an OPC server is written for a particular device, it can be reused by any application that is able to act as an OPC client (Figure 8.21).
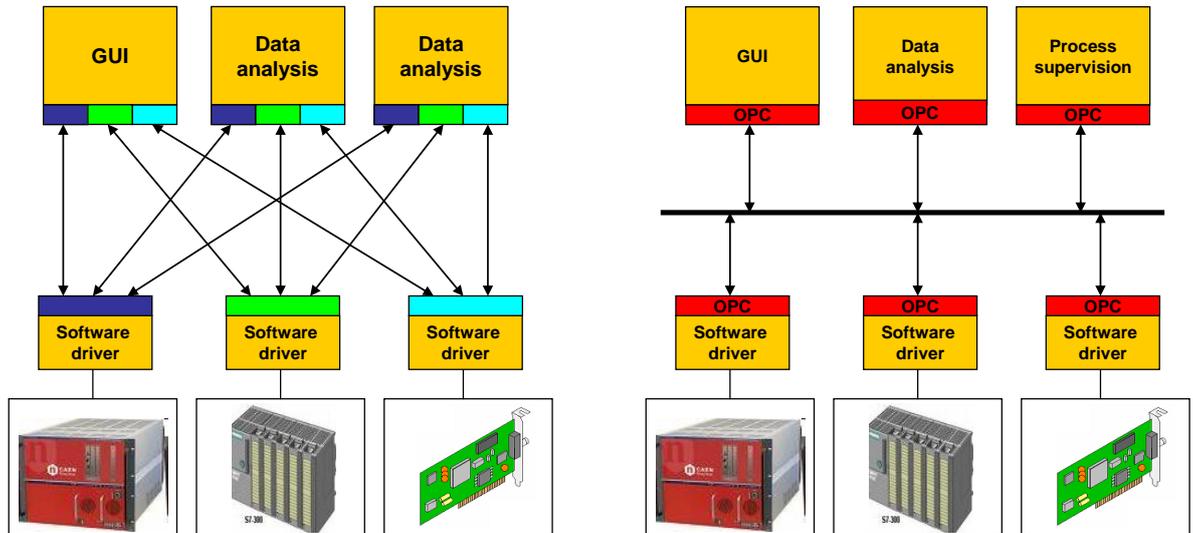
Figure 8.21: Interconnectivity problem (left) and solution (right) using OPC technology.

The next stage of OPC is the OPC Unified Architecture (OPCUA). OPCUA replaces DCOM technology by a current communications Service Oriented Architecture (SOA) model and introduces a cross-platform architecture for process control, while enhancing security and providing an information model.

OPCUA can be implemented with Java, Microsoft .NET, etc., and is not a 'Windows' based platform anymore, which is one of the most important drawbacks of OPC. It combines the functionality of the existing OPC interfaces with web services technologies such as XML (eXtensible Markup Language) messages or the SOAP (Service Oriented Architecture Protocol) standard to deliver higher level of support. As a result, OPCUA looks to become the standard for exchanging industrial data, however it is too early (too late) to implement these new technologies into the controls of the LHC experiments. DIM and DIP are used to cover the hole that OPC had left open.

8.6.2.2 DIM: DISTRIBUTED INFORMATION MANAGEMENT

DIM was originally developed for the DELPHI experiment at LEP.   DIM allows inter-process communication between heterogeneous processes running on 'Windows' systems and on 'Linux' with interfaces available in both C, C++ and Java.

DIM, like most communication systems, is based on the client/server paradigm. The basic concept in the DIM approach is the concept of service which is provided by the server to the client.

A service is a set of data (of any type or size) which is normally requested by the client (normally once at start-up) in different ways: (a) the client requests information only-once; (b) the client requests the information to be updated at regular time intervals; (c) the client requests the information to be updated whenever it changes; and (d) the client sends a command to the server.

In order to allow for the required transparency (i.e. a client does not need to know where a server is running), as well as to allow for easy recovery from crashes and migration of servers, a name server is present. Servers publish their services by registering them within the name server (normally only once, at start-up). Clients subscribe to services by asking the name server which server provides the service and then contacting the server directly, providing the type of service and the type of update as parameters. The name server keeps an up-to-date directory of all the servers and services available in the system. The interaction between servers, clients and the name server can be seen in the Figure 8.22.
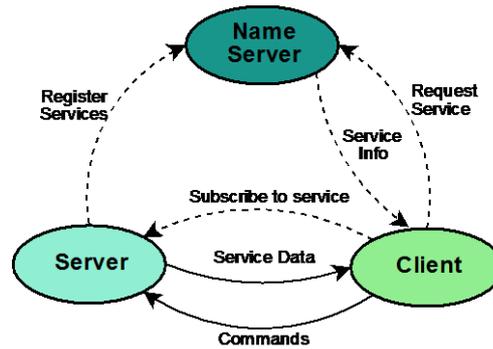
Figure 8.22: DIM service (picture taken from DIM web site http://dim.web.cern.ch).

The PVSS-DIM toolkit provides a way to interface devices which do not provide any of PVSS's supported protocols (such as OPC). Thanks to a custom PVSS manager developed at CERN, PVSS can behave as a DIM Client (i.e. receive information from or send commands to DIM servers) or as a DIM Server (i.e. send information to or receive commands from DIM clients). The PVSS-DIM toolkit provides a bridge between PVSS and DIM services.

8.6.2.3 DIP: DATA INTERCHANGE PROTOCOL

DIP is a simple and robust publish/subscribe system which supports an on-change data exchange. DIP defines a single data exchange mechanism between loosely coupled systems involved in the LHC operations. It allows small amounts of real-time data to be exchanged between systems that don't need very low latency and the exchanged data should always be summarized data instead of low-level parameters (i.e. cooling plant status rather than the opening level of a particular valve).

DIP is based on the DIM protocol and for this reason the publisher and its subscribers don't know each other explicitly, since it is the DNS that connects the subscribers with their publishers of interest.

DIP supports primitives, arrays of primitives and complex structures of primitives and it is available for C++ and Java both for Windows and Linux. The DIP data format includes a timestamp and quality flag. There is a negotiated contract between the consumer and the provider. Hence, the consumer is expected to know the name of the data item, its meaning and its data type.

There are no particular security mechanisms implemented in the DIP protocol. For example, a subscriber has no means to authenticate the source of the data it receives. Similarly, the DIP protocol doesn't offer a publisher the possibility to restrict access to a set of authenticated subscribers. Another example, the publications names are not nominative. In other words, when a publisher stops, the publications names it was using are from then on freely available to other publishers.

There is no feedback to a publisher that the data published actually reached its subscribers (also known as "one-way" communication). Hence there is no retransmission in case of a data delivery issue.

As for DIM, a custom PVSS manager was developed at CERN to provide PVSS with a client which can "talk" to any DIP server.

8.7 THE ONLINE SYSTEM

The Online System in LHCb comprises all aspects of online computing, experiment controls, as well as the Timing and Fast Control. It provides the infrastructure for the data taking for the High Level Triggers, as well as the control, configuration and monitoring of the entire experiment. A scheme of the online system is depicted in Figure 8.23.

In order to properly understand the Online system, one must previously remind that the LHCb experiment currently implements a two stage trigger level (while in the past were three). The LHC beam crossing rate is of ~40 MHz and the visible interaction rate at LHCb is about ~10 MHz. The L0 trigger is hardware implemented and it implements an accept rate of ~1 MHz, whereas the High Level Trigger (HLT) is a completely software trigger that reduces the accepted event rate to ~2 kHz.

Small detector signals are amplified and sampled in the L0 FE electronics at the rate of ~40 MHz and then stored in the L0 pipeline buffer during the L0 latency. Upon the reception on a L0 accept, data are extracted from the L0 pipeline and passed to the L0 derandomizer buffer waiting to be transferred to the L1 front-end

electronics. The L0 trigger decision time must be lower than the ~4 µs L0 FE pipeline length otherwise the data will be lost. The data is further processed in the L1 stage, sent to the to the readout network at an approximate rate of ~1 MHz and arrives at the HLT farm where the accepted event rate is finally reduced to a ~2 KHz event rate and sent to final storage.

A brief description of the LHCb DAQ and the TFC systems is given below. The ECS system itself is the whole of this chapter.

8.7.1   THE DAQ SYSTEM

The DAQ part is responsible for sending the selected data from the sub-detectors to permanent storage. In the DAQ system there are two separate networks, the data network used for transmitting the data from the FE electronics to the storage and the controls network for the DAQ infrastructure ECS.

Upon a positive trigger decision all front-end electronics boards (top) send their data to the L1 readout boards, where data are filtered, packaged and merged to optimize the load on the network data links. Data is further transmitted to the HLT farm through readout network at the rate of 1 MHz. This is done by means of a standard Gigabit Ethernet switching network, the central element of which is a large Gigabit Ethernet switch (currently capable of handling more than 120 Gb/s). Data belonging to the same event is sent to the same destination, that is, the fragments belonging to a specific event are guaranteed to reach the same entry point into the event filter farm, where the event is assembled and sent to the CPU where the HLT algorithm operates on the entire event and reduce the data to a finally accepted rate to storage of some 2 to 5 KHz (depending on the luminosity of the detector). Once the event has been accepted by the HLT it goes to permanent storage. The mass storage system is called CASTOR and it is hosted at CERN, however the storage network in the LHCb pit allows for several days of data storing in case of problem with CASTOR.

In addition to the HLT farm where the HLT algorithms run, there is also a monitoring farm that allows sub-detectors to run special analysis jobs to determine the sub-detector performance. A partial reconstruction of the physics data or a subset of data coming from specific scheduled triggers will allow the monitoring of the data quality. The analysis jobs should produce histograms that will be displayed by the control system.
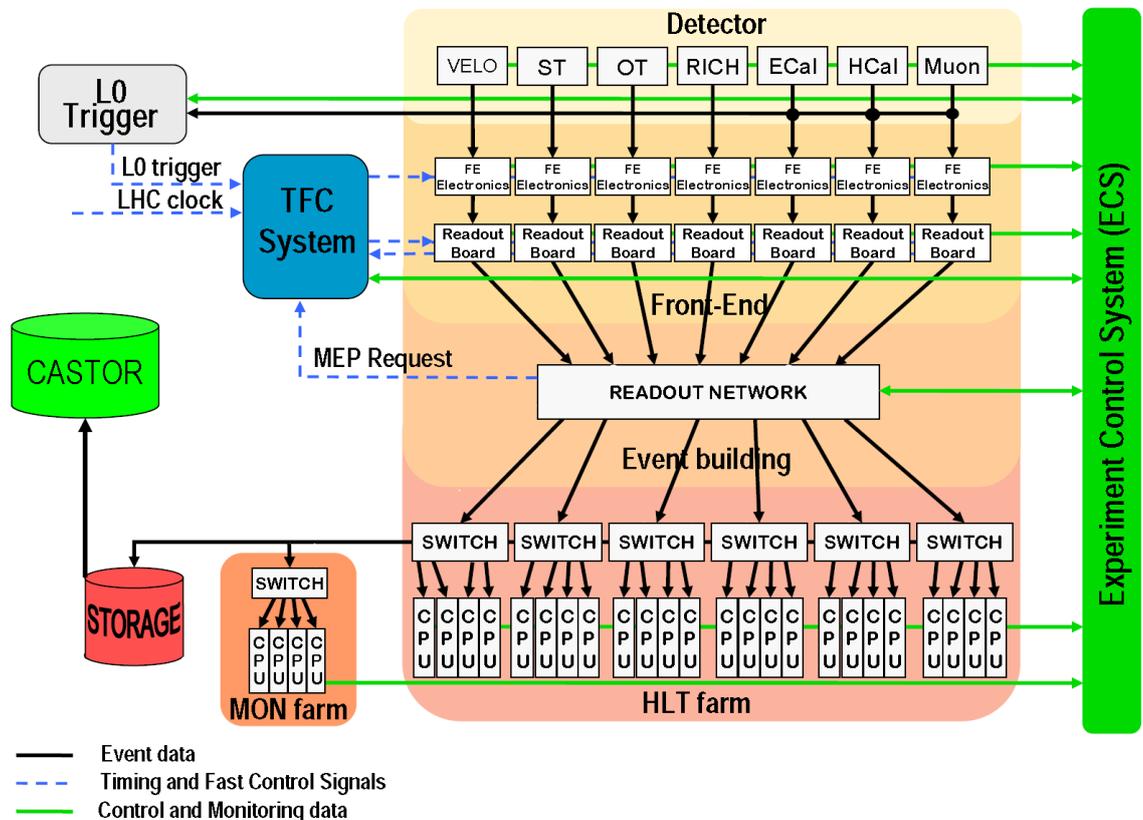
Figure 8.23: the LHCb Online system. Figure from [75].

The protocol used for the data transport is a simple push-through protocol. There is no lateral synchronization and no central event-manager like entity. Every source sends data as soon as it is ready to do so, and it is

assumed that the destination is always ready to receive. To avoid random loss of data due to buffer overflow, a throttling mechanism has been introduced. Components of the system can disable the trigger by sending a signal to the TFC system, which will stop further events from entering the system, until the condition clears.

8.7.2 THE TFC SYSTEM

The Timing and Fast Control (TFC) system controls the entire readout of the LHCb detector from the FE electronics up to the distribution of events to the processing farm. The TFC has three principal tasks:

- Distribute the LHC clock, the trigger decisions to the front-end buffers at a rate up to 1.1 MHz and all the fast signaling.

- Protect the online system from buffer overflows by means of buffer occupation simulations and signal throttling in non deterministic behavior devices.

- Drive the event data distribution to the different farm nodes implementing a load-balancing mechanism.

As the name suggests, the TFC system not only distributes the timing with a minimum jitter of the LHC clock to the front-end and readout electronics but also all the related fast control signaling. It provides trigger control, distribution and data routing and incorporates functionality to prevent buffer overflows in the entire readout chain. Besides it implements enough functionality to perform different types of auto-triggering for tests and calibration runs.

The TFC system comprises the LHCb central TFC elements, the TTC (Timing and Trigger Control) distribution network and the TTC transceivers in the FE electronics. The central elements are essentially the Readout Supervisor (also known as ODIN), the programmable TFC switch (THOR), the Throttle switch (MUNIN) and the Throttle OR (HUGIN).

The Figure 8.24 shows a simplified scheme of the TFC architecture layout. The ODIN is the master of the TFC system, it has the vital role of performing fast control of LHCb in such a way that it guaranties a safe data taking with the no risk of overflow in the FE and DAQ systems.

The ODIN is the main real-time controller of the complete front-end system and hence performs a long list of crucial tasks:

- Receive the LHC reference clock at ~40 MHz (bunch crossing and bunch structure synchronization signals) as received from the LHC timing generators and distribute it to all FE modules.

- Generate and distribute real time resets to front-end.

- Receive L0 trigger decisions from L0 trigger decision unit, ensure the correct synchronization of each trigger and redistribute it to the FE.

- Control and regulate the flow of L0 trigger accepts to prevent buffer overflows in the L0 electronics, the L1 front-end and the DAQ system.

- Distribute Event readout types to L1 electronics.

- Handle event packing factor for Multi Event Packets (MEP) depending on trigger types.

- Distribute Multi Event Packet destinations to L1 front-end electronics performing load balancing of DAQ CPU farm.

- Send general event information to DAQ system.

- Generate reset and trigger sequences for debugging/testing and calibration runs.

- On ECS request send monitoring counter snapshot TTC broadcast to front-end electronics.

In case the physics trigger rate gets abnormally high or data congestion occurs in the event building network, there is a potential risk of overflow in the buffers of the FE electronics or data loss in the readout network. To prevent this, ODIN controls and regulates the trigger rate according to the status of the buffers and the throttle signals. The status of the fast buffers is simulated in the ODIN whereas other slower buffers are monitored locally by the modules themselves. In the case of the locally monitored buffers, imminent overflows are signaled via the throttle network to the appropriate Readout Supervisor thanks to the Throttle Switch.

Another feature that the TFC must provide is partitionability. During normal experiment running the whole experiment is centrally and coherently controlled, while during debugging/testing and calibrations each sub-detector can use and independent branch of the TFC system and doesn't interfere with others. This is achieved by the use of the switches together with different ODINs which can independently drive configurable ensembles of front-end electronics. The programmable TFC Switch THOR carries out the task of cross-connecting a specific ODIN to the TFC network sub-branches that drive the different sub-detectors. This is the feature that permits programming different ODINs with different configurations that bear up a completely different timing, triggering and control, being even possible to connect to local trigger sources. The Throttle switch (MUNIN) performs a task similar to the THOR but for the trigger throttle signaling coming from the FE and other systems. Related to this latter module is the HUGIN (Throttle OR) which is basically a throttle signal fan-in near the FE electronics. The Figure 8.24 also shows a partitioning example: in the red color a stand-alone partition for the VELO and in the blue the rest of the experiment detectors.
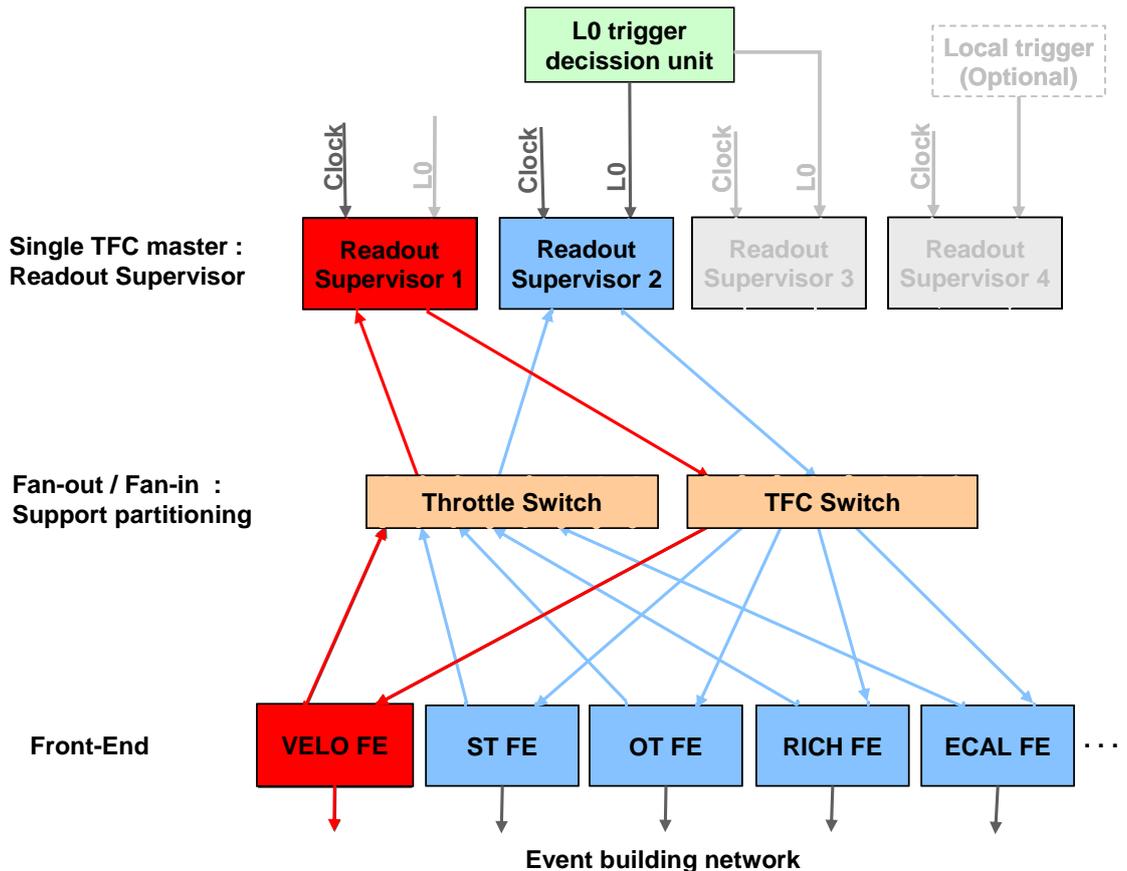
Figure 8.24: TFC architecture simplified logical scheme showing a partitioning example.

8.7.3   THE TTC SYSTEM

The TFC distribution network is based on the RD12 TTC system, a CERN standard optical network used by the four LHC experiments [42]. The TTC system distributes information optically on two serial channels:

- Channel A: transmits the LHCb L0 trigger decisions to the Front-End electronics in the form of an accept/reject signal at 40 MHz.

- Channel B: transmits framed and formatted broadcasts, including Hamming code error detection/correction. Two types of broadcasts are available: 16 bit frames which have 8 bits of user information, so called short broadcasts, and 42 bit frames so called long broadcasts. The short 8 bit broadcast command is used to distribute several important signals to the front-end and the long broadcast for sending the MEP destinations to the L1 electronics.

The two channels are time division multiplexed (TDM) and bi-phase mark encoded before being converted to an optical signal. The bi-phase signal also allows transmitting the clock with low jitter. The encoding is done in the TTC Readout Supervisor module and the electrical to optical conversion is done by the TTC Transmitter (TTCtx) modules, which have 14 high-power transmitters. The optical fan-out TTC Optical

Couplers (TTCoc) allows distributing the signal to 32 destinations, which means that one TTCtx can drive up to 448 destinations. The TTC Receiver (TTCrx) ASIC reconstructs the 40 MHz clock and converts the encoded signal into the user information. The TTCrx also provides means to adjust the timing of the TTC information in order to time-align all FE boards. A version of the TTCrx, mounted on a mezzanine (TTCrq), is also available if an easier access to the chip is needed. Another TTC module, the TTC machine interface (TTCmi), interfaces the local experiment TTC system with the accelerator timing system. It's responsible of providing the LHC clock and the LHC orbit signal.

8.7.4   INTERACTION BETWEEN THE TFC AND THE DAQ

The L1 electronics (the TELL1 and UKL1 boards) need to specify the IP address of the destination farm node when building their MEP packets. They get this information from the RS who knows which farm nodes are free. Indeed when a farm node is free, it sends a request for events to the readout supervisor. Then based on a dynamic destination assignment algorithm, it selects one farm node among the free ones.

8.8   THE LHCB ECS

The LHCb experiment consists of a huge amount of electronic devices and systems that must under the control of the ECS, and this equipment is divided into different domains depending on their activity:

- DAQ (Data Acquisition): consisting of the devices or elements taking part in the data acquisition process such as the FE electronics, the L1 electronics and the readout network.
- Detector operations (DCS) and High Voltage (HV) domains: elements such as the LV power supply system, the cooling system, the temperature monitoring devices, the gas system (for gas technology detectors), the high voltages…
- The DAI: it consists of the data acquisition infrastructure, such as crates, computers…
- The Time and Fast Control (TFC) system.
- The High Level Trigger (HLT).
- The L0 trigger system.
- The monitoring farm.
- External independent systems such as the LHC accelerator, the CERN safety system, the Detector Safety System (DSS), etc.

The Figure 8.25 depicts the LHCb ECS scope. The ECS provides an interface between the user and the equipment.
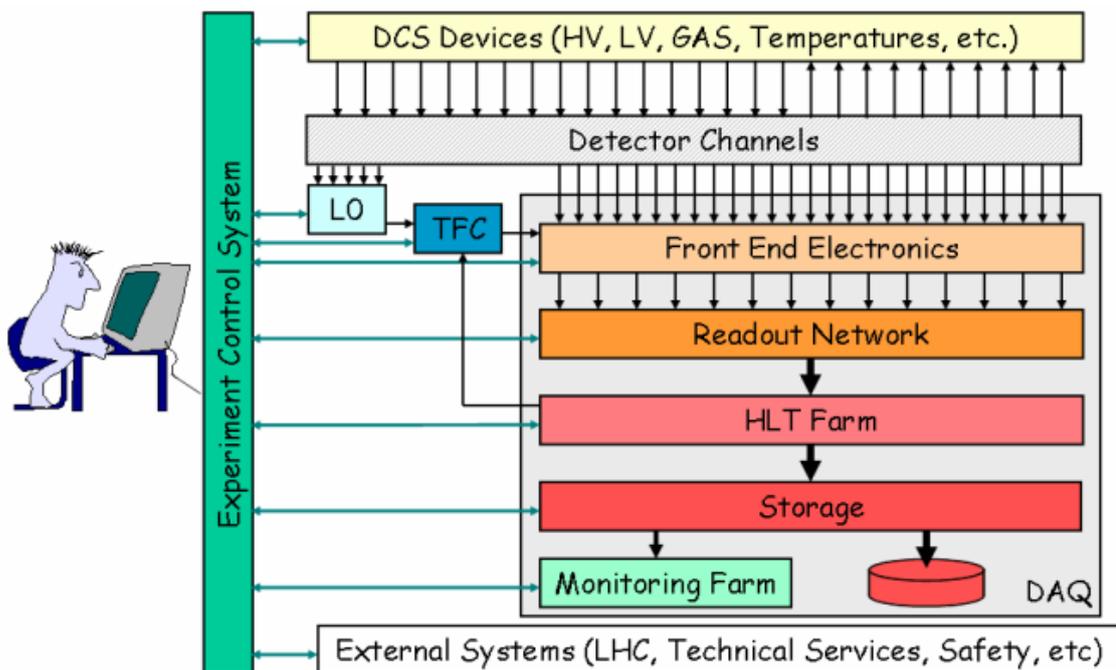
Figure 8.25: Scope of the Experiment Control System. Picture taken from [76].

As mentioned before, the control system is based in hierarchical structure because each control node has only one master node from which it receives commands to execute and a limited set of direct children to which it can send commands. A control node implements the entry point to supervise a sub-part of the detector. The root node controls the full detector.

This is a reactive system because the control nodes react to events, i.e., a command reception coming from the parent node or a state change in one of the children controllers. Event occurrences trigger two types of control flows:

1. The command flow which propagates along the tree from the operator (or control program) connected to a CU down to the lower level nodes DU acting on hardware to have some real action executed.

2. The state change flow which propagates along the tree from the hardware component that changes its status, up to the control node CUs that can take the necessary action to cope with this state change.

All control nodes publish a unique state representing the state of the sub-detector part they control. The way this state is computed is different for CU (and LU) and DU.

8.8.1 LHCB ECS ARCHITECTURE

The FSM driven hierarchical architecture approach used in the LHCb experiment for controlling the whole experiment is a tree-like mapping of the sketch in Figure 8.26. The Figure 8.27 depicts the LHCb hierarchical FSM architecture.

There is a top CU called ECS from which all activity domain CUs hang. Each of the top activity domain CUs controls a specific activity of the experiment, which can either be considered as LHCb exlcusive control domains (L0 trigger system, external independent systems such as the LHC accelerator, the CERN safety system, the Detector Safety System, etc.) a common service for all sub-detectors (TFC, HLT, storage, monitoring farm.) or sub-detectors related (DAQ, DCS, HV, DAI).
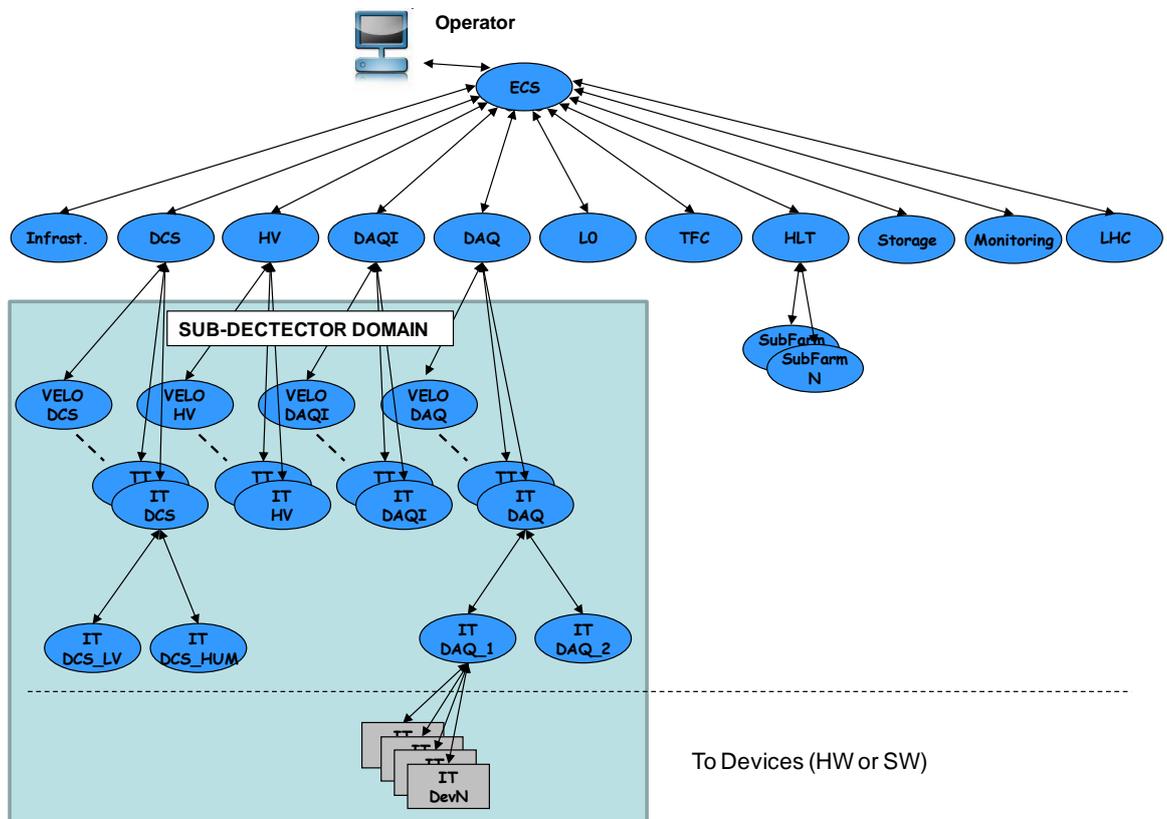
Figure 8.26: LHCb ECS architecture.

The control of each of the sub-detectors of the LHCb experiment should be split into those four separate domains each of which providing its root CU node. This domain based division common to all sub-detecotrs has eased the integration of the overall ECS system [77]. These four different domains which encompass four different types of functions necessary to run a detector are:

1. DAQ and trigger domain (DAQ): this domain includes all the components which purpose if to acquire and filter the data. The components are the FE boards, readout boards like the TELL1s, trigger boards and farm algorithms.

2. DAQ infrastructure (DAI): this domain includes all the components that are providing power, network and computing resources necessary for the data acquisition system. For example the FE, trigger and TELL1 crate power, the communication networks for control or data transportation.

3. Detector infrastructure (DCS): this domain comprises all the components related to the correct detector operation. This includes monitoring sensors related to pressure, temperature, magnetic field, and also the control of the low voltage supply to the electronics. It also contains the supervision of systems like: gas systems, cooling systems, electrical supply systems, etc.

4. HV domain (HV): this domain contains all the components that provide the high voltage power supply to the detector instruments and electronic.

In order to simplify the design of the ECS system and to exhibit a common look and feel to the shift operators a common FSM scheme with common standardized states and transitions has been defined to implement in all the CU nodes within a given domain [77]. The same states and actions are also recommended for the DUs in the same domain whenever possible. In the following chapters we will describe the implementation of these ECS domains in the ST.

While one and only one hierarchical control tree is meant to take the control of the experiment, 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 experiment gives a way to control a sub part of it, for instance, to have independent control of each sub-detector with all the necessary components to run stand alone. A way to implement this is linking the four previously mentioned domains under a sub-detector run control node as shown in Figure 8.27. This run control CU has also been standardized for every detector.

Running in stand-alone mode is not only feasible thanks to the partitionability of the FSM hierarchy but also to the partitioning of the LHCb common resources, namely: TFC, HLT, Storage and Monitoring. Whenever the detector needs to be run in this autonomous way it will need to allocate an online partition with all the necessary resources for performing full detector activities, that is, it will need a TFC partition (allocate one Odin) for driving the fast control signaling as well a slice of resources from the HLT, the 'Storage' and the 'Monitoring' for handling the acquired data (collect the data, store it and display online results). These resources are dynamically assigned to the detector whenever running in stand-alone is requested.

For easing the sub-detector commissioning, testing and calibration tasks, the sub-detector FSM top node (the run control CU) implements the needed capabilities for performing run sequences. A run sequence basically consists of a routine that performs a loop of "n" data acquisitions where one of the detector parameters is modified iteratively for each of the steps. This parameter might be the gain of the FE amplifiers, the delay of the trigger signal, the latency of the L0 pipeline, or any other one for which it would be interesting to perform a scan of the parameter in a defined range.

8.8.1.1 THE DSS SYSTEM

As ECS is software based it cannot be used as a safety system. For that purpose an independent Detector Safety System (DSS) has been built. The DSS protects the experiment against major damage to equipment or people in case the software based ECS system is not running correctly. It consists of a front-end system with stand-alone capability based on PLCs. A PVSS-II based system provides supervisory functions for the DSS and also publishes the status data to the ECS. The ECS doesn't disturb the operation of the DSS, but real-time information of the DSS status is used by the ECS to take preemptive actions before the DSS system takes an action.
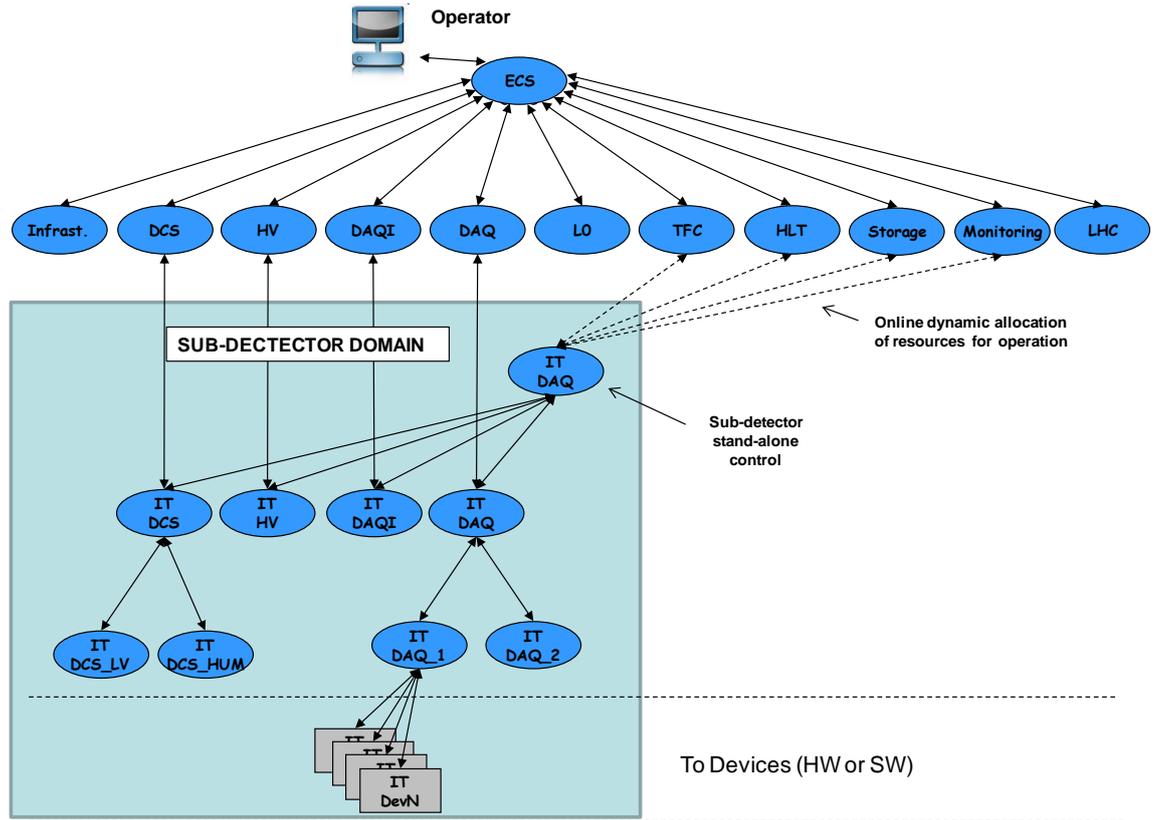
Figure 8.27: ECS architecture including the sub-detector top control domain for stand-alone operation.

8.8.1.2   ECS AUTOMATION

Automation in the ECS is one step beyond in the control system for reducing the need of human intervention in the system. The automation is provided by the FSMs as they permit sequencing actions, send commands and react to events. Automating a system implies having a good knowledge on how the detector behaves, how it works and what actions should be taken in the event of an abnormal situation.

There are two clear use cases where automation should be implemented:

1.   Start-up procedures.
2.   Error recovery.

Either at the LHCb or sub-detector level, the different elements of the system usually need to be started up in a well defined sequence. For instance, at the LHCb level, it is senseless to try to perform a data acquisition run if the TFC system is not properly set up, so in that case the control system itself must prevent the user from starting the data taking. At the sub-detector level there are obvious sequences that must be followed in or the get the detector ready for operation: first of all, the LV power supplies must be switched on; then the detector environmental monitoring tasks should run (temperatures, humidity…); finally the system is ready to be configured. These are just very few simple examples of how sequenced start up is necessary. Usually the procedures are much more complex requiring in many cases complex low level electronics procedures to be properly implemented, i.e., sometimes some registers of specific devices must be properly programmed before other FE chips can be configured. These procedures are commonly very sub-system specific so a good knowledge of the system is needed and that is why the task of defining them must usually correspond to the hardware experts. The FSM is the main tool that permits performing this automation at the system high level, while the PVSS scripting is used at the DU layer.

The other case when automation can be very helpful (and in some critical circumstances absolutely mandatory) is in the event of an abnormal functioning of the experimental equipment. When any of the devices shows a malfunctioning or reports and anomalous status, the associated DU must propagate the error status to the top nodes. In some situations, where the integrity of the detector is being risked, it might make sense that the FSM node that supervises the malfunctioning device, implements enough intelligence to automatically perform the appropriate actions, granting the safe status of the equipment. Also in case some

equipment stops running properly without posing a risk for the people, the experiment or detector itself, but for the data quality, automation can be needed.

Earlier we mentioned that the FSM allows alternate trees to take control of the FSM. For fault detection and automatic response to safety critical situations a parallel representing the topological view of the detector seems to be more appropriate. This tree can be used not just for sending automatic 'Emergency Commands' but also for visualization to help in fault detection. This parallel FSM tree would implement all the intelligence to perform the appropriate automatic actions in order to always keep the sub-detector in a safe state as well as provide detailed info of the detector status without the need of permanent human supervision.

8.8.2   LHCb ECS DEPLOYMENT

The use of common tools such as PVSS and the framework components has been the basis for accomplishing the task of building coherent PVSS based control system. But, in order to integrate such a large control system, it is not enough.

On one hand, the LHCb ECS is a huge hierarchical distributed control system. It consists of many control nodes running in different computers and different PVSS systems that need to run as a one. On the other hand, there are many people and groups involved in the development of the system distributed through many places around the world. While the development and maintenance of the sub-detector specific domains (DAQ, DCS, DAI and HV) is responsibility of the each sub-detector team, for the LCHb common part the LHCb online group is the responsible.

In order to overcome these difficulties, a set of guidelines for integrating the control software and the hardware equipment of the different sub-detector had to be defined [78]. One of the rules has already been mentioned, the CU definition of the top DCS, DAQ, DAI and HV domains is standardized for everybody. The same applies for the detector run control CU. The other important guideline is the use of a common naming convention.

Naming conventions are intrinsically needed for communication. When setting up PCs and PVSS projects for LHCb, naming conventions must be respected in order to coherently integrate the different sub-detectors and sub-systems so one can uniquely identify applications, systems, FSM nodes, FSM types or files:

1. Sub-System Names: a list of names has been defined for the sub-systems. There is a unique name that identifies each sub-detector as well as a name for each of the central sub-systems. For instance, for the Inner Tracker and Tracker Turicensis the corresponding names are IT and TT.

2. Domain names: the four standard domain names for each detector are DAQ (DAQ and Trigger domain), DCS (Detector infrastructure), DAI (Data Acquisition Infrastructure) and HV (High Voltage).

3. Computer names: computer names (i.e. network name) of control PCs follow a convention that takes into account the sub-system name, the geographical position and the operating system.

4. Other Ethernet Equipment Names (Credit-Card PC, CAEN, etc.): their name takes into account the sub-system name, the equipment type and geographical position.

5. PVSS System Names & Project Names: a PVSS project must be given a name at creation time, and a system name which is used (in conjunction with a system number) to identify it in a distributed system. The naming convention includes the sub-system name, the name of the top most CU domains and geographical division.

6. PVSS System Numbers: each PVSS project is required to have a unique system number in order to allow connection to the distributed system for the LHCb ECS. There is a range of system numbers attributed to each sub-system. Each sub-detector has 10 numbers at its disposal so up to 10 systems could be built per detector for control purposes.

7. FSM Naming Conventions: a way to build the FSM names for sub-detector hierarchy has been defined in such a way that the name is meaningful enough to have an idea of the part of the detector covered by the CU domain.

8. PVSS Panel Names: the name of this panel should be the same as the FSM object.

Together with these guidelines for building up the PVSS ECS system based in a naming convention, there are some rules on how to use the filesystem infrastructure where the PVSS projects will be installed. The software must be installed in specific places using a naming convention for the folders and file names.

8.8.3   ALARM HANDLING

PVSS has an extensive integrated alarm concept for safe, reliable monitoring of sensitive equipment:

- Multiuser acknowledgement for multiple workstations.
- Alarms via speech output, pager, SMS or e-mail.
- Definition of shared alarm properties (priorities according to importance, type of acknowledgement) in alarm classes.
- Automatic sum alarms.

The aim of the alarm handling system is that all problems related to the detector equipment should be reported to the operator using one single tool: The Stan99dard PVSS/FW 'Alarm Screen'. The Alarm Screen can be configured in different ways and in particular can be used centrally to report all messages from all sub-systems, or locally by each sub-detector to view only their messages.

Alarms can have three severities: *WARNING*, *ERROR* and *FATAL*. All messages reported in the 'Alarm Screen' imply that an action should be taken by the operator (most of the time by contacting the expert). The meaning of the different severities is as follows:

- *FATAL*: action should be taken immediately otherwise there could be damage to the detector (HV, Gas, Cooling problems for example).

- *ERROR*: action should be taken, but there is no immediate danger.

- *WARNING*: action may need to be taken, the problem needs to be followed up (a temperature or humidity level is starting to raise, for example)

While these rules are quite obvious for power supplies, cooling systems, temperature monitoring, etc. They are less obvious for errors happening in the DAQ area. There the following definition would apply:

- *ERROR*: a component failed to configure, the message should indicate the cause or during a "run" a problem is detected that either prevents from running or affects permanently the quality of the data. It should also trigger the FSM to go to ERROR state.

- *WARNING*: problems that do not prevent from running and don't affect permanently the quality of the data but should be followed up. It should not trigger the FSM to go to ERROR

In general the idea is that whenever the FSM goes into ERROR state, the operator should find in the 'Alarm Screen' an explanation message from the source of the problem, and then alarm should disappear from the screen as soon as the problem is gone.

CHAPTER 9

THE ST EXPERIMENT CONTROL SYSTEM

ABSTRACT

This chapter provides a general overview of how the ECS for the ST will be developed following a common approach for the IT and TT.

"Exígete mucho a ti mismo y espera poco de los demás. Así te ahorrarás disgustos"

Confucio

9.1 OVERVIEW

The Silicon Tracker (ST) detector control and readout infrastructure consists of several thousands of elements between devices and sensors. They are placed in the cavern and need to be controlled and monitored. In addition there is equipment placed off the experimental area used for readout and system operations. The Experiment Control System (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 ST ECS is a highly hierarchical Finite State Machines distributed control system. The controls hierarchy is structured in a functional and a geographical division basis and keeps a similar layout for IT and TT detectors. A CB was developed in order to provide among other functionalities, slow control (I^2C interfaces, PT1000 temperature readings…) and fast control signaling for the ST front-end (FE) electronics using the LHCb SPECS electronics interface system. Other LHCb specific hardware interfaces such as the CCPC are used for remote control and monitoring, this is the case for the TELL1. In the middleware layer specific software interfaces such DIM are used as well as industry standards such as CAN or OPC.

9.2 THE ST CONTROL SYSTEM OUTLINE

The ST detector, despite of its physical size, consists of a huge amount of readout channels due its high spatial resolution. For this reason the amount of devices to be controlled and monitored is about several thousands of FE elements. The ST consists physically of two sub-detectors, the IT and the TT, each of which runs its own electronics infrastructure and control software. The two of them are same technology detectors and use the same readout, control and operations hardware but with a different partitioning of the comprising elements. Despite this, and thanks to a close care design of the readout hardware based upon a well defined hardware partitioning and to the development of a common CB, a closely similar control system could be developed for both detectors.

The ST ECS is a high performance system that must ensure the safe and reliable operation and control of the detector. Contrary to the approach followed in other LHC experiments, the ECS has to take care not only of the DCS electronics (monitoring of temperatures, humidity, voltages, current…) but also of the DAQ equipment, i.e., configuring and monitoring the FE and Back-End (BE) TELL1 electronics. In order to be able to configure and monitor such large systems, a high degree of parallelism is necessary. The control system is built as a hierarchy of FSM sub-domains distributed over several computers running in different platforms using the SMI++ toolkit and the PVSS SCADA.

For each of the sub-detectors a closely similar hierarchical structure has been defined. Each LHCb detector has to implement four domains, namely DAQ, DCS, DAI and HV. Below these domains a FSM structure based on the physical geometry of the detector is deployed. The FSM hierarchy has been kept similar for IT TT despite of the different detector geometries and thus the different the grouping of the detector elements. The ST ECS not only controls the detector hardware but also interfaces to the LHCb central ECS, which during normal running will take control of the detector. The ECS has the responsibility for providing for the automation of standard procedures and for the automatic recovery from error conditions in a hierarchical fashion.

The use of the different hardware and software solutions adopted for the ECS communications such us SPECS, CCPC, DIM and OPC will be briefly described.

9.3 THE ST HARDWARE AND MIDDLEWARE STRUCTURE

Already at the early stages of the LHCb ECS design a set of guidelines were defined in order to ensure that all the detectors used common ways to interface their FE and BE electronics and operations equipment. While standard industry solutions were usually preferred, such as CAEN HV and WIENER LV power supplies, custom control hardware developments were needed to interface the FE and BE equipment, on one hand due to the high radiation environment in the cavern and in the other hand due to the high complexity of the readout system. This hardware had to be put later on under the scope of the ECS. The ST ECS is required to control ~270 000 readout channels. Table 9.1 shows the amount of devices that comprise the ST and need to be configured, as well as the number of parameters that need to be monitored. In both tables the physical location of the element is shown.

	IT	TT	Location
Beetle	1008	1120	[Hybrids]
GOL	1008	1120	[DB]
SPECS	48	48	[CB]
DCU	336+72	280+36	[DB]+[CB]
Delay25	24	24	[CB]
TTCrx	24	24	[CB]
LVR	1008+72	780+72	[SB]+[CB]
CCPC-TELL1	42	48	[Barrack crate]
CAEN ch	84		[Barrack crate]
MARATON ch	48	48	[Barrack crate]
Cooling station	1	1	[Barrack]

	IT	TT	Location
PT10001	336+24+12+12	280+24+12+12	[Hybris]+[SB]+[CB]+[DetBox]
HMX20002	12	12	[DetBox]
DAQ-FE status			[DB DCU]+[CB I/O]
LVR voltage	336	280	[DB DCU]
LVR INH	420	376	[CB I/O]
LVR OCM	672+336	560+280	[DB DCU]+[CB I/O]
MARATON V	48	48	[Barrack crate]
MARATON I	48	48	[Barrack crate]
CAEN V	84		[Barrack crate]
CAEN I	84		[Barrack crate]
CCPC-TELL1	42	48	[Barrack crate]

Table 9.1: Summary of devices to be configured (left) and parameters to be monitored (right).

The electronic devices in the FE, that is, hybrids, DBs and SB backplanes 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, as well as the CCPC. Whilst the SPECS is intended to interface to the FE, the CCPC is used for the TELL1 BE electronics and both use the DIM protocol as middleware. The CAEN HV and MARATON LV power supplies are commercial equipment that provides an OPC server as middleware interface and use TCP/IP over Ethernet to connect to the ECS as transport and physical protocols. The cooling station is controlled using MODBUS and the racks and crates in the barrack, where the DAQ and detector BE electronics is housed use the ELMB and CANBus and CANOpen standards. The scheme of Figure 9.1 summarizes the overall architecture of the Silicon Tracker. The TFC is and external system that provides the fast signaling for the DAQ electronics. The DSS is a completely separate highly reliable and real-time response system whose main aims are to protect the equipment and prevent situations leading to severity alarms. A way to publish the DSS information to the sub-detectors had been foreseen so the control system could react in consequence.

In order to structure the data readout, a partitioning of the TELL1 DAQ links was early defined during the detector design. This grouping was extended also to the LV, HV, FE and control electronics during the development process of the hardware so that the definition of a partition of devices matches in all domains and hence a highly coherent control system has been built.

The CB plays an important role in the ECS system acting as a bridge between the FE electronics and the ECS, apart from performing specific control and monitoring tasks. It also plays a role in the detector partitioning.

9.3.1 THE L0 AND L1 HARDWARE

The Beetle chips are mounted together with passive electronic components on the so-called hybrid, which consists of 3 Beetles for the IT and 4 Beetles for the TT detector. The readout of all silicon sensors is performed by 336 hybrids in the IT and 280 in the TT.

For each hybrid there is a corresponding Digitizer Board. In the DB, the signals of each Beetle are sampled and a CERN GOL serializer device retransmits it to the TELL1 through a 2.5 Gbps optical link. The DBs are installed together with the Control Board in the Service Box out of the detector acceptance. Each IT SB houses 16 DBs while the TT SB houses 12, which makes a total of 48 Beetle chips that both types of SBs can handle.

---

1 The PT1000 temperature sensors are read out using the DCU chip.

2 HMX2000 is a MEMS humidity sensor from 'Hygrometrix Inc'.
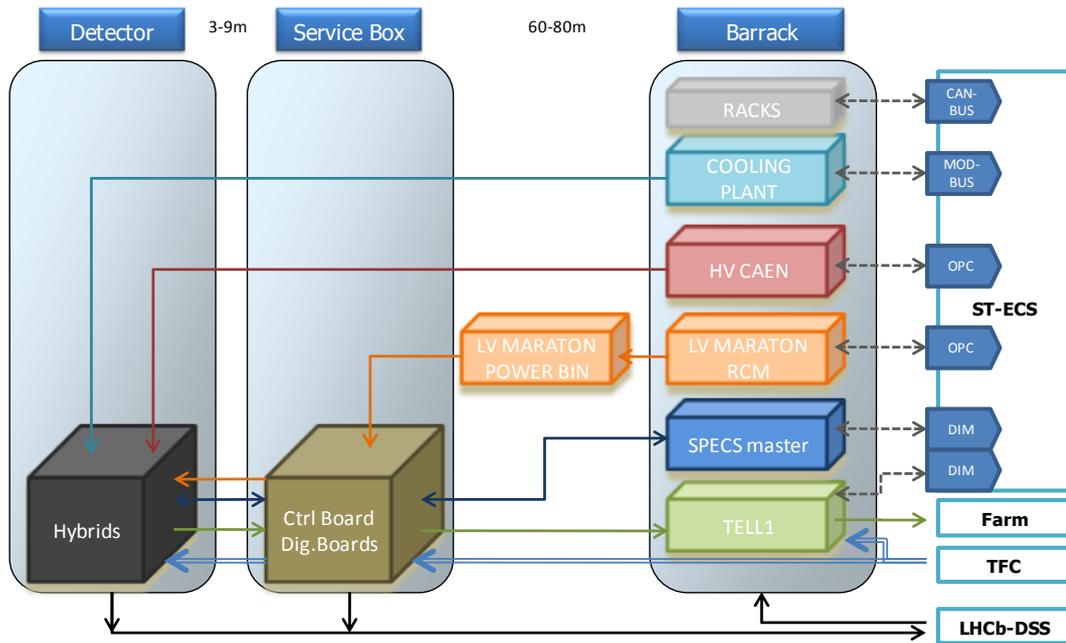
Figure 9.1: Silicon Tracker equipment scheme.

The Control Board performs a twofold task: delivers the fast control signaling to the FE electronics and handles all the slow control tasks in the SB. The TTCrx chip receives TTC signals that are further decoded and delivered to the hybrids and DBs via the backplane. In addition it interfaces the SB electronics with the ECS through the SPECS system providing: eight I^2C buses for the Beetles, GOLS and DCUs; ADC lines for monitoring of temperature and humidity signals; and I/O multipurpose lines for LVR (Low Voltage Regulators) control, devices reset signaling, etc. The CB is common for IT and TT service boxes. For the CB the electronics downwards the hybrid is exactly the same in both detectors, just the commands that need to be sent differ slightly due to the different hardware partitioning, so this is handled by the ECS software.

The backplane in the service box is the physical interface between CB and DBs. The backplane is the element that implements the physical partitioning at the FE level and that permits using a common CB. It redistributes the fast signals delivered by the CB to the DBs and hybrids. Besides it houses the LVR needed to power the FE electronics arranged in sets according to hardware partitioning.

The ST is deeply populated with PT1000 sensors. They are placed one in each hybrid and they are read out by the DB DCU. The PT1000 sensors used for the detector box, backplane and CB temperature monitoring are read out by the CB. The CB also performs the readout of the radiation hard HMX2000 humidity sensors placed in the detector boxes.

In the back-end side of the readout electronics sit the TELL1 boards; a specific acquisition board handling L0 accepted data before transmission to the DAQ. They are interfaced by a CCPC placed in-board.

A DIM server called 'SpecsServer', which is running in the computer where the SPECS master cards are installed implements the middleware interface to the ECS for the FE electronics. For each TELL1 CCPCs there is also a DIM based 'ccserver' running locally in the board which performs the ECS tasks.

9.3.2   THE LV PS EQUIPMENT

The ST LV power system consists of Wiener MARATON power supplies that provide bulk power to the regulators placed in the backplane. The MARATON system consists of several pieces, from which the RCM, placed in the counting room far from the radiation environment, is the controller module that connects to the control system. The MARATON bin located in the cavern houses the LV modules which supply the power to the regulators. The regulators are controlled (on/off via *INHIBIT*) and monitored (over current status via *OCM* signal) by the CB, and they are arranged in sets such the each partition is independent from the other, as in Figure 9.2.

The Wiener PS provides an OPC server for interfacing to the control system.
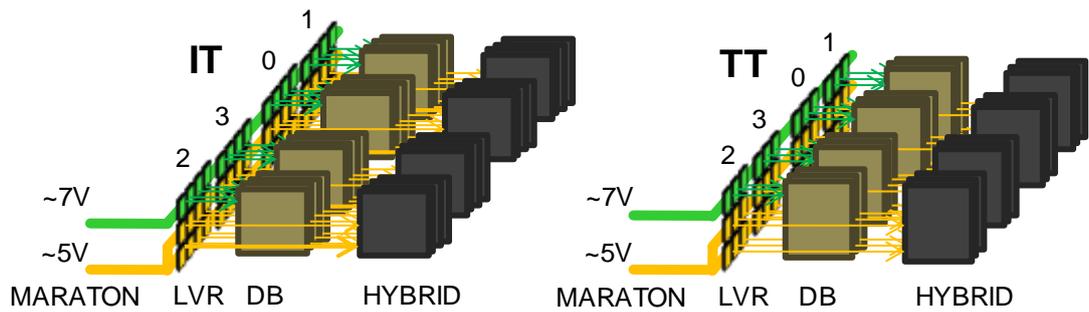
Figure 9.2: LV regulators partitioning in the SB. Two channels of 7 and 5 V provide power to the Low Voltage Regulators (LVR), which are arranged in four sets or partitions, numbered as in the figure. Each set provides the power to 4 DBs and Hybrids for the IT and 3 in the case of the TT.

9.3.3 THE HV PS EQUIPMENT

The HV bias for the sensors is provided by multichannel CAEN power modules. This is commercial equipment that offers a standard OPC server for connecting to the PVSS SCADA. A passive patch-panel splits off the HV channels into several depending on the partitioning granularity as in the figure 9.3. In the IT all the 84 CAEN channels are split into four giving a total of 336. In the TT, 64 for channels are split into 3 and 88 are kept without splitting giving a total of 280. The reason for the un-split channels is that there are readout modules that need to be biased by a single CAEN channel due to the expected high leakage current after irradiation. Jumpers permit the manual disconnection of the HV for split off channels. Figure 9.3 summarizes the given explanations.
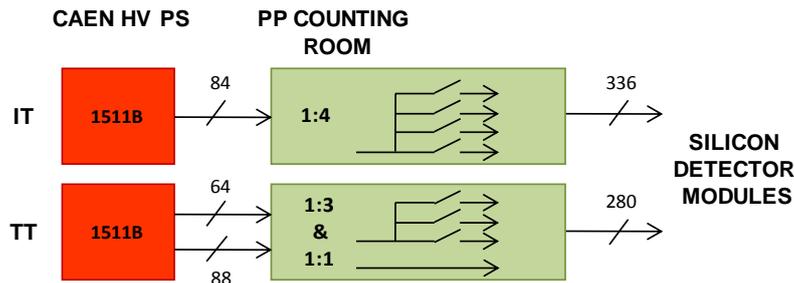
Figure 9.3: HV distribution in the ST.

9.3.4 COOLING SYSTEM

Two independent C_6F_{14} cooling plants provide the cooling to the IT and TT detectors. Each plant is controlled by a PLC, which is connected to a gateway PLC that offers a MODBUS TCP interface for connecting to the ECS. A PVSS manager running in the CERN technical network provides high level inter-communication between the PLC and the LHCb control system, as shown in the Figure 9.4. Relevant information about the cooling unit status and the measured physical parameters is available on the PVSS side and only some settings can be accessed by the authorized operators.
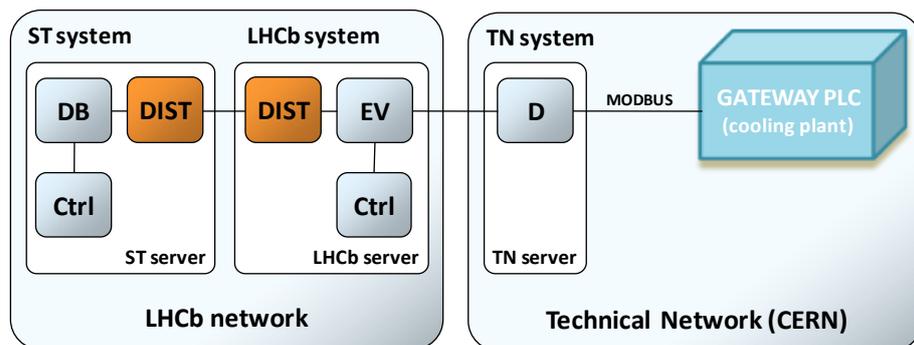
Figure 9.4: Cooling plant remote control.

9.3.5 MIDDLEWARE

OPC and DIM are the most popular middleware interfaces used in LHCb. However, other ones such as MODBUS and DIP are sporadically used.

DIM (Distributed Information Management) is a high level client/server communication protocol for distributed and mixed environments that provides a network transparent inter-process communication layer. The basic concept in the DIM approach is the concept of service, a set of data recognized by a name, which is provided by the server to the client. In order to allow for transparency (i.e., a client does not need to know where a server is running) as well as to allow for easy recovery from crashes and migration of servers, a name server known as DNS was introduced. Servers publish their services by registering them with the name server and the clients subscribe to services by asking to the name server which server provides the service and then contacting the server directly, providing the type of service and the type of update as parameters. The name server keeps an up-to-date directory of all the servers and services available in the system. The DIM protocol is used in LHCb for the remote control of the CCPC and SPECS devices. Figure 9.5 depicts how the 'SpecsServer' processes are deployed in one of the IT control computers. There is a 'SpecsServer' process per SPECS Master running in the same computer where the Master PCI boards are installed. The DNS and the PVSS DIM client which provides connection to the ECS are also running in the same computer. In Figure 9.6 we see how the deployment is like for the TELL1 'ccserver' processes. In this case, there is a 'ccserver' per TELL1 board running in the CCPC. They publish the services via the DNS that is running in the same computer as the PVSS DIM client.

OPC (OLE for Process Control) is used in the ST for interfacing the CAEN and Wiener MARATON power supplies. The 'WienerOPCServer' and the 'CaenOPCServer' run both in the same 'Windows' computer. PVSS provides a manager to inter-communicate with the OPC servers.

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. DIP is essentially used for publishing DSS information. Figure 9.8 shows how one of the IT projects retrieves the DSS information from a scattered dip manager running in a "gateway" computer. This "gateway" is the only mean to access the DSS information publishes somewhere in the CERN internal technical network.
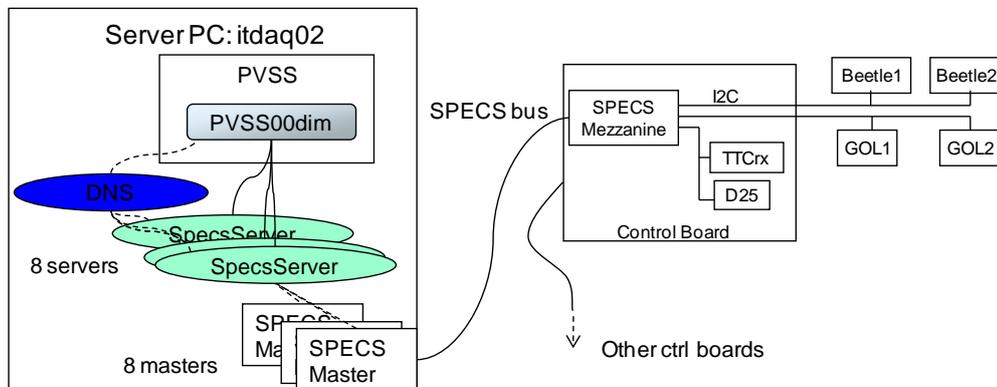
Figure 9.5: deployment of DIM the 'SpecsServer' processes.
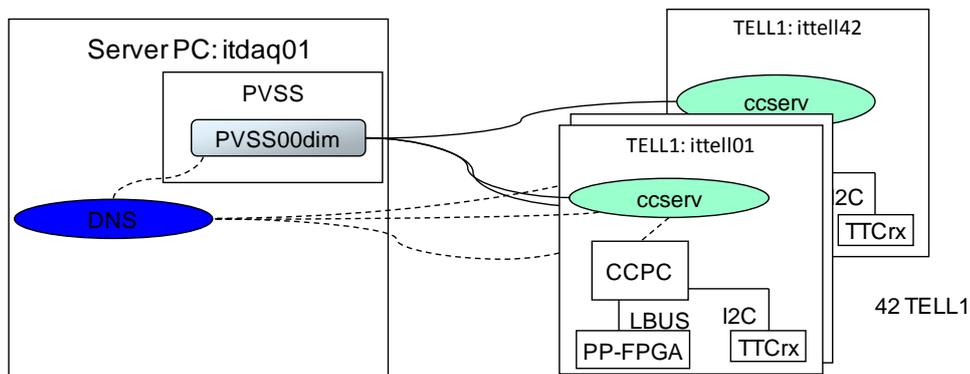
Figure 9.6: distribution of DIM 'ccserver' processes.
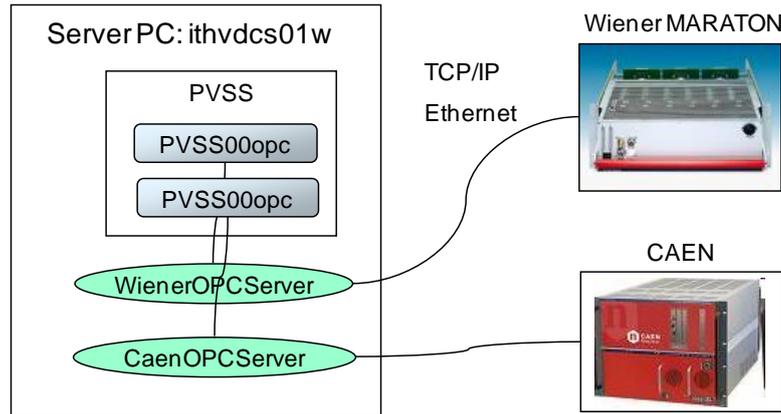
Figure 9.7: OPC middleware in the ST.
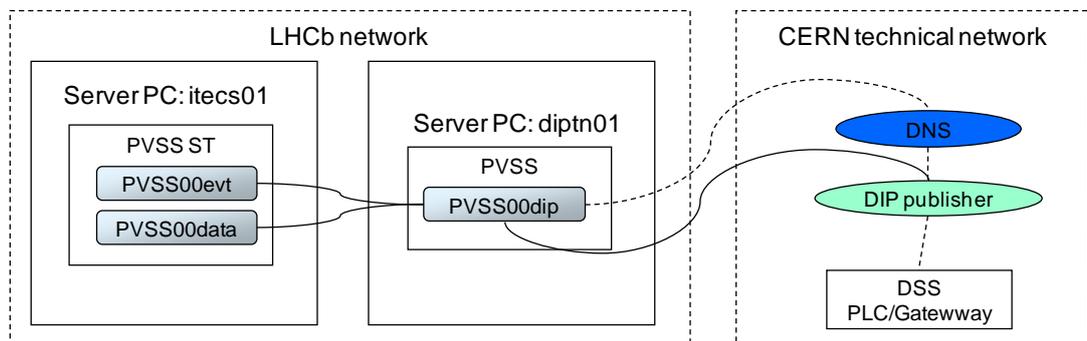
Figure 9.8: DIP deployment in LHCb.

Modbus is a serial communications protocol mainly used in programmable logic controllers (PLCs). It has become a de facto standard communications protocol in industry, and is now the most commonly available means of connecting industrial electronic devices to supervisory computers. The main reasons for the extensive use of Modbus over other communications protocols are: it is openly published and royalty-free; relatively easy industrial network to deploy; it moves raw bits or words without placing many restrictions on vendors. It is used for the cooling plant control and the deployment scheme has already been shown in Figure 9.4.

9.3.6 PARTITIONING OF THE SYSTEM

Since the early stages of the ST electronics design the partitioning concept was defined. Based on the original data readout partitioning, the concept was extended to the FE electronics design and to the ECS.

The original DAQ readout partitioning defines minimum groups of 12 optical readout links, that is, an ensemble of four or three DBs in IT and TT respectively, as this corresponds to a cable bundle transporting the data of 12 beetles (see Figure 9.9). Each TELL1 receives the data of two optical fiber bundles. This is the so-called partition in the ECS. The LV power system granularity also follows this partitioning, so one can switch ON/OFF single partitions of DBs and individual hybrids. The figure shows the assignment of partitions in the FE service boxes. Each group of hybrids belonging to the same partition is HV biased by the same channel in the case of shared high voltage channel as explained before, and individual control can be achieved either manually using the jumpers in the patch-panel for the shared channels or acting on the channel for the non shared ones. The CB in the service box provides and independent I²C bus for each partition of DBs and hybrids. As a result, the maximum granularity level at which one can has full control from the ECS is at the partition level, for example, one could not power off individual DBs but partitions.

The service box is the next top level in the grouping concept. Each SB is comprised of 4 partitions of FE devices (hybrids and DBs), the CB providing the slow and fast control and the backplane interconnecting all boards. Each SB has one input connection from the ECS and the TFC and all the signal propagation delays for the FE hybrids attached to it are the same as the cable lengths are equal.
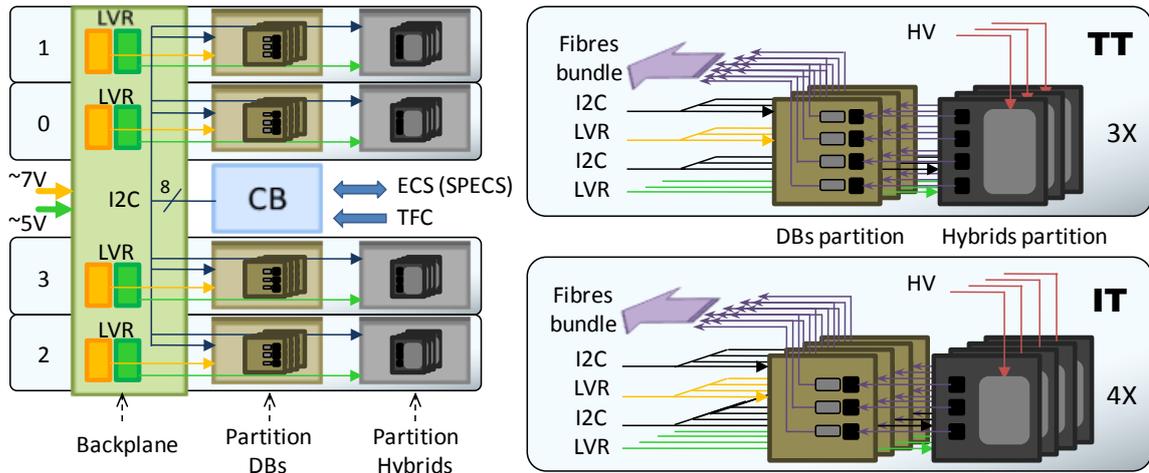
Figure 9.9: IT and TT FE partitioning.

From the ECS point of view, higher levels of abstraction are needed above the SB level. Different grouping concepts are possible so it was decided to develop a grouping that followed as much as possible the physical distribution of the hardware. The grouping from the SB upwards follows the grouping of the readout electronics links. In the IT there is a nice matching between readout grouping and control links at the half-station layer as it corresponds to one SPECS bus. In the TT that matching does not exist as the SPECS control link of the B region is always shared with one of the others regions. The readout grouping was done with the aim of equalizing the difference in propagation delay of the readout signals due to the time of flight of the traversing particles. The SPECS control links deployment was made according to the geographical placement of the service boxes in four stacks of six, each stack in one of the quadrants of the detector. This mismatching has as consequence that one cannot power off the power supplies of single regions because the CBs belonging to the same SPECS link are connected in a daisy chain (each CB implements some LVDS repeaters to retransmit the signals) so they need to be always powered on. As a result, the LV power supplies control is not partitioned as in the IT where individual power control of the half-stations is possible. Figure 9.10 synthesizes the grouping hierarchy for IT and TT, showing between parentheses the number of elements in each layer and also including the TELL1s' grouping, which is different for both detectors.

These differences in the grouping are easily handled at the ECS level thanks to the flexibility the hierarchical FSM toolkit provides.
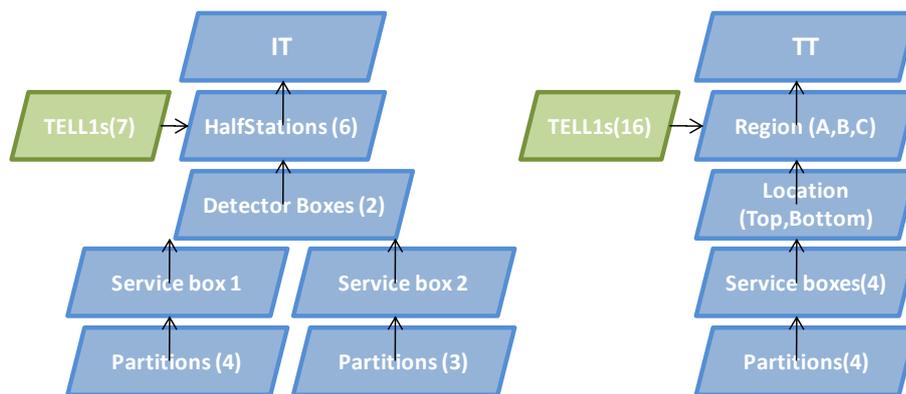
Figure 9.10: Grouping of the ST equipment in different ECS layers.

9.3.7 HARDWARE ADDRESSING

Every device has to be uniquely identified by a hardware address in the control system. For the TELL1 CCPC this is achieved assigning a unique IP address to each of them. The same applies to the cooling plant gateway PLC, the CAEN and the Wiener MARATON PS.

For the FE devices controlled via SPECS there is also a unique space address. It consists of the following items: the computer name where the SPECS masters are installed; the SPECS master ID within the computer; the SPECS slave ID; I^2C bus number (in case of I^2C devices); I^2C address.

One has to keep in mind the limitation on the chips space address (see Figure 9.11). On the other hand, there is a limitation on the I^2C space address given by the standard as explained in a former chapter. As a result, only the address range from '8' to '119' is free for use.

A6   A5   A4   A3   A2   A1   A0

7 Bit addressing						
6 Bit addressing						X
6 Bit addressing						X
4 Bit addressing				X	X	X
4 Bit addressing				X	X	X

Beetle: all seven I2C address bits can be used. Its programming scheme only occupies a single address.
GOL: occupies two I2C addresses, the 1st address for the register pointer and the 2nd for the register data value.
TTCrx: uses two addresses in the same way as the GOL

DCU: it the 3 least significant bits to distinguish between the 8 internal registers. Just bits A6 to A3 of free address space.
D25: occupies 8 addresses as the DCU.

Figure 9.11: Devices space addressing.

Four of long distance I^2C buses delivered by the CB are dedicated to the hybrids and four to the DBs. The internal one is for the CB inside (TTCrx, Delay25 and DCU). Keeping in mind the address restrictions and the number of I^2C buses, the I^2C addressing scheme of Figure 9.12 was defined.

Long Multiplexed I2C addressing

A6 A5	A4	A3	A2 A1	A0
BOARD-ID	1	0	CHIP-ID	0

Beetle Chip I2C addressing

A6 A5	A4	A3	A2 A1	A0
BOARD-ID	1	0	CHIP-ID	X

GOL I2C addressing. 'X' bit is used internally

A6 A5	A4	A3	A2	A1	A0
BOARD-ID	0	1	X	X	X

DCU I2C addressing. 'X' bits are used internally

CB internal bus addressing

A6	A5	A4	A3	A2	A1	A0
0	1	0	0	X	X	X

D25 chip I2C addressing. 'X' bits are used internally

A6	A5	A4	A3	A2	A1	A0
1	0	1	1	1	1	X

TTCrq I2C addressing. 'X' bits is used internally

A6	A5	A4	A3	A2	A1	A0
1	1	0	0	X	X	X

DCU chip I2C addressing. 'X' bits are used internally

Figure 9.12: Front-end addressing scheme.

The addressing of the CB internal devices poses not much complication since the number of chips in the bus is just three, and the assignment is shown in the figure.

9.3.8   THE DSS SYSTEM

Although it is not foreseen that the time response of any corrective action to be taken in the ST hardware has to be fast, under some circumstances it has to be clearly limited in time. The ST ECS is not a real-time system, even though it can cope with all the control tasks during normal running. In fact, in principle it should be able to respond in "pseudo" real time to any event. Under normal circumstances, the ECS will take appropriate actions in case of problems. For example, an abnormal temperature in the detector will cause the generation of error messages to the control operator so that he/she can intervene or the ECS can perform automatic actions. As the real time and complete reliability of the ECS cannot be assured due to the complexity and size of the distributed control, a parallel hardware oriented system called Detector Safety System (DSS) will take the appropriate action to protect the hardware against potential causes of damage, either caused by a software failure or by a missing human intervention.

The Detector Safety System (DSS) will take action to protect the hardware from human error or component failure. Personnel safety is taken care of by the CERN Safety System (CSS). DSS will check a few critical parameters with dedicated sensors. Commercial thermostats tested for magnetic field tolerance are used for preventing potential overheating inside the detector boxes and of the service boxes cooling blocks. In the TT, commercial gas flow switches are used to verify that N_2 gas is flowing into the box and water flow switches

are used to check that the mixed water is flowing in cooling water system. The cooling station just sends directly signals to the DSS in case it detects any internal malfunctioning.

The logic output of all these sensors will be routed to a Programmable Logic Controller (PLC). This is a diskless compact computer, widespread in industrial and safety applications, designed for robustness and reliability. PLCs operate in a closed-loop control cycle. They read input sensors, take a simple decision, then act on output devices (such as switching off power supplies). Information will also be exchanged with the Experiment Control System (ECS) via DIP to inform about the status of the experiment and in some cases even take preventive actions such as ramping down the HV before a sudden switch off.

9.4   ST CONTROL SYSTEM DESIGN

The ST control system follows the concept of Distributed Hierarchical Finite States Machines (DHFSM). With the aim of facilitating the integration of the different detectors control system, the system has been designed following the general guidelines defined in [77]. The PVSS SCADA and the JCOP framework are the fundamental blocks used for the control system design and development. Within the JCOP framework the SMI++ toolkit is the essential element that permits implementing DHFSMs. Hereafter we describe the reasons behind using the DHFSM concept [62].

Why FSMs? The FSMs are very well suited for this application for many reasons. Modeling 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 behaviors. They also reduce the number of parameters exported and managed in the SCADA supervisory layer. The SMI++ permits using two types of objects: abstract objects and physical objects. Abstract objects implement logic behavior, they have a list of allowed states, in each state a list of allowed actions and when an action gets triggered (either by the reception of a command or by a state change of another object) they execute instructions like sending commands to other objects or testing the state of other objects. The behavior of the object is coded using a very simple language called SML. Physical objects implement the interface to real devices, they also have a list of allowed states, and in each state a list of allowed actions, but when they receive a command they have to act on the device they model, and when the device's data changes they eventually change their state. Physical objects can be coded in any language (C, C++, or using PVSS scripts).

Why hierarchical? Hierarchical Control is needed because the data gathered by the different machines have to be summarized in order to present a simplified but coherent view to the users and other FSMs. Grouping in a hierarchical way also permits decentralized decisions, each sub-system should be capable of taking local decisions since a centralized decision engine would be a bottleneck.

Why distributed? Distribution and parallelism is needed due to the large number of devices and input/output channels to configure and monitor. The acquisition and monitoring of the data has to be done in parallel and distributed over several machines running in different platforms.

The ECS has also to provide intuitive User Interfaces (UI). Since the operators will not be control system experts it is important that the user interfaces provide a uniform and coherent view of the system and are easy to use. Apart from this, the ECS also offers the possibility to store data, handles alerts and provides standard interfaces to connect to hardware.

A high degree of parallel processing is needed not only for the FSMs, which can be running on one or several computers but also for the rest of software processes that carry out the control tasks such as the DIM and OPC servers.

The JCOP adopted a hierarchical, tree-like, structure to represent the structure of sub-detectors, sub-systems and hardware components. This tree is composed of three types of nodes, a Control Unit (CU), a Logical Unit (LU) and a Device Unit (DU) that serve as basic building blocks for the entire hierarchical control system. The CU and LU model and control the sub-tree below it and the device unit "drives" a device. The DUs are the representation of the SMI++ physical objects while the CUs and LUs correspond to the abstract objects.

The ST control hierarchy is designed such it 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 in the other hand it can be driven by the top LHCb overall ECS. For that reason the control is divided into four domains that every sub-detector has to implement, namely DAQ, DCS, DAI and HV. This way, the LHCb top control node can take control of the detector via these standard domains. It is also possible to control the detector in stand-alone mode from an IT top node that links these 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 or a parallel hierarchy that performs automatic actions in the event of an error. Another application is the safety tree that has been developed to ensure the continuous running of the detector in a safe way.

9.4.1 CONTROL DOMAINS

It has been found useful to split the LHCb detectors ECS systems into four different domains which encompass four different types of functions necessary to run a detector. This has eased the integration of all the detectors into the LHCb overall ECS system. The domains are the DAQ, DAI, DCS and HV, which are described in the following paragraphs.

In order to simplify the design of the ECS system and to exhibit a common look and feel to the shift operators a common FSM scheme with common standardized states and transitions has been defined to implement in all the CU nodes within a given domain. The same states and actions were also recommended for the DUs in the same domain whenever possible. The following paragraphs describe the FSM proper to the CUs, LUs or DUs of each of the four ECS domains.

9.4.1.1 DAQ DOMAIN DEFINITION

The data acquisition and trigger domain (DAQ) includes all the components which purpose is to acquire and filter the data.

The components included in this domain are the front-end boards (DBs and hybrids), readout boards like the TELL1s, trigger boards (CB) and farm algorithms.

Some transitions are automatic, i.e. they are not the result of a command. For a DU they are the result of monitoring device parameters, for a CU they are the result of a state change in one of their children. This is the case for the automatic transitions to *ERROR* and *UNKNOWN*.

Before it can be used a DAQ and trigger component should be configured with the proper settings. The configuration settings depend on the desired running mode. The devices have to be reconfigured when changing running mode, for example from *PHYSICS* to *CALIBRATION*. The *Configure* command will carry a parameter *RUN_TYPE*, specifying the desired running mode.

The *Start* command can also be used to apply some settings to the device. These settings should not depend on the running mode and applying them should be fast (less than one second), for example resetting the TTCrx counters in the CB.

Figure 9.13: DAQ domain states.

If the transition *NOT_READY* to *READY* to configure the system is long (more than a few seconds) an intermediate *CONFIGURING* state should be published (see figure) until the transition is finished. The control node will automatically transit out of this *CONFIGURING* state either by detecting that the device is *READY* (for a DU) or when all children are *READY* (for a CU). DUs should implement a Time-out and publish an *ERROR* state in case the transition is too long.

The meaning of the states for DUs and CUs is described in the tables.

UNKNOWN	It is not possible to communicate with the device. We have no information about it and we cannot apply any command to it. For example the "driver" process is dead.
NOT_READY	The device is under the control of the ECS but it needs to be configured before it can be used.
READY	The device is ready to take data. The "Start" command can, for example, clear some counters.
RUNNING	The device is fully configured and taking data. Its configuration cannot be changed while in this state. It should be stopped if we need to change is configuration.
ERROR	The device has detected an error which means that it is not able to take valid data.

Table 9.2: DAQ states description for DUs.

UNKNOWN	At least one of the children is in state UNKNOWN.
NOT_READY	At least one of the children is in state NOT_READY or children are in mixed states, for example some are READY some are RUNNING, etc.
READY	All children are READY to take data.
RUNNING	All children are taking data.
ERROR	At least one of the children nodes is in ERROR state.

Table 9.3: DAQ states description for LUs/CUs.

9.4.1.2 DCS DOMAIN DEFINITION

The Detector infrastructure (DCS) domain includes all the components related to the correct detector operation. This includes monitoring sensors related to temperature, humidity, and also the control of the low voltage supply to the electronics. It also contains the supervision of systems like: cooling systems, electrical supply systems, etc. These components are normally stable throughout a full running period.

The DCS is subdivided into LV (low voltage), TEMP (temperature), HUM (humidity) and COOLING domains.

The *Switch_ON* command will carry a parameter *RUN_TYPE*, specifying the desired running mode. In case the operating settings are different for different running modes. For monitoring devices, like temperature sensors, these settings can be flags indicating whether the sensors is enabled or disabled.
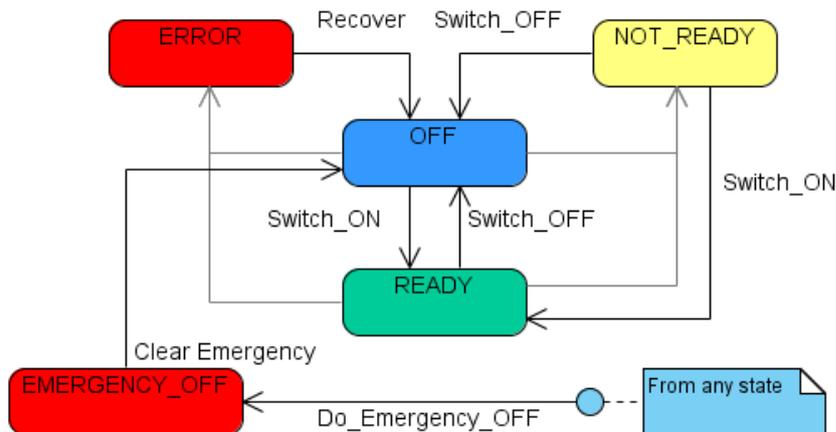
Figure 9.14: DCS domain states.

The FSM containing the *EMERGENCY_OFF* state can be used for equipment that needs to be switched off when external problems are detected. Examples of such equipment can be crates, power supplies, etc.

For the C_6F_{14} cooling plant control, the behaviour of the state machines has slightly been modified. The states have been kept the same but the commands have different names so the plant is always on and does not go off with the rest of the electronics. This behaviour is needed because the silicon sensors need to be always cooled down.

OFF	The channel or device is off (not meaningful for a temperature probe).
NOT_READY	Not necessary for a DU, but can be used to signal a problem, for example a "driver" not running.
READY	The channel or device is set to nominal operation values, or the monitored values are OK (within the operating limits).
ERROR	The channel or device has detected an error, for example a voltage trip or a temperature too high error.
EMERGENCY_OFF	The channel or device has been switched off due to external causes (for example a temperature too high or a cooling problem detected). The device or channel can not be switched back on until a "Clear_Emergency" command is received.

Table 9.4: DCS states description for DUs.

OFF	All children are off.
NOT_READY	Not all children are READY.
READY	All children are READY.
ERROR	At least one of the children is in ERROR state.
EMERGENCY_OFF	At least one of the children in is EMERGENCY_OFF state.

Table 9.5: DCS states description for LUs/CUs.

9.4.1.3 DAI DOMAIN DEFINITION

The DAQ infrastructure (DAI) includes all the components that are providing power, network and computing resources necessary for the data acquisition system. These are the CAEN, MARATON RCMs and TELL1 crates power. These components are normally stable throughout a full running period.

The *Switch_ON* command will carry a parameter *RUN_TYPE*, specifying the desired running mode. In case the operating settings are different for different running modes.

The DAQ infrastructure components publish a state that simply shows if they are *READY* to be used or not. The reason why they may not be *READY* is not part of the FSM. This information is available at the lower level DU where the ECS component specific display panel should provide the necessary information.
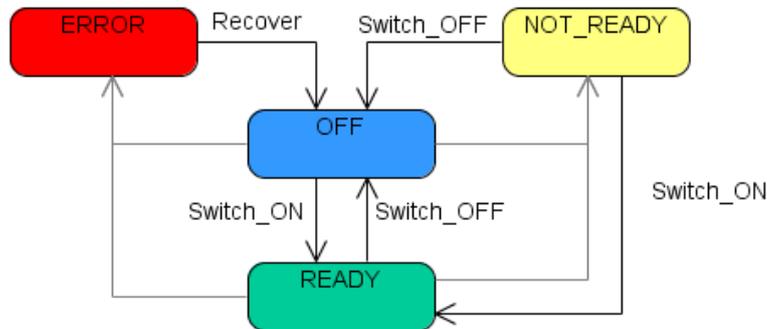
Figure 9.15: DAI domain states.

OFF	The channel or device is off.
NOT_READY	Not necessary for a DU, but can be used to signal a problem, for example a "driver" not running.
READY	The channel or device is set to nominal operation values.
ERROR	The channel or device has detected an error.

Table 9.6: DAI states description for DUs.

OFF	All children are off.
NOT_READY	Not all children are READY.
READY	All children are READY.
ERROR	At least one of the children is in ERROR state.

Table 9.7: DAI states description for LUs/CUs.

9.4.1.4 HV DOMAIN DEFINITION

The HV domain (HV) contains all the components that provide the high voltage power supply to the detector instruments and electronic. Their operation is normally related to a "FILL".
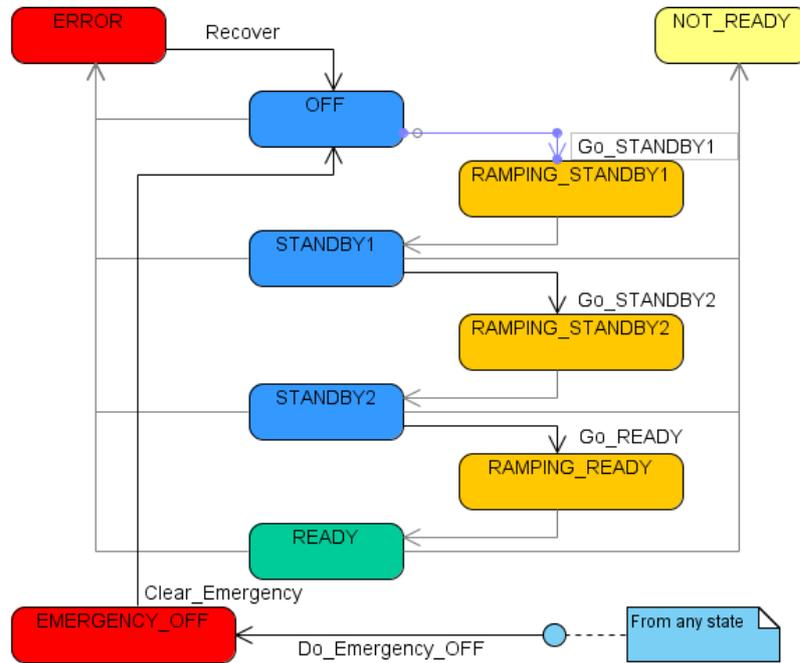
Figure 9.16: HV domain states.

The HV FSM has been designed to give the possibility for the component to have several reference HV values. The main one is the operational value which is the higher one. But in some situations it may be useful not to maintain permanently this value in order to protect the detector or avoid HV tripping. The raising of HV is a time consuming operation that it is necessary to minimize. The idea is thus to maintain the HV value at lower standby levels when the detector is not used in order to protect it while limiting the time necessary to move from this standby value to the operational value when needed.

Since ramping up voltages is time consuming, intermediary states for the various *RAMPING* stages were defined.

The *Go_STANDBY1*, *Go_STANDBY2* and *Go_READY* commands will carry a parameter *RUN_TYPE*, specifying the desired running mode, so that different voltage settings can be applied for different running modes.

The HV DUs (like for the other domains) should implement a time-out and go to the *ERROR* state if the specified voltage level was not reached during the predefined delay.

OFF	The HV channel is off.
NOT_READY	Not necessary for a HV DU.
STANDBY1	The HV channel is set to intermediate low voltage value.
STANDBY2	The HV channel is set to intermediate high voltage value.
READY	The HV channel is set to nominal operation values.
RAMPING_XXX	The HV channel is moving towards a new value after a "Go_XXX" command. XXX can be STANBY1, STANDBY2, READY or OFF.
ERROR	The HV channel has detected an error, for example a voltage trip.
EMERGENCY_OFF	The HV channel has been switched off due to external causes (for example a temperature too high or a cooling problem detected). The device or channel can not be switched back on until a "Clear_Emergency" command is received.

Table 9.8: HV states description for DUs.

OFF	All children are off.
NOT_READY	Not all children are READY.
STANDBY1	All children are in STANDBY1.
STANDBY2	All children are in STANDBY2.

READY	All children are READY.
RAMPING_XXX	All children are moving towards a new value after a "Go_XXX" command. XXX can be STANBY1, STANDBY2, READY or OFF.
ERROR	At least one (or a majority) of the children is in ERROR state.
EMERGENCY_OFF	At least one of the children in is EMERGENCY_OFF state.

Table 9.9: HV states description for LUs/CUs.

9.4.2 THE HARDWARE TOOL

The 'Hw tool' is part of the LHCb specific framework. It is a component developed with the purpose of offering a way to build complex *datapoint* structures in PVSS that connect to the hardware via DIM hiding the underlying complexity of the data exchange mechanisms. This tool is intended to be used either for SPECS and CCPC interfaced hardware, as both of them provide a dedicated DIM server, the 'SpecsServer' and the 'ccserver' respectively.

Both the 'SpecsServer' and the 'ccserver' are DIM servers which offer complete flexibility to associate named services and commands to hardware resources, and complete freedom for the services and commands namespace. The hardware tool permits defining on the client side (PVSS) complex hierarchical fashion structure types, known as hardware types, which correspond to hardware devices. These types are mapped in PVSS to complex structures holding the DPEs needed for the control process. Once the structure is defined any instantiated object of that type will represent a hardware component, and in order to make it available in the server name space, the client must subscribed it in the server. In the subscription process, the 'Hw tool' first sends to the server the service/command names available in the instantiated object together with the associated hardware resources. Just afterwards the server publishes the services/commands and the client connects to them. The name of the service/command is built from the instantiated object name and it connects to the specific hardware addresses defined in the 'Hw tool'. Figure 9.17 depicts how the aforementioned procedure is like:
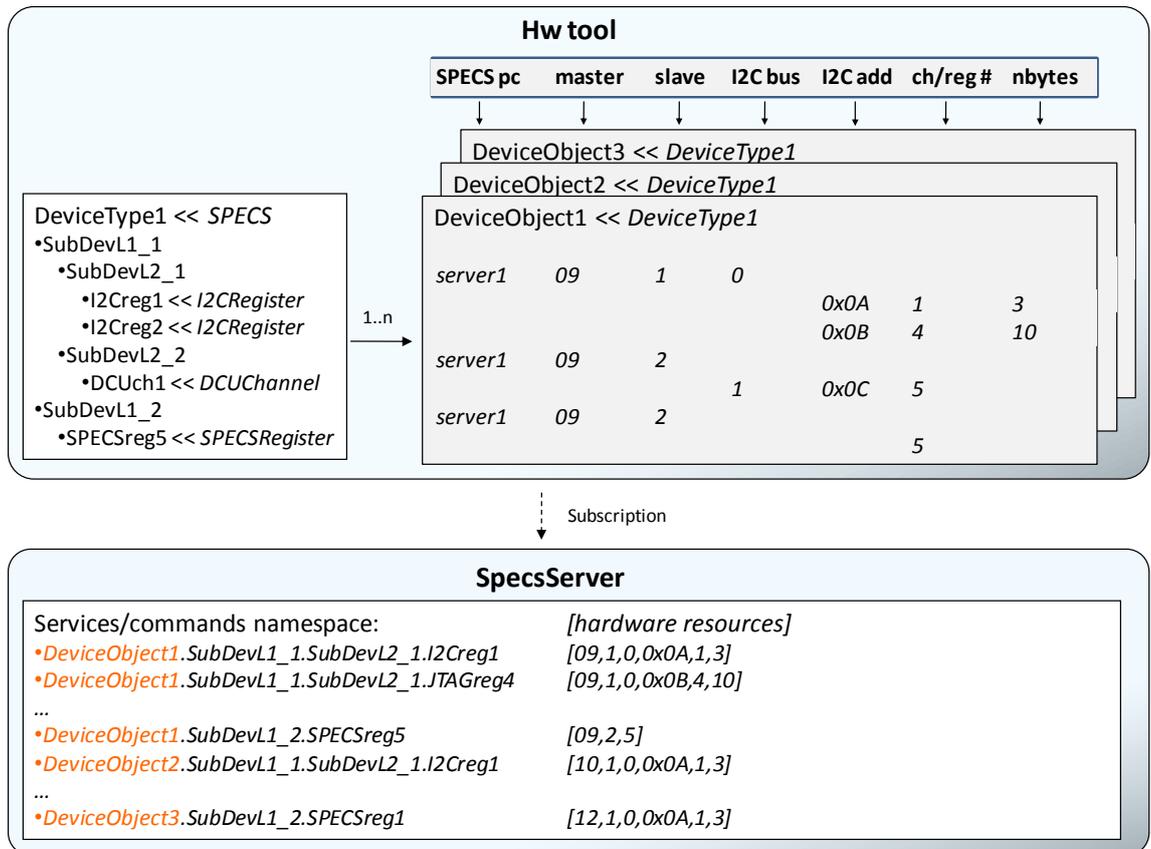
Figure 9.17: Hw tool device definition and subscription diagram. A Device type is defined in the top-leftmost part of the picture. From 1 to n instances of this type can be made with complete freedom for the names. In this case they are called 'DeviceObject1', 'DeviceObject2' and 'DeviceObject3'. Once they are instantiated, the corresponding hardware space has to be assigned and then the subscription process takes place.

1. An example of a device of type *DeviceType1* is defined in the top-leftmost part of the picture. It is of type *SPECS* (thus it represent hardware that is interfaced via SPECS). This type has a complex hierarchical structure: there is a first subdivision into two sublevels *SubDevL1_1* and *SubDevL1_2*; *SubDevL1_1* is further subdivided into two sublevels; the bottom-most level corresponds to real hardware elements of types *I2CRegister*, *DCUChannel* and *SPECSRegister*, but we could also have here abstract entities such as *integers*, *floats*...

2. From 1 to n instances of this type can be made. There is complete freedom for choosing the instance names. In this case they are called 'DeviceObject1', 'DeviceObject2' and 'DeviceObject3'. Once they are instantiated, the corresponding hardware address space has to be defined:

   a. For the bottom-most levels that correspond to real hardware elements, the address (e.g., the I^{2}C address in for the I^{2}C devices), the register number and number of bytes must be set. There are other additional parameters that could be defined but they are out of the scope of what needs to be explained here.

   b. In the overlying levels the following must be put: the SPECS PC where the 'SpecsServer' is running, the Master id, the slave id and the I^{2}C bus in case there are I^{2}C hardware elements in the underlying sublevel.

3. Then the subscription process takes place. The services and commands namespace is built from the instantiated object name and type structure as can be seen in bottom part of the figure. Each service/command has assigned the hardware resources defined in the hardware address space.

9.5 DETECTOR CONTROL HIERARCHICAL ORGANIZATION

The top level division into functional domains has already been mentioned: DAQ, DCS, DAI and HV. The DCS is further subdivided into LV, TEMP, HUM and COOLING. From these top layers downwards the control system follows a subdivision based on the detector physical structure. The sub-layers structure tries to be faithful with the grouping shown in Figure 9.10, but some slight differences have to be introduced depending on the domain.

In the IT detector the previously stated domains are first subdivided into six half-stations with two detector boxes each, except for the cooling system, which is directly underlying below the DCS/COOLING domain and is not subdivided further down. The DCS/HUM domain has not more logical subdivisions below either. The boxes are further down divided into two service boxes and then partitions. One of the service boxes is fully used and houses one CB and four partitions of FE boards and the other one is just utilized partially with one CB and three partitions. The DAQ domain is a bit more complex than the others as the partitions are not the bottom layer, but the DB and FE hybrids, and also at the half-station level there is a separate subdivision for the TELL1s.

In the TT the subdivision follows a similar philosophy but starting the division into regions. The cooling system and the humidity control underlie directly below their respective domains without subdivisions. Just underneath the regions there is a subdivision into top/bottom location of the service boxes and then the service boxes with four partitions each. The TELL1s fit just under the regions sub-division.

The Figure 9.18 corresponds to the hierarchical structure of the DAQ domains for IT and TT. The difference among them is the physical division of the detector and the placement of the TELL1s in different levels of the hierarchy.

9.5.1 NAMING CONVENTION FOR CONTROL NODES

The naming of the CUs is very important as they uniquely identify the FSM control processes. Therefore it is very important that they are meaningful and they do not clash with names of other subsystems. As a rule, every subsystem's CU has to start by the sub-detector identifier which in our case are IT a TT respectively.

The name of each node should be significant and clearly shows what detector subsystem and domain it is part of, e.g., IT_DCS_LV. The children of the sub-detector domain node can be either a functional division or a geographical division, for examplee: IT_DAQ_ST1C, TT_DCS_LV_REGA. The next levels down follow the same convention, e.g. IT_HV_ST1_BOXA, and they are similar for the different domains.

The device unit names follow a different naming convention and their meaning is related to the physical position where they are located or placed in. The following example, IT_ST1C_BOXT_SB3_DB12, corresponds to the Digitizer Board in slot 12 of the Service Box 3 in the first half-station in the 'cryo' side of the IT detector.
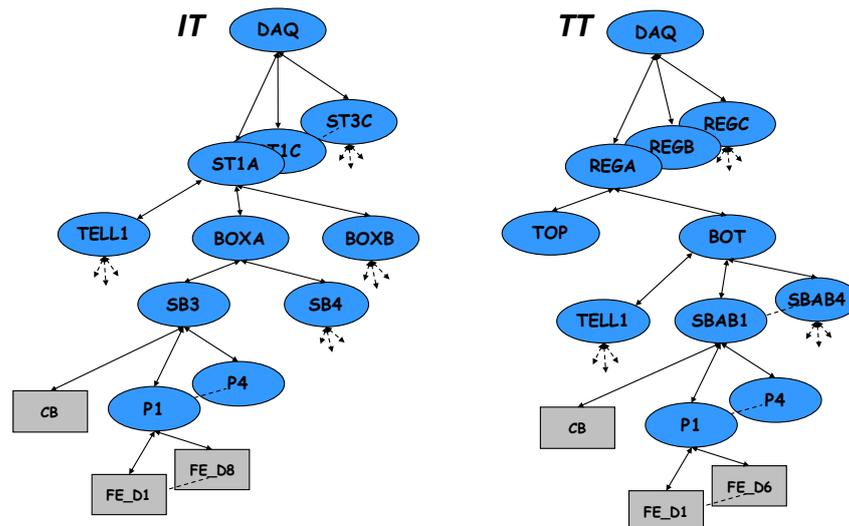
Figure 9.18: IT(left) and TT(right) DAQ domain hierarchies.

9.5.2   CONTROL SYSTEM DISTRIBUTION AND DEPLOYMENT

The ECS control of the IT and TT detector systems is each distributed into four computers, together with other resources distributed among many processing units such as the CCPCs and the cooling plant control. As depicted in the Figure 9.19, the orange blocks represent the computers where the ECS software is running and the green the devices or hardware interfaces for the equipment under control. The blue boxes represent the name of each of the PVSS systems that integrate the detector ECS. The heterogeneity of the system is marked with some small icons which indicate whether the execution environment is 'Linux' or 'Windows'.

The diagram doesn't represent the middleware layer which was described in 9.3.5, issue that also needs to be considered.

This diagram represents the deployment of the IT system. The TT is exactly similar differing just in the naming.

9.5.3   INTERACTION WITH LHCB

The four common domains are the entry points for the LHCb overall ECS to control the sub-detectors, namely: IT_DAQ, IT_HV, IT_DCS and IT_DAI. For stand-alone operation the sub-detectors need some resources that are not part of the ST infrastructure itself but LHCb wide: TFC, HLT, Storage and Monitoring. An online partition with all the necessary resources to perform full detector activities is needed, that is, a TFC partition for driving the fast control signaling as well a slice of resources from the HLT, the Storage and the Monitoring for handling the acquired data, i.e., collect the data, store it and display online results. These resources are dynamically assigned to the detector whenever running in stand-alone is requested. This is achieved by implementing a parallel hierarchy that wraps the four top entry domains and which all hang from a standard top CU type that implements all the functionalities for performing the overall control of the detector as shown in the Figure 9.20.

9.6   DETECTOR CONFIGURATION

The control system is composed of hundreds of devices whose configuration parameters will be retrieved from the configuration Db. The devices can either be hardware of software entities that need a specific configuration according to the running mode of the experiment.

This variety of modes of operation and the change in behaviour of the hardware will require some flexibility and re-configurability of the devices. The configuration of these devices and applications from a configuration database must be synchronized in order to allow for the coherent operation of the experiment for a given run type. This is done by means of the 'FSM-Configuration DB tool', which is a specific component integrated in the FSM control that automates the handling of configuration data at run time, ensuring the availability of configuration data during the FSM transitions. The tool defines specific device units called *configurators* which act as links between the FSM and the Configuration DB. The *configurators* access the database and preload the configuration data into the recipe caches so it is available from PVSS.

This dynamic configuration data are grouped into the so called recipes. A recipe is a collection of settings (values or/and alert parameters) corresponding to a number of DPs in PVSS and identified by a unique name. The recipe name is built from the run type and the command name, so that in the event of a command reception the *configurator* receives the *RUN_TYPE* and the command name parameters from the top CUs, builds the recipe name and finally is able to load the right recipe data from the cache to be applied to the concerned DUs.

There are a set of basic *RUN_TYPE* foreseen for global running, such as *PHYSICS* (normal data taking), *COSMICS* (for runs where we take cosmic events) and *TEST* (for global tests).

The tool offers the flexibility of creating compound names in a staggered fashion and overloading recipe values. When the *configurator* seeks for the parameter values it always start looking in the recipe with the entire name, e.g. *ST_TEST/ORX*, then searches one level down, that is *ST_TEST*, and finally looks up in the *DEFAULT* recipe in case it did not find the value in the previous. The *configurator* eventually goes to error state if it does not find the recipe value anywhere.
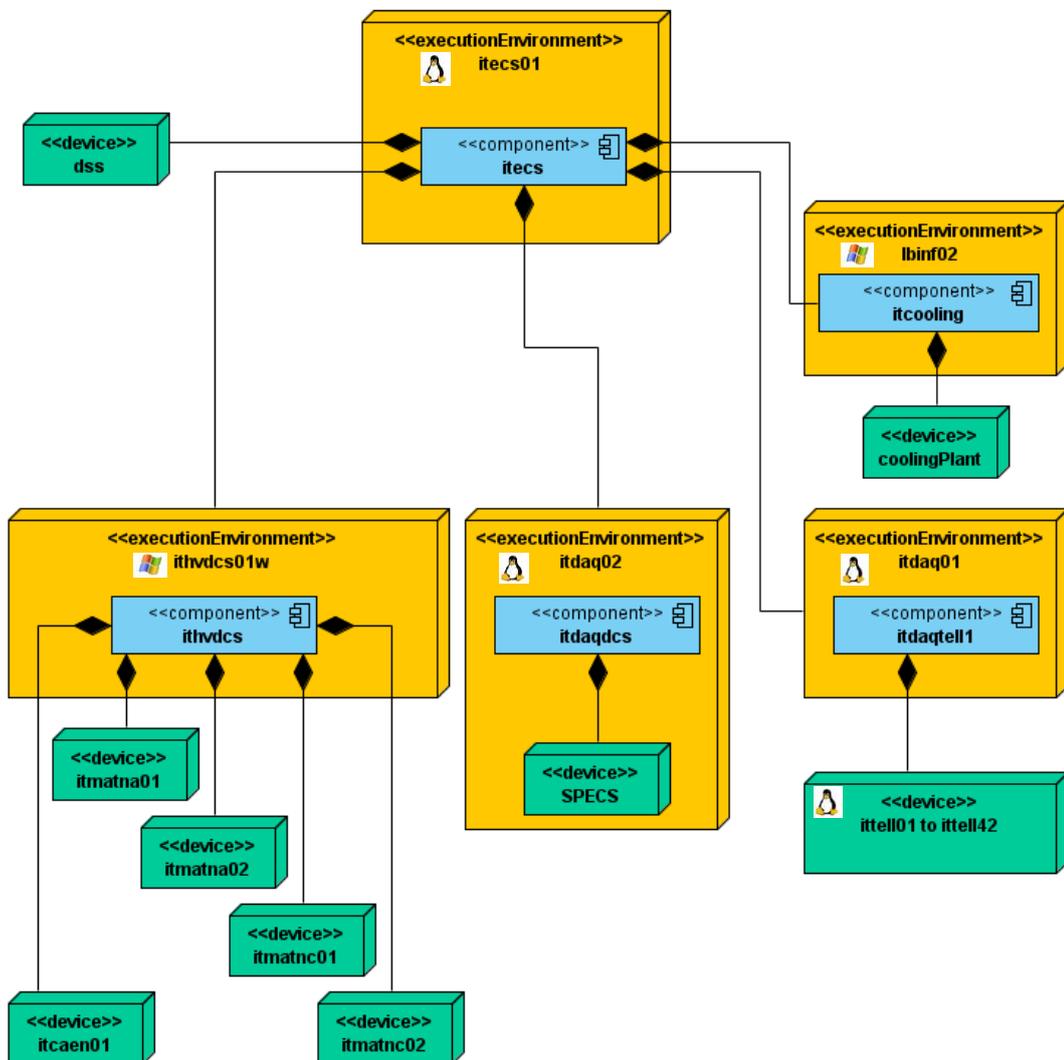
Figure 9.19: Control system deployment of the IT detector.

- 174 -
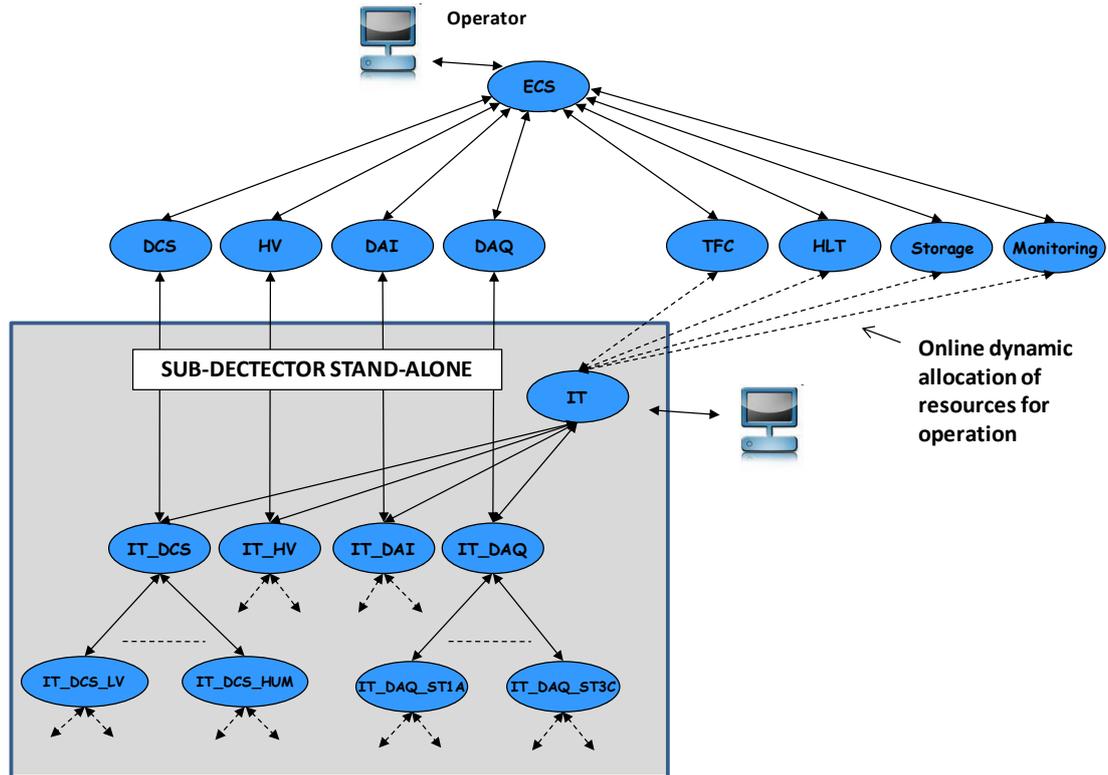
Figure 9.20: ST interconnection with LHCb.

9.6.1   ALARMS, ARCHIVING AND AUTOMATION

The alert handling, monitoring data archiving and process automation is a matter that has been considered of major importance to ensure the proper and safe working of the detector. A specific review with external referees took place at CERN to assess that the ST has put in place the right procedures and that is why there is a dedicated chapter for this issue.

CHAPTER 10

THE IT AND TT ECS IMPLEMENTATION

ABSTRACT

This chapter describes the technical implementation of the LHCb Silicon Tracker Experiment Control System. It describes how the hierarchical control for IT and TT has been designed and developed using the PVSS SCADA and the JCOP Framework and following the LHCb integration rules.

"No engañes a tu corazón con inútiles palabras que solo demostrarían la escasez de tu inteligencia"

Confucio

10.1 OVERVIEW

The Silicon Tracker (ST) Experiment Control System is, as the real detector, sub-divided into two separate systems, the IT and the TT systems. A common approach has been followed from the hardware and software point of view to build a similar system for both sub-detectors without needing to duplicate efforts. The partitioning concept of the hardware has led to follow the same partitioning of the different software entities that model the control system. In this chapter we will describe the implementation of the IT and TT hierarchical ECS based on the JCOP FSM framework and PVSS from the top to the bottom layers.

10.2 THE ST CONTROL SYSTEM OUTLINE

The detector control is split into four top entry domains, namely DAQ, DCS, DAI and HV. In addition the DCS domain is subdivided into the LV, TEMP, HUM and COOLING domains. The devices functionalities must be under the control of one of these domains in the form of a Device Unit (DU). A DU does not necessary need to match with an entire hardware entity or board as it could be the case that the board may develop different tasks that belong to different domains. For example, the Control Board carries out tasks that belong to the DAQ domain (TTCrx, Delay25 chip …) and the DCS (temperatures and humidity monitoring …).

In the DCS_LV domain the list of DUs used are:

- *FwMARATONCh*: they are provided by the FW. They are used to model the MARATON power supply channels.
- *LVCB*: it is a DU created to monitor the status of the low voltage power in the CBs. It is also used to initialize the CB at start-up by resetting and initializing the SPECS slaves and the rest of the other elements.
- *LVPartition*: they are used to control and monitor the regulators that power the Digitizer Boards and hybrids.

The DUs in the DCS_TEMP domain are:

- *TempBox*: it monitors the detector box temperatures signals, provided by a PT1000 that is read out by a DCU placed in the CB.
- *TempSB*: it monitors the Service Box temperatures (PT1000 read-out by a DCU placed in the CB).
- *TempPartition*: used for monitoring the temperature in a partition of hybrids. The DCU that reads the PT1000 housed in the hybrid is placed in the DB.

In the DCS_HUM domain the only DU is the one that reads the humidity sensor:

- *HumBox*: this DU monitors the humidity and the dew point in the detector box. It uses the readings of the HMX2000 sensor inside the box and the temperature in the sensor surrounding. This sensor is read-out by a DCU housed in the CB.

There are two types of DUs in the DCS_COOLING domain modeling the cooling plant behaviour which have been developed by the ST group:

- *FwCaVPlantIT*: it models the entire IT cooling plant behavior but the temperatures of the loops and the cooling circuit valve.
- *FwCaVLoopMasterIT*: it models the IT "master loop", i.e., the loop which can open/close the valve for the whole cooling circuit. It also monitors the temperature of the loop.
- *FwCaVLoopIT*: it monitors the temperature of each of the eleven IT normal loops.
- *FwCaVPlantTT*: it models the entire TT cooling plant behavior but cooling circuit valve.
- *FwCaVLoopMasterTT*: models the TT "master loop" which can open/close the circuit.

The DAQ domain, the one that deals with the electronics which taking part in the data acquisition, defines four types of DUs:

- *CONTROL_BOARD*: it configures the CB elements that have to do with the data acquisition process, that is, the TTCrx, the Delay25 chip, the channel B decoder of the SPECS slave, enable the drivers for the TFC signals and start the monitoring of the TTCrq status signals.
- *DIGITIZER_BOARD*: it configures the GOLs in the board.

- *LADDER*: for historical reasons the name of ladder is being used for modeling the hybrid. It configures the Beetles of a given hybrid.

- *TELL1_ST*: it represents the TELL1 model of the ST. It handles the TELL1 configuration and it is provided by the framework.

In the HV side, the only DU type is the *FwCAENCh*, which is used to control the CAEN HV channels. This DU is provided as part of the FW.

For the DAI domain, the DUs are already provided by the LHCb 'Online group'.

The previous DUs are logically arranged in each domain in a hierarchical FSM structure that follows the detector partitioning. They are also physically distributed among different computers following the same philosophy in IT and TT: the DUs that use the SPECS protocol are in a dedicated computer (*LVCB, LVPartition, TempBox, TempSB, TempHybrid, HumBox, CONTROL_BOARD, DIGITIZER_BOARD, LADDER*); one computer for the TELL1s as programming all the TELL1s uses a lot of CPU load; the *FwMARATONCh* and *FwCAENCh* also run in separate computer as they communicate via OPC and the servers just run in 'Windows'; and finally one computer where all the top and Safety FSM nodes are running. Just one project in each computer handles the ST control.

Each of the DU and CU panels has to provide useful and easy to understand user interfaces for the piece/s of hardware or detector they represent.

The software is composed of four components, one per PVSS project, and two subcomponents… The components install the DUs, CUs and all the libraries needed for each of the projects.

10.2.1 SOME BASICS ABOUT THE FSM SOFTWARE INTEGRATION IN PVSS

Before start describing how the DUs are implemented and connected to PVSS, there are some basics about PVSS that need to be understood. PVSS is an event-driven system. If a certain function needs to be executed as a result of an event (for example, a change in value of a DP attribute), then, a connection between the required DP attribute and the function to be run must be set up. This is achieved by the function *dpConnect()*. As different events, sometimes occurring simultaneously, can trigger different functions, such functions (also known as *Callback()* or *Work()* functions) are executed in parallel. One instance of a function processed in this way is called a thread. A thread has its own data or stack area of memory, i.e. if a certain *Callback()* function is connected by *dpConnect()* to various DP attributes and then executed, sometimes in parallel, when these attributes are changed, with the local variables are only visible or accessible inside a thread, but not between threads. This mechanism is used in the DUs so it permits a high degree of parallelism when updating the DUs states. Together with this function, the *dpSet()* (assigns DP attributes values or creates DP configs) and *dpGet()* (reads values of DP attributes in variables) are most relevant functions to work with *datapoints*. There is also a special attribute of the DPEs which permits simple implementing mathematical links between *datapoints*, the so called "DP function" or *'_dp_fct'* (the *datapoint* function contains one or more parameters used to calculate the value of the defined function and to assign it to the original value of the *datapoint*).

In SMI++ FSM modeling tool there are two types of objects: abstract (CU and LU) and physical (DU) objects. Physical objects implement the interface to real devices, have a list of allowed states, and in each state a list of allowed actions. When they receive a command they have to act on the device they model, and when the device's data changes they have to maybe change state. The connection between the SMI++ processes and PVSS is done via DIM in a transparent way to the user and the ECS developer. On the other hand, a DU corresponds in PVSS to DP of a certain type, and this DP is what provides the real connection to the hardware values through any kind of PVSS supported driver. For each DU there are three *Work()* functions (PVSS script) that are in one hand somehow *'dp_connected'* to the DU DP, and in the other hand using DIM to connect to the SMI++ physical object that models the device:

1. *Initialize()* function: this function is just called at the startup of the FSMs, usually just when the system is initialized.

2. *Value_Changed()* function: within this function is where the new state of the DU is set depending on the value of the DPEs values passed as arguments. The function is called whenever there is a change in any of the DPE represented by the function arguments. The state set in this PVSS script will automatically be propagated to the SMI process modeling the device.

3. *Do_Command()* function: whenever a command for the device is received by the SMI object this PVSS script will be executed. Therefore, it is in this place where the sequence of actions that must be executed for each of the commands must be coded, evidently in the PVSS scripting language.

In addition to the previous code, the developer also has to define the list of possible states and the list of allowed actions in each state. The "Device Editor Navigator" tool of the FW permits defining it in a transparent way without needing to code any SML code.

The DEN also permits creating CU/LU types and defining the CUs behavior in a completely transparent way to the developer, without any need to know how the SML code looks like. In the DEN one must define the list of possible states, the list of allowed actions in each state, the command actions that are propagated to the children and the state transition conditions for each of the CU types. Besides, it permits starting and stopping the SMI++ processes which represent the CUs.

10.2.2 CONTROL SYSTEM DISTRIBUTION AND DEPLOYMENT

The control system for each IT and TT is distributed into several computers. In each of the computers one PVSS project carries out the control tasks. The structure of the control system deployment is as described in Figure 10.1.
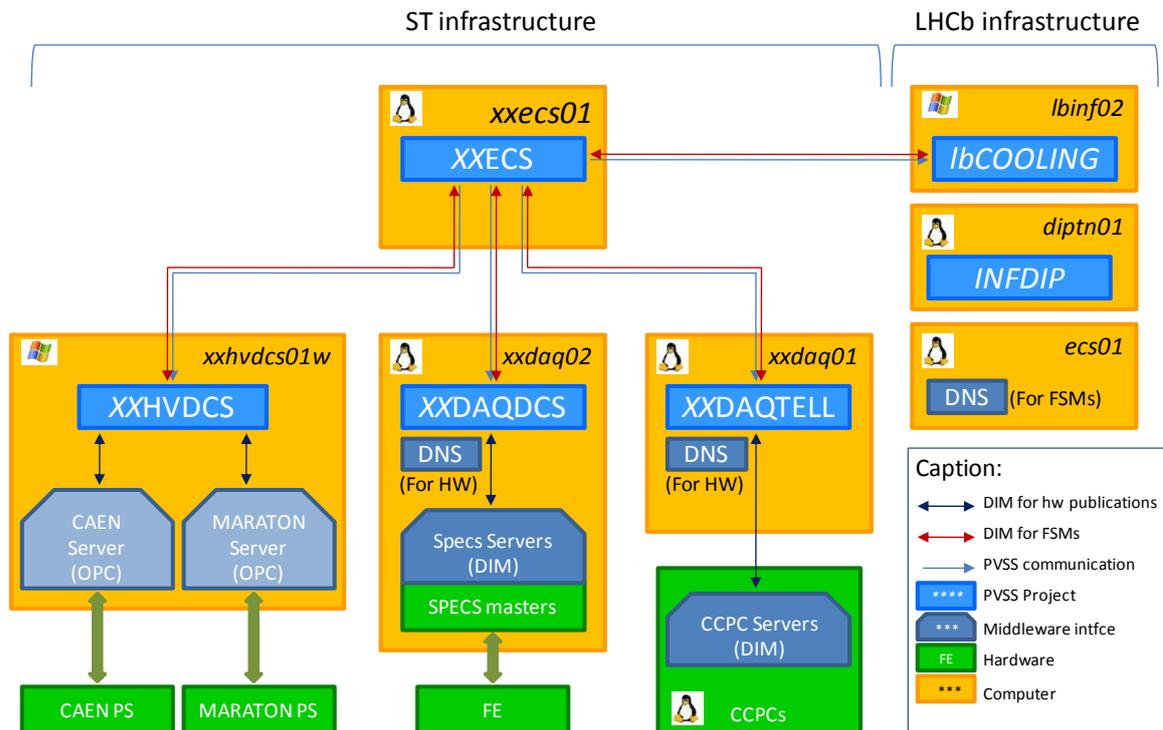
Figure 10.1: Control system distribution and deployment.

There are four computers for IT and four for TT for specific use by the ST group. The '*XX*' term in the figure is a wild card for '*IT*' or '*TT*'. In addition there is also a connection to external LHCb computers that provide cooling plant control and DIP connectivity. The description for each of the computers is:

- '*xxecs01*': this computer is for the 'XXECS' PVSS system, which runs all the high level FSMs (figure xx), linking together all rest of the systems. It also runs the automatic actions Safety FSMs and interfaces with the DSS system.

- '*xxhvdcs01w*': it runs the OPC servers for the CAEN and MARATON PS and also the system ('*XX*HVDCS') where the FSMs associated to this hardware are deployed.

- '*xxdaq02*': runs the DIM 'Specsservers' (a server process per SPECS master), i.e. the middleware interface for the whole FE. The services are published in the local DNS also running in this computer. The lowest level FSMs for all this hardware are deployed in this computer.

- '*xxdaq01*': the is the system used to carry out the control of all the TELL1s. The 'ccservers' in the CCPCs publish their services in the DNS of this computer.

- '*lbinf02*': this is a LHCb computer that provides a control interface with the cooling plant. The '*lbCOOLING*' project provides communication with the cooling plant and implements the FSMs of the DCS_COOLING domain.

- '*diptn01*': this computer publishes the DIP services of the protected computers in the CERN technical network.

- 'ecs01': this computer runs the DNS where the services for all LHCb FSMs are published. In this way the FSMs of all detectors are 'visible' for the top LHCb control nodes.

10.2.3 THE TOP CONTROL NODES TYPES

We will use UML class diagrams to describe how the hierarchical structure is like. The CU type for top nodes is shown in the following scheme.
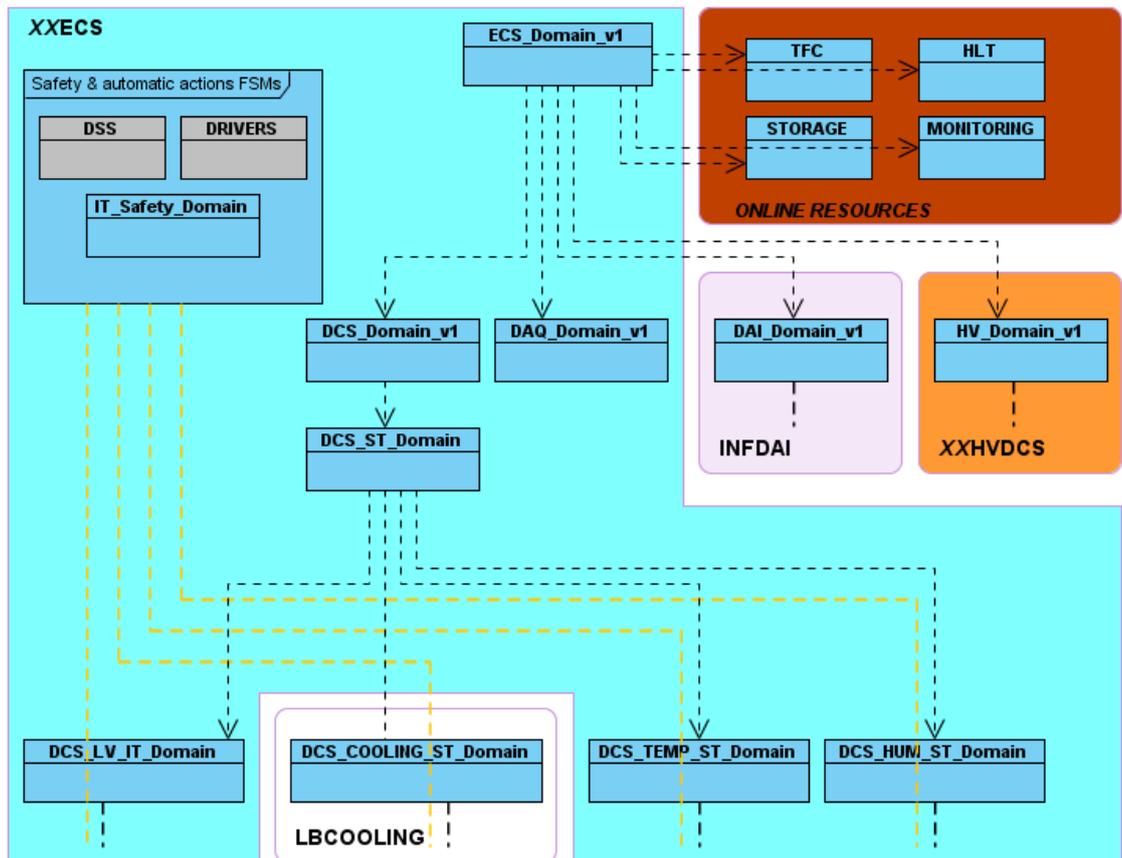
Figure 10.2: Top control domains class type and deployment.The blue background indicates that FSMs of those types run in the 'XXECS' PVSS system, the orange corresponds to the 'XXHVDCS' system, the white to the 'lbCOOLING', the light purple to the 'INFDAI' and the online resources run in several computers. The system 'XXDAQDCS', which should be below the 'DCS_HUM_ST_Domain' is not included in this picture. The top node is of class 'ECS_Domain_v1', and the black dashed indicate a hierarchical dependency. The yellow dashed lines correspond to logical links for the parallel Safety tree, which is used to perform automatic actions.

The previous picture depicts how the system top nodes are structured. The background colors indicate the different systems where the FSMs run. The top node is of class type *ECS_Domain_v1*, from which hung the rest, indicated with the black dashed lines. The *DCS_Domain_v1, DAQ_Domain_v1, DAI_Domain_v1* and *HV_Domain_v1* types correspond to the four top entry domains mandatory for each sub-detector. The elements over *ONLINE RESOURCES* correspond to the dynamically allocable resources provided by the 'Online System' needed for standalone runs. The incomplete lines in the bottom of the picture mean that hierarchy continues downwards.

The *IT_Safety_Domain* type corresponds to the top node of the parallel *Safety tree*. This tree is compound of several FSMs which link to the normal tree via logical links, indicated with the yellow dashed lines. This just represents the philosophy of how it is implemented, not the detailed implementation. The diagram just means that this domain uses the information from the lower LV, Temperature, etc. DCS domains and will act on the cooling and LV domains accordingly. The *Safety tree* implementation is sub-detector specific and it can link to other domains than the indicated in the picture.

10.2.4 THE CONTROL SYSTEM COMPONENTS

Each of the ST PVSS systems is packaged as a FW component. Figure 10.3 shows the four different components for the IT corresponding to the *ITECS*, *ITHVDCS*, *ITDAQDCS* and *ITDAQTELL* PVSS systems, plus one other for the cooling control. Besides those, the *itCore* and *stCore* are subcomponents that are needed by the rest.

The *stCore* component provides all the libraries and DPs that are common for IT and TT and are needed by the rest of components as depicted in the Figure 10.3. This includes libraries with global constants such as error definitions, hardware masks for hardware, states definitions; libraries implementing the behavior of the DUs; common UI panels; and also the *datapoints* that define the DUs and CUs behavior needed by the FSM tool.
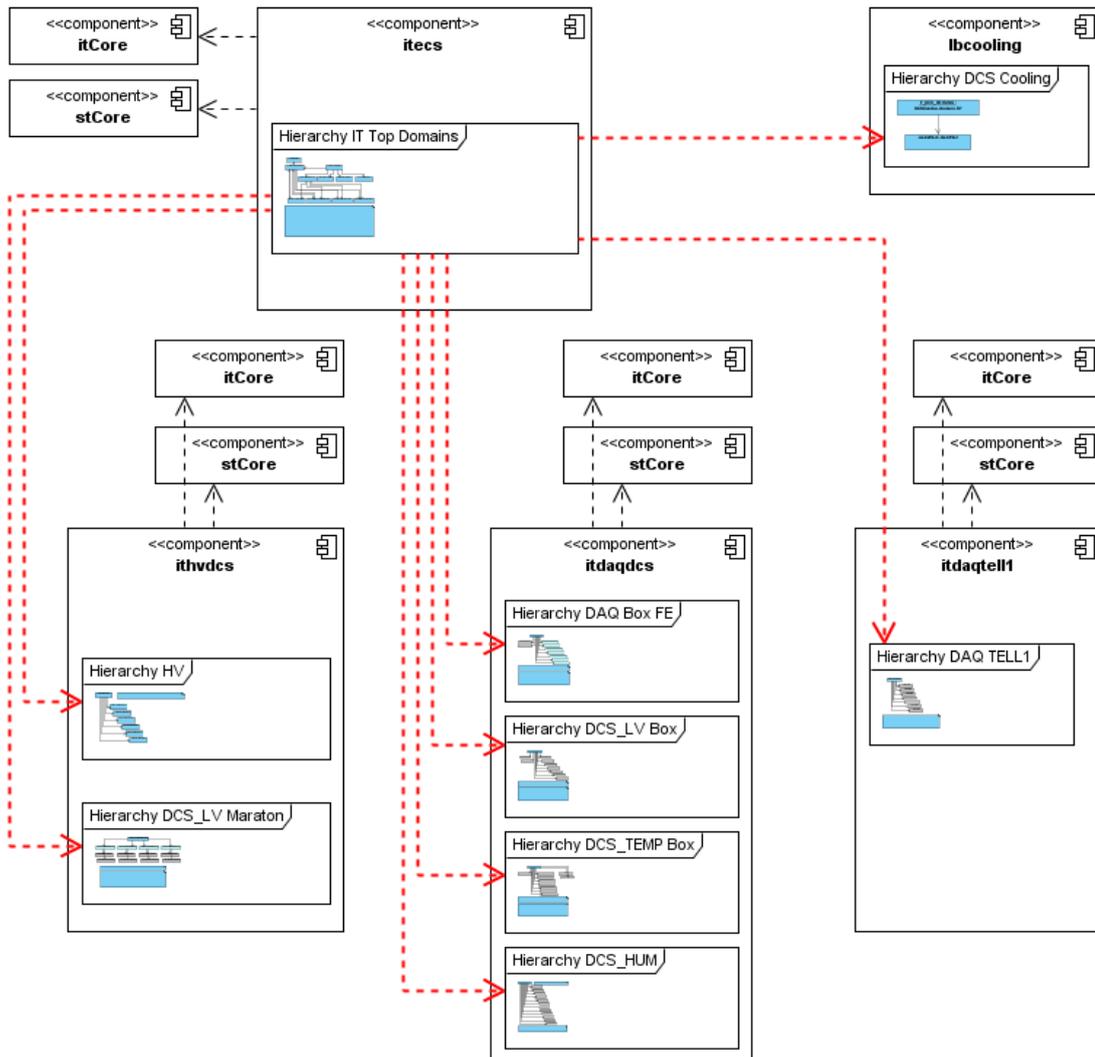
Figure 10.3: IT components distribution.

The *itCore* or *ttCore* component provides all the libraries and DPs that are sub-detector specific and common for all IT and TT projects. In these libraries there are a set of functions that are the interface for sub-detector specificities, which means that the function definition is the same in both components but the implementation different, and like this the different geometries and partitioning of the detectors can be handled in a transparent way by the rest of the components. They are the "interface" components.

The *itecs*, *ithvdcs*, *itdaqdcs*, *itdaqtell* provide the libraries to build the DPs and the FSM objects of the corresponding project. Building the DPs implies putting the right settings of the hardware, applying the alarm thresholds settings and activating the right archiving DPEs and options. These libraries are equal in both sub-detectors except in library that builds the FSM hierarchy related to that system. They also provide the

detector specific panels. The *lbcooling* provides the libraries and DPs needed by the ST in the cooling project.

These components also request to install the right versions of all the need JCOP FW components, such as the "Hw tool", "Specs FW", "CAEN FW" …

The explanations above extend also to the TT.

10.3 THE DEVICE UNITS DYNAMIC BEHAVIOR

In the PVSS side, the DUs are defined by the DP that provides the data structure and three wrapping function scripts that implement the behavior. The functions are: *Initialize()*, *Value_changed()* and *Do_Command()*.

The DP hardware types have been created using the '*Hw tool*' component as it provides a standard way to interface electronic devices using the SPECS system. There is a set of base types that are used to build the hardware types that represent the DUs which will be introduce later in this section .

Inferring the DU state from the hardware values could be in some cases a quite complex task and may depend on several DPE values. As an example, the status of the low voltage for a partition of DBs and hybrids depends on the *OCM* signals from the regulators, the *INH* lines that power on/off the regulators and the voltage reading. The way to deal with this is to implement a *dpConnect()* call in the DU *Initialize()* function that will update the DU status based on the aforesaid DPE values, which are passed as an argument list in the *dpConnect()*. The connected "*Work()*" function will be executed as a parallel thread whenever any of the DPEs passed in the argument list gets updated. The Figure 10.4 shows the skeleton of the wrapping around the DP structure to implement the DU behavior. The caption in the figure gives the pertinent explanations of how this works.

```
void Initialize(domain, device){
        ....
        dpConnect("updateStatus", dpe1, dpe2, dpe3, ....);
        ....
}

void Value_Changed(domain, device, status_dpe, *exitState){

        // Set the DU state (exitState) depending on the 'status' dpe.
}

void Do_Command(domain, device, command){

        // Execute the sequence of actions for the given command
        // and set the 'status' dpe to the right valus so it updates
        // the DU state.
}
```

Event driven & parallel execution function
```
updateStatus(dpe1, dpe2, dpe3, ....){

        // Calculate the new state depending on dpe1, dpe2 ...
        // and update the dpe 'status' to the new value so it will
        // update the DU state
}
```

Figure 10.4: Wrapping scripts and state update scheme skeleton. We find here the 'Initialize()', 'Value_changed()' and 'Do_Command()' functions of the DU. In the 'Initialize()' we make a 'dpConnect()' to the "work" function 'updateStatus()', which receives as arguments the list of "sensible" DPEs and implements the needed algorithms to calculate the new status of the DU. The new state is applied by writing to the DPE 'status' of the DU, which will force a call to the 'Value_changed()' function. It is there in where the new state of the DU is set by overwriting the 'exitState()' "output" argument of the function according the value of the 'status' DPE.

All DUs implement two DPE:

- 'status': to define the status of the DU (*NOT_READY*, *OFF*, …)

- 'mode': to indicate whether the DU status gets updated as a function of the current hardware readings (*ACTIVE_MONITORING*, *PASIVE_MONITORING*). This is basically used during startup or in the event of an error. At start up not all the measurement may be still updated with the real HW readings so the DU is set temporarily to *PASIVE_MONITORING* before all the DPEs with real readings have been updated. In the event of an error a burst of bad readings caused for example due to LV failure can flood the alert screen or overload the computer recalculating the state of the DU, which we already know that it has to be *ERRROR*. In this latter case the DU will stay in *PASIVE_MONITORING* until the error is acknowledged.

The hardware types used to perform ADC measurements also implement the 'mode' DPE, but the meaning is different at this scope:

- 'mode': indicates whether the measurement is *ENABLED* or *DISABLED*. If it is *ENABLED* the monitoring of this parameter will be initiated when the system is started up, the alert handling enabled and the status of the DU updated accordingly. If *DISABLED* the value will not be used to recalculate the state.

It is out of the scope of this writing to describe the specific implementation for each of the DU types described in the next sections. Nevertheless it is worth to mention that thanks to the approach followed during the design and implementation of the control software both IT and TT use the same PVSS scripts to implement the DUs behavior.

10.3.1 BASE HARDWARE TYPES

The following hardware types are the most basic building elements which are used to model the DUs associated hardware types. In some cases they correspond to entire hardware entities (in this case to a chip), such as the TTCrx, the Delay25 chip, the Beetle and the GOL. A direct correspondence between a base type and a device would be the most reasonable approach, but as the ECS is structured in a domain based rule (DAQ, DCS ...) that does not work out. For this reason, some of the base types just correspond to a subpart of a device, that is the case for the *TEMP_CHANNEL* which represents a DCU channel.

In the subsequent tables the names show the names of registers defined for each type, the type of hardware register together with the size in bytes, the access mode and a brief description. The type of register can either be a real hardware register such as I^2C or a abstract primitive type (*string*, *float* …). The field 'Mode' is basically for the I^2C registers and indicates the way they are accessed:

- '0': I^2C "*combined*" mode for most I^2C standard chips (Beetle, for example).
- '1': I^2C "*separated*" mode for chips with a pointer register (TTCrx for example).
- '2': I^2C "*combined1*" mode for the Calorimeter chips.
- '3': I^2C "*separated1*" with pointer register but no auto-increment (GOL).
- '5': I^2C "*combinedFifo*" for writing/reading (in particular reading) Beetle shift registers.
- '6': I^2C "*combinedNoAutoInc*" for writing/reading all Delay25 registers in one go.

10.3.1.1 TTCRX_ST

The hardware type for the TTCrx chip defines all the chip internal registers, and there is not any reference to the data and pointer registers. The software makes transparent to the user and the PVSS developer the access to the internal registers by choosing the right mode, so it is not needed to know how the real sequence of commands at the I^2C level is done. All the registers are of I^2C type and the I^2C "*separated*" (mode for chips with a pointer register).

TTCRX_ST			
Reg. name	Type	Mode	Description
FineDelay1	I2C[1]	1	Clock1 fine delay register in steps of 104ps following chip coding
FineDelay2	I2C[1]	1	Clock2 fine delay register in steps of 104ps following chip coding
CoarseDelay	I2C[1]	1	Delay of trigger signal
Control	I2C[1]	1	Control register
Status	I2C[1]	1	Status register
Config1	I2C[1]	1	Configuration register 1

Config2	I2C[1]	1	Configuration register 2
Config3	I2C[1]	1	Configuration register 3
SingleErrCnt	I2C[1]	1	Counter of single errors from Hamming decoder
DoubleErrCnt	I2C[1]	1	Counter of double errors from Hamming decoder
SEUErrCnt	I2C[1]	1	Single Event Upsets counters
BunchCnt	I2C[1]	1	BXid counter
EventCnt	I2C[1]	1	Event counter

10.3.1.2 DELAY25

This type is for the Delay25 chip of the CB. It implements the six device registers so they can be accessed individually. There is also one called 'All' which in fact represents the group of six and permits accessing all the registers in one go by software transparent means, using the sub-address mode "*CombinedNoAutoInc*".

Delay25			
Reg. name	Type	Mode	Description
CR0	I2C[1]	0	Control register for delay line 0
CR1	I2C[1]	0	Control register for delay line 1
CR2	I2C[1]	0	Control register for delay line 2
CR3	I2C[1]	0	Control register for delay line 3
CR4	I2C[1]	0	Control register for delay line 4
GCR	I2C[1]	0	General control register
All	I2C[6]	6	The group of 6 registers

10.3.1.3 SPECSMEZZANINE_ST

It represents the SPECS slave mezzanine internal 16-bit registers, which have to be defined of type '*REG*'.

SpecsMezzanine_ST			
Reg. name	Type	Mode	Description
RegOutMSB	REG[1]		Clock1 fine delay register in steps of 104ps following chip coding
RegOutLSB	REG[1]		Clock2 fine delay register in steps of 104ps following chip coding
ConfRegOutMSB	REG[1]		Delay of trigger signal
ConfRegOutLSB	REG[1]		Control register
CtrlReg	REG[1]		Status register
StatReg	REG[1]		Configuration register 1
BusSelReg	REG[1]		Configuration register 2
IntVectReg	REG[1]		Configuration register 3
DatePROMReg	REG[1]		Counter of single errors from Hamming decoder
ChanBDelayReg	REG[1]		Counter of double errors from Hamming decoder
DefIntVectReg	REG[1]		Single Event Upsets counters

10.3.1.4 BEETLE_ST

It represents the Beetle chip. The device has twenty control registers assigned to the least significant sub-addresses, three control shift registers and a SEU status register in the last sub-address. The hardware type is implemented such the twenty registers are software encapsulated into one of size 20 called '*Config*'. The three shift registers ('*TpSelect*', '*CompMask*', '*CompChTh*') have to be accessed separately as the addressing mode is different. The '*SEUCounts*' register is also accessed individually because the address is not consecutive with the first twenty. The '*I2CfailCnt*' is an integer element that could eventually be used for counting the number of erroneous I^{2}C accesses to the Beetle chip.

BEETLE_ST

Reg. name	Type	Mode	Description
Config	I2C[20]	0	The first 20 registers of the Beetle
CompChTh	I2C[128], sAdd 20	5	Comparator channel threshold, shift register
CompMask	I2C[16], sAdd 21	5	Comparator mask, shift register
TpSelect	I2C[16], sAdd 22	5	TestPulse selection, shift register
SEUCounts	I2C[1], sAdd 23	0	Single event upset counter
className	string		Used to Beetles to logical groups

'className' was foreseen for a ST specific tool ('*Beetle tool*') intended to group sets of Beetles under different classes depending on the physical location in the detector. It might be possible that Beetles placed closer to the beam with higher occupancy would need to be configured with different settings from the others; hence their recipes would be built based on their '*className*' assignment.

10.3.1.5 GOL

The GOL chip also uses the pointer and data registers I^2C scheme to access the internal registers. As the internal registers follow a consecutive numbering, a special sub-addressing mode ("*SeparatedNoAutoInc*") has been software implemented, so several registers can be accessed in one go. As the two last registers are not writeable, '*ConfigAll*' is for writing the three first control registers and '*ReadAll*' is for reading back all of them.

GOL

Reg. name	Type	Mode	Description
ConfigAll	I2C[4]	3	The 4 configuration registers of the GOL. They are W/R registers
ReadAll	I2C[6]	3	The registers of the GOL, 4 configuration + 2 status registers. The 2 status registers are only read
GOLStatus	int		Comparator mask, shift register

10.3.1.6 CB_REGULATOR

It is used to monitor the three regulators implemented in the CB. The output voltage is read out by one of the DCU channels in one of the SPECS slaves ('*ADC_VOLT*') and a conversion function is needed to get the voltage value ('*VOLT_REGULATOR*'). The conversion formula, implemented as a 'DP function', is:

► '*VOLT_REGULATOR*' = '*ADC_VOLT*' • '*DCU_GAIN*'

CB_REGULATOR

Reg. name	Type	Mode	Description
ADC_VOLT	DCUChannel		ADC channel to read the regulator output voltage
VOLT_REGULATOR	float		Value in volts. dpFunction defined.
DCU_GAIN	float		Estimated gain of the DCU
mode	int		Channel enabled/disabled
status	int		Status of the regulator
errCount	int		Number of bad consecutive readings

10.3.1.7 DCU_VREF

This type is only used to read out the voltage reference channel of a DCU.

DCU_VREF

Reg. name	Type	Mode	Description
DCU_Channel	DCUChannel		ADC channel to read the DCU internal Vref channel

10.3.1.8 HUM_CHANNEL

This type is used to get the voltage output from the HMX2000 sensor. This device is read out by one DCU channel, thus '*ADC_HMX2000*' is of type '*DCUChannel*'. The device output voltage is obtained in '*VOLT_HMX2000*', which is connected to the ADC reading in '*ADC_HMX2000*' through a DP function'. This function is:

▶ '*VOLT_HMX2000' = 'ADC_HMX2000' • 'CURVE_SLOPE' + 'CURVE_OFFSET'*

'*CURVE_SLOPE*' and '*CURVE_OFFSET*' are parameters obtained during the calibration of the Control Board DCUs.

HUM_CHANNEL			
Reg. name	Type	Mode	Description
ADC_HMX2000	DCUChannel		ADC channel to read the HMX2000 humidity sensor output voltage
VOLT_HMX2000	float		Value in volts of the humidity sensor. dpFunction previously defined
CURVE_SLOPE	float		Calculated slope of the calibration curve for the conditioning and acquisition circuitry
CURVE_OFFSET	float		Calculated offset of the calibration curve for the conditioning and acquisition circuitry
mode	int		Measurement enabled/disabled
status	int		Status of the measurement
errCount	int		Number of bad consecutive readings

10.3.1.9 LV_HYBRID

It is used to handle the hybrids low voltage regulators and to monitor their status. The formula for the analog power regulator output value is given by:

▶ '*VOLTS_HYBRID_Vana' = (2.5- ('ADC_VOLTS_HYBRID_Vana' • GAIN_VOLTS_HYBRID_Vana')) • 4.3/3.3*'

The OCM readings from the DCU in the DB need to be translated into a binary value that indicates whether there is an over-current or not. The used 'Dp function' is an evaluation of the logical expression:

▶ '*OCM_HYBRID_Vxxx' < OCM_HYBRID_THRESHOLD*'

The inhibit signal for the hybrid low voltage regulators comes from the CB, more explicitly, from the SPECS I/O registers. The right bit must be masked for each of the hybrids in a service box to read back the status coherently. This is done by means of a logical expression which is different for each partition of regulators; the following is just an example:

▶ *!(((p1[1]&128)==0)&&((p2[1]&128)!=0))*

The meaning of the items is the following:

LV_HYBRID			
Reg. name	Type	Mode	Description
ADC_OCM_HYBRID_Vana	DCUChannel		ADC channel to read the hybrid analog power regulator OCM signal
ADC_OCM_HYBRID_Vdig	DCUChannel		ADC channel to read the hybrid digital power regulator OCM signal
ADC_VOLTS_HYBRID_Vana	DCUChannel		ADC reading for the analog regulator output voltage (in the regulator side)
GAIN_VOLTS_HYBRID_Vana	float		The DCU nominal gain, 2.22 LSBs/mV
VOLTS_HYBRID_Vana	float		Voltage value obtained according to the formula
OCM_HYBRID_Vdig	int		Status of the OCM signal for the hybrid digital LVR ('1'☐ over current)
OCM_HYBRID_Vana	int		Status of the OCM signal for the hybrid analog LVR ('1'☐ over current)
OCM_HYBRID_THRESHOLD	int		Constant threshold value set to 600

INH_HYBRID_status	int	Read back status of the hybrid regulators inhibit signal
INH_HYBRID_set	int	Value that must be applied to the regulators, loaded from the recipe
mode	int	Hybrid regulators control enabled/disabled
status	int	Status of the hybrid regulators
errCount	int	Number of bad consecutive readings

10.3.1.10   TEMP_CHANNEL

It is used to get the temperature measurement from a DCU channel reading. The conversion formula that translates the ADC value to degrees is given by:

▶ *'DEGREES_TEMP' = 'ADC_TEMP' • 'CURVE_SLOPE' + 'CURVE_OFFSET'*

The '*CURVE_SLOPE*' and '*CURVE_OFFSET*' come from the CB calibration.

TEMP_CHANNEL			
Reg. name	Type	Mode	Description
ADC_TEMP	DCUChannel		ADC channel to read the temperature
DEGREES_TEMP	float		Value in degrees from the PT1000 sensor. dpFunction for the equation previously defined
CURVE_SLOPE	float		Calculated slope of the calibration curve for the conditioning and acquisition circuitry
CURVE_OFFSET	float		Calculated offset of the calibration curve for the conditioning and acquisition circuitry
hwAssigned	string		The name of the hardware to where the measurement belongs to (Ctrl Board name, hybrid name…)
mode	int		Measurement enabled/disabled
status	int		Status of the measurement
errCount	int		Number of bad consecutive readings

10.3.2 DU HARDWARE TYPES

These are the types used to model the hardware devices associated to a DU. They are composed of primitive types such as *integers*, *strings*, etc. and the base types explained before.

10.3.2.1 CONTROL_BOARD

As the name says, it models the Control Board. The CB houses elements that are used for different domains: DAQ, DCS LV, DCS TEMP… Without any kind of funded argument (as currently they do not exist but they did in the past) the CB was put in the DAQ domain.

The CB contains the base types *TTCRX_ST*, a *DELAY25*, two *SpecsMezzanine_ST* and other primitive elements that contain information about the CB and summarize data of several important signals. The *SpecsMezzanine_ST* represents the SPECS slave register and contrary to the other elements, holds information not belonging to the DAQ domain. The reason for this is that the I/O register controls and monitors various signals which belong in the *SpecsMezzanine_ST* and hence the CB. The way the bits of the I/O registers are read is by a 'Dp function' that masks the proper bit and assigns the result to an *integer* DPE. This is the case for all the '…_status' signals as seen in the table.

CONTROL_BOARD			
Reg. name	Type	Mode	Description
STsettings	string		*NOT USED*
TTCrq_RESET_status	int		Status of the TTCRx chip reset signal
TTCrq_QPLL_RESET_status	int		Status reset signal for the QPLL housed TTCRq. dpFunction to the bit of the specs register

DBs_QPLLs_reset_status	int		Status of the reset signal for the QPLLs of all DBs in a SB. dpFunction to the bit of the specs register
RST_GOLs_status	int		Status of the reset signal for the GOLs of all DBs in a SB. dpFunction to the bit of the specs register
L0ACCEPT_DE_Ctrl_status	int		Status of the LVDS driver DE signal (drive enable) for the L0 signal in the CB. dpFunction to the bit of the specs register
DRV_DOWN_EN_status	int		Status of the enable signal for the quadruple LVDS driver top (L0ACCEPT, L0RESET, CALIB0, ClockDes1), the one next to the backplane connector. dpFunction to the bit of the specs register
DRV_UP_EN_status	int		Status of the enable signal for the quadruple LVDS driver top (ClockDes2), the one in the CB end next to the service box connector. dpFunction to the bit of the specs register
TTCrqREADY_status	Int		Status of the TTCrq 'Ready' signal. dpFunction to the bit of the specs register
TTCrqLOCKED_status	int		Status of the TTCrq QPLL 'Locked' signal. dpFunction to the bit of the specs register
TTCrqERROR_status	int		Status of the TTCrq QPLL 'Error' signal. dpFunction to the bit of the specs register
ctrlBoardNumber	int		Control Board ID
lvcbAssigned	string		'LVCB' DU assigned to the CB
mode	int		Monitoring status of the DU: active or passive
status	int		Status of the DU
errCount	int		Number of bad consecutive readings in any of the previous signals
TTCrx	TTCRX_ST		The TTCrx device in the CB
Delay25	DELAY25		The Delay25 device
specsSlave1	SpecsMezzanine_ST		The Specs slave 1
specsSlave2	SpecsMezzanine_ST		The Specs slave 2

As the I/O register is placed within the CB, it is mandatory that the DUs of type *CONTROL_BOARD* have to exist before creating the rest. If they do not exist, the DUs used to control the LV and the system startup, that is, the *LVCB* ones, cannot be created as they need information from the I/O reg. The same happens with the *LV_PARTITION* and others.

10.3.2.2 DIGITIZER_BOARD

This is the type for the DB. In terms of real hardware, it includes the GOL devices (three in the IT and four in the TT) and one of the channels of the DCU since it is the only channel used for DAQ monitoring purposes. This channel monitors the DB QPLL locked status signal.

DIGITIZER_BOARD			
Reg. name	Type	Mode	Description
STsettings	string		*NOT USED*
cbAssigned	string		corresponding CB name of the service box for this digitizer board
ADC_QPLL	DCUChannel		DCU channel reading for the QPLL locked status signal
QPLL_LOCKED_status	int		status of the QPLL locked signal, i.e. binarization of the 'ADC_QPLL' reading with a dpFunction
status	int		Status of the DU
errCount	int		*NOT USED*
GOL1	GOL		GOL device 1
GOL2	GOL		GOL device 2
GOL3	GOL		GOL device 3

GOL4	GOL		GOL device 4. Only exists in the TT.

10.3.2.3 LADDER

It represents the hybrid. The motivation of this name has a historical background, coming from the fact that the IT hybrids are assembled in the so-called ladders and the IT system was the first one developed. It includes three or four Beetles for the IT and TT respectively. In addition it implements other DPEs that are foreseen to keep information about ladder, such as the name of the biasing channel, the LV channels…

LADDER			
Reg. name	Type	Mode	Description
HVchannel	string		name of the LV MARATON DU channel. *NOT USED*
HVjumperStatus	string		status of the jumper that switches the HV bias. *NOT USED*
cbAssigned	string		corresponding CB name of the service box for this ladder
LVstatus	int		*NOT USED*
status	int		Status of the DU
errCount	int		*NOT USED*
Beetle1	BEETLE_ST		Beetle device 1
Beetle2	BEETLE_ST		Beetle device 2
Beetle3	BEETLE_ST		Beetle device 3
Beetle4	BEETLE_ST		Beetle device 4. Only exists in the TT.

10.3.2.4 LVCB

It is needed to monitor the status of the 2.5 V, 3.3 V and 5 V LVRs in the CBs. It implements three elements of type *LV_REGULATOR,* namely *REGULATOR_2500mV*, REGULATOR_*3300mV* and *REGULATOR_5000mV*, which monitor the three working voltages. It gets the *OCM* signal status of the 2.5 V and 5 V regulators from the I/O register via a '_dp_fct' masking the right bit. The elements *errCount*, *status*, *mode* and *cbAssigned* play the same role as in the previous types. *lvChannelsAssigned* is an array with the two MARATON channel names that power the SB. Finally *specs1Init* and *specs2Init* were thought to check whether the SPECS slaves are initialized, but they are currently not being used.

LVCB			
Reg. name	Type	Mode	Description
OCM_REG_5000mV	int		OCM signal from 5V CB regulator
lvChannelsAssigned	dyn_string		Name of the LV MARATON channels assigned to this SB
cbAssigned	string		CB name assigned to this DU. There is a one to one matching
mode	int		Monitoring status of the DU: active or passive
status	int		Status of the DU
errCount	int		Number of bad consecutive readings in any of the previous signals
specs1Init	int		Indicates whether the SPECS1 in the CB has been initialized
specs2Init	int		Indicates whether the SPECS2 in the CB has been initialized
REGULATOR_2500mV	CB_REGULATOR		2.5V regulator measurement
REGULATOR_3300mV	CB_REGULATOR		3.3V regulator measurement
REGULATOR_5000mV	CB_REGULATOR		5V regulator measurement
REGULATOR_2500mV_REF	CB_REGULATOR		5V voltage reference measurement

10.3.2.5 LV_PARTITION

This type is used to control and monitor the regulators that power the Digitizer Boards and hybrids. The LVR are arranged in partitions as explained in chapter 4 and this type represents a partition. In each partition there are 4 (IT) or 3 (TT) pairs of LVR that power a hybrid each. Besides, there are a set of regulators for the DBs. The hybrid regulators are represented by sub-elements of type *LV_HYBRID*. The *OCM_DBs_...*, *INH_DBs...* DP elements are used for the DB regulators.

LV_PARTITION			
Reg. name	Type	Mode	Description
OCM_DBs_5V	int		OCM signal from 5V CB regulator
OCM_DBs_33V	int		OCM signal from 3.3V CB regulator
OCM_DBs_25Vn1	int		OCM signal from 2.5V CB regulator #1
OCM_DBs_25Vn2	int		OCM signal from 2.5V CB regulator #2
INH_DBs_status	int		Status of the inhibit signal for the DB regulators of this partition
INH_DBs_set	int		Value at which the inhibit signal should be set
INH_DBs_mode	int		Indicates whether inhibit signal control is enabled/disabled
lvcbAssigned	string		'LVCB' DU assigned to the CB
cbAssigned	string		CB name assigned to this DU. There is a one to one matching
mode	int		Monitoring status of the DU: active or passive
status	int		Status of the DU
errCount	int		Number of bad consecutive readings in any of the previous signals
LV_HYBRIDn1	LV_HYBRID		Set of regulators for the hybrid #1 of the partition
LV_HYBRIDn2	LV_HYBRID		Set of regulators for the hybrid #2 of the partition
LV_HYBRIDn3	LV_HYBRID		Set of regulators for the hybrid #3 of the partition
LV_HYBRIDn4	LV_HYBRID		Set of regulators for the hybrid #4 of the partition. Only exists in the IT.

10.3.2.6 TEMP_BOX

Used for the monitoring of the PT1000 temperature sensors inside the detector box, which are read out by the DCU in one of the SPECS slaves. Each CB is able to read up to four box temperatures, which matches nicely with the number of sensors in each IT box. For this reason, this data type consists of four temperature channel base types. In addition it implements the needed elements for the DU control and another one that provides the average value of the four temperature readings. There is also an element that contains the name of the DU of *LVCB* type associated to the *TEMP_BOX*, which will be used to update the status of the box temperatures in the event of a problem with the *LVCB*.

TEMP_BOX			
Reg. name	Type	Mode	Description
average_temp	float		Average temperature in the detector box
lvcbAssigned	string		'LVCB' DU assigned to the CB
mode	int		Monitoring status of the DU: active or passive
status	int		Status of the DU
errCount	int		*NOT_USED*
TEMPERATUREn1	TEMP_CHANNEL		Temperature of PT1000 sensor #1 in box
TEMPERATUREn2	TEMP_CHANNEL		Temperature of PT1000 sensor #2 in box
TEMPERATUREn3	TEMP_CHANNEL		Temperature of PT1000 sensor #3 in box
TEMPERATUREn4	TEMP_CHANNEL		Temperature of PT1000 sensor #4 in box

10.3.2.7 TEMP_PARTITION

It represents the temperature readings from the PT1000 in the hybrids of a partition. The type implements three or four objects of base type *TEMP_CHANNEL* for the readings depending on the detector, TT or IT.

TEMP_PARTITION			
Reg. name	Type	Mode	Description
lvPartitionStatus	string		*NOT_USED*
lvPartition	string		Name of the LV partition matching with this DU
mode	int		Monitoring status of the DU: active or passive
status	int		Status of the DU
errCount	int		*NOT_USED*
TEMPERATUREn1	TEMP_CHANNEL		Temperature of PT1000 sensor #1 in box
TEMPERATUREn2	TEMP_CHANNEL		Temperature of PT1000 sensor #2 in box
TEMPERATUREn3	TEMP_CHANNEL		Temperature of PT1000 sensor #3 in box
TEMPERATUREn4	TEMP_CHANNEL		Temperature of PT1000 sensor #4 in box. Only exists in the IT

10.3.2.8 TEMP_SVCEBOX

There are three temperature sensors placed in the service box, two PT1000 in the backplane and one PT100 in the rear side of the CB. As the others, they are read by specific DCU channels in the CB.

TEMP_SVCEBOX			
Reg. name	Type	Mode	Description
lvcbAssigned	string		'LVCB' DU assigned to the CB
mode	int		Monitoring status of the DU: active or passive
status	int		Status of the DU
errCount	int		*NOT_USED*
BACKPLANE_TEMP1	TEMP_CHANNEL		Temperature of PT1000 sensor #1 in SB backplane
BACKPLANE_TEMP2	TEMP_CHANNEL		Temperature of PT1000 sensor #2 in SB backplane
CB_REGULATOR_TEMP	TEMP_CHANNEL		Control Board regulator PT100 temperature sensor

10.3.2.9 HUMIDITY

The output voltage from the HMX2000 sensor has to be translated to relative humidity measurement. This type includes an object of base type *HUM_CHANNEL* that provides the sensors reading. The environmental temperature and the calibration information of the HMX2000 sensors are also needed in order to obtain the RH measurement. The temperature we get it in *tempValue* and the calibration functions information in *SLOPE_V*, *SLOPE_TEMP* and *OFFSET_RH*. The conversion formula is:

▶ *'HUM_CHANNEL' = 20+((P1*1000-P2-(P3*P4))/P5)*

- ▪ *P1 = HMX2000 voltage reading*
- ▪ *P2 = 'OFFSET_RH'*
- ▪ *P3 = 'SLOPE_TEMP'*
- ▪ *P4 = 'tempValue'*
- ▪ *P5 = 'SLOPE_V'*

In addition, also the dewpoint is calculated from the temperature and humidity readings using a complex formula [57]:

▶ *(243.12*((17.62*p1)/(243.12+p1)+(((log(p2)/log(10))-2)/0.4343))/(17.62-((17.62*p1)/(243.12+p1)+(((log(p2)/log(10))-2)/0.4343))))*

TEMP_BOX			
Reg. name	Type	Mode	Description
tempAssigned	string		Temperature measurement DPE assigned to correct the humidity measurement
tempValue	float		Temperature value used to correct the humidity measurement
lvcbAssigned	string		'LVCB' DU assigned to the CB
tempCoolingPlant	float		Temperature value at which the cooling plant is set
dewPointTempMargin	float		Margin allowed for the dewPoint
BOX_RH	float		Relative humidity measurement
dewPoint	float		Dew point obtained inside the box
SLOPE_V	float		Slope in V from the calibration plane
SLOPE_TEMP	float		Slope in TEMP from the calibration plane
OFFSET_RH	float		Offset value from the calibration plane
mode	int		Monitoring status of the DU: active or passive
status	int		Status of the DU
errCount	int		*NOT_USED*
HMX2000_MEAS	HUM_CHANNEL		Output value from the humidity channel measurement
TEMP_HMX2000	TEMP_CHANNEL		Temperature measurement made in the HMX2000 sensor. Only for the TT

10.4 THE IT HIERARCHY

The description of the IT hierarchy will be done in a domain by domain basis, showing the naming convention and the distribution of nodes among the systems. In first place the logical division in layers for the different domains is described and then deployment of each layer. The coloring scheme of the objects indicates whether it is CU (blue), LU (light blue) or DU (grey), and the background color indicates the system where the objects are deployed.

The Figure 10.5 depicts the top nodes' deployment. The *ONLINE RESOURCES* description will not reported in here as this part is centrally managed by the LHCb group. These resources correspond to a slice of the TFC, HLT, STORAGE and MONITORING, which is needed in order to perform a data acquisition. They need to be allocated when taking control of the system and the necessary infrastructure for fast signaling, data filtering, storage and monitoring will be provided by them.

In the following subsections we will see that the names of the hardware types created with the 'Hw tool' do not necessarily match with the DU types names, though there is a univocal relationship between them. The reason has to do with the way the whole framework is implemented. Furthermore, in the following figures we will use a naming convention which differs slightly from the one used in the real implementation. The convention for the 'Hw tool' vs. DU names is:

- *CONTROL_BOARD ↔ ControlBoard*
- *DIGITIZER_BOARD ↔ DigBoard*
- *LADDER ↔ Ladder*
- *TELL1_ST ↔ TELL1DU*
- *LVCB ↔ LVCB*
- *LV_PARTITION ↔ LVPartition*
- *TEMP_BOX ↔ TEMP_BOX*
- *TEMP_PARTITION ↔ TEMP_PARTITION*
- *TEMP_SVCEBOX ↔ TEMP_SVCEBOX*
- *HUMIDITY ↔ HUMIDITY*
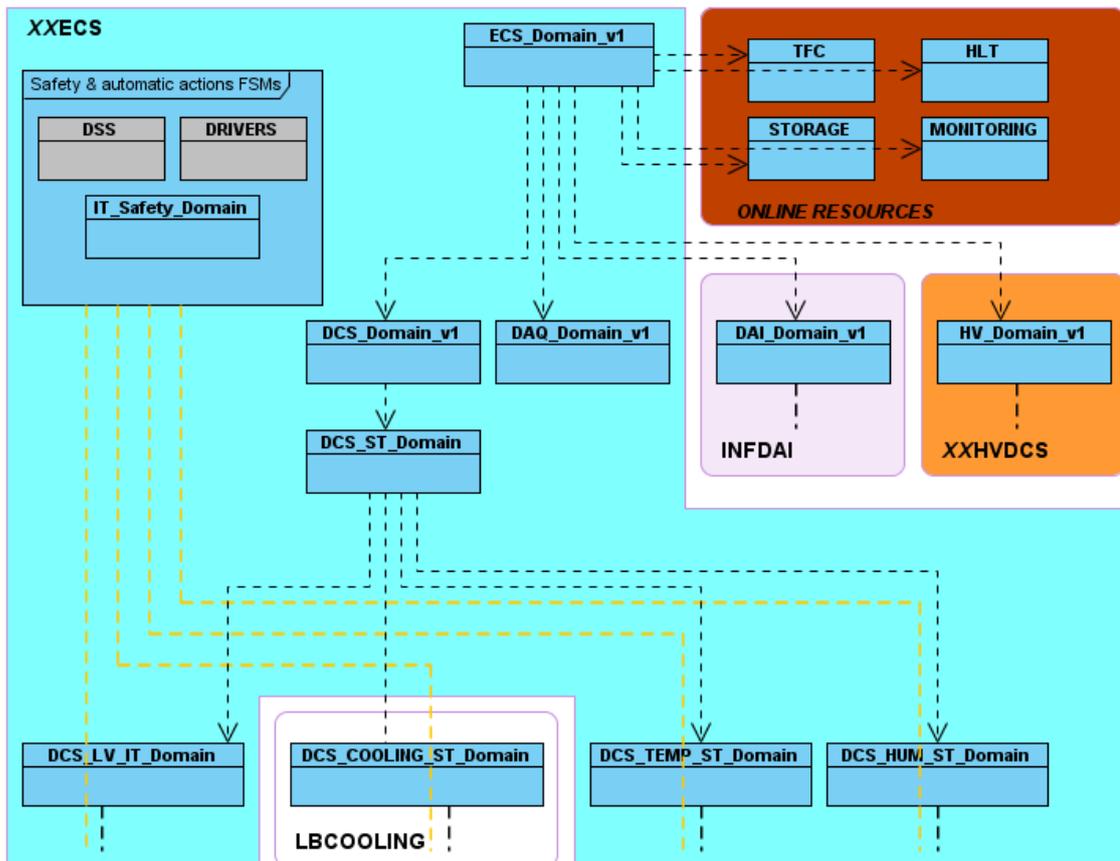
Figure 10.5: Deployment of the IT top nodes.

10.4.1 DAQ DOMAIN DEPLOYMENT

The top and half-stations DAQ nodes are deployed in the *XXECS* system, where '*XX'* is a wildcard indicating the sub-detector name (IT or TT). The system is down split into TELL1 and FE domains for each of the half-stations. The TELL1 domain is deployed in the *XXDAQTELL* system and consists of CU that holds the TELL1 DUs. The FE domain, which represents a FE box, is sub-divided into services boxes which are in turn made up of a *ControlBoard* DU and the partitions of DBs and hybrids. Each of the partitions, implemented as LUs, is composed by DBs (of type *DigBoard*) and Hybrids (of type *Ladder*).

10.4.1.1 THE DAQ TOP DOMAIN

The DAQ domain is first divided in six half-stations using the following naming:

▶ *"IT_DAQ_$i": $i identifies the half-station: "ST1A", for half-station 1 access side; "ST1C" for half-station 1 cryo (cryogenics) side; "ST2A" (half-station 2 access); "ST2C"; "ST3A"; "ST3C" (see).*

10.4.1.2 THE HALF-STATIONS

Each of the half-stations is sub-divided in the two corresponding boxes and a TELL1 control domain. The naming convention rule for this layer consists of adding a suffix to the parent CU name:

▶ *"_TELL1": for the Tell1s CU domain.*

▶ *"_FE_BOX$": where the value of $ depends on the parent half-station name: if the half-station belongs to the access side (ex, "IT_DAQ_ST2A") the possible values for $ are "A" (Access) and "B" (Bottom); if the half-station belongs to the cryo side (ex, "IT_DAQ_ST3C") the two values are "T" (Top) and "C" (Cryo).*
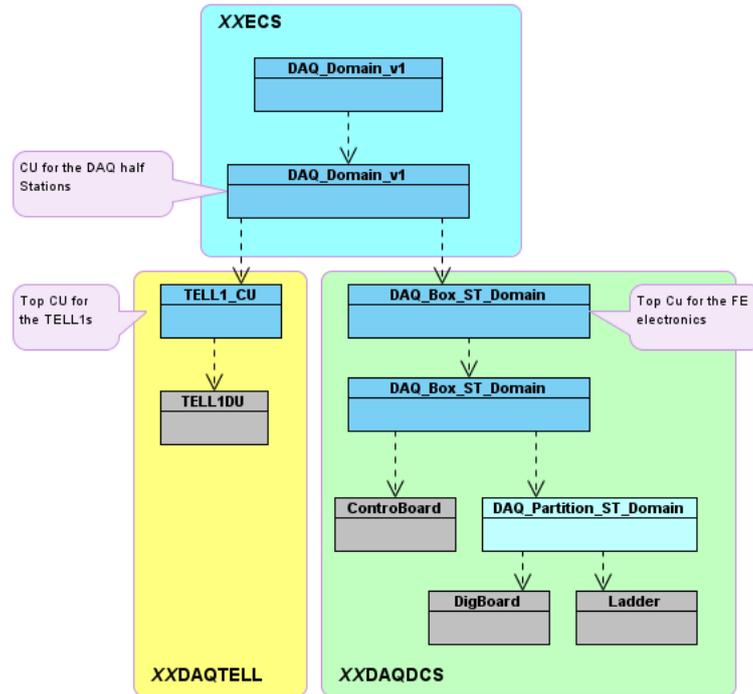
Figure 10.6: DAQ domain hierarchy deployment.
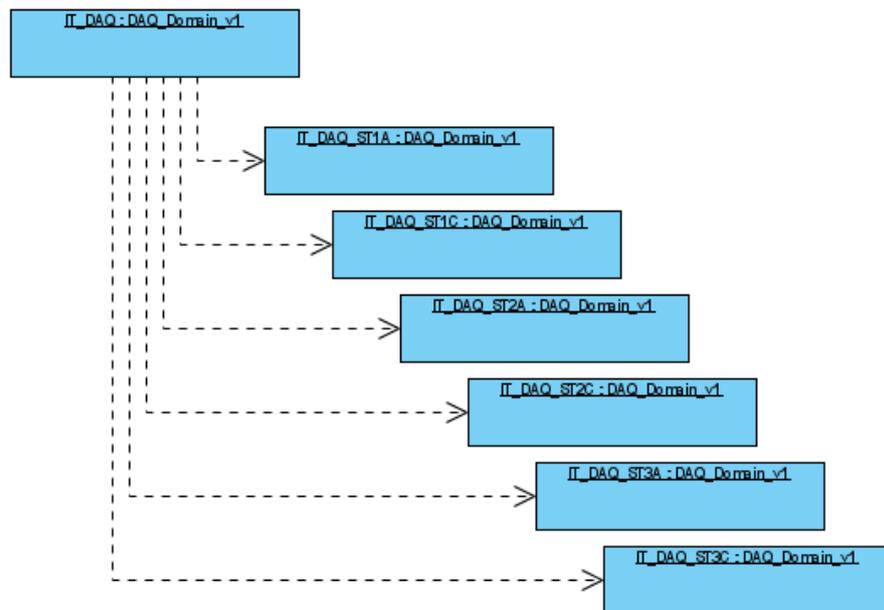
Figure 10.7: IT_DAQ domain split up into six half-stations.
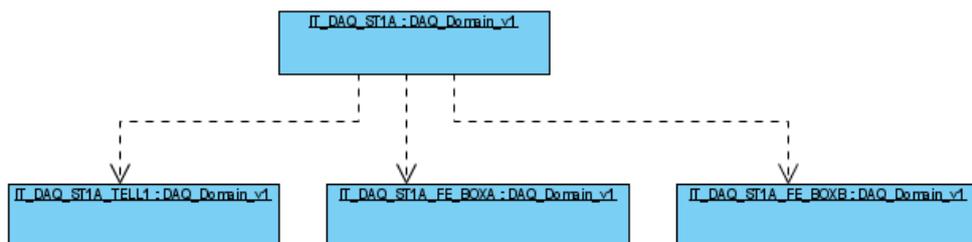
Figure 10.8: Half-station DAQ subdivision.

10.4.1.3 THE TELL1S

Seven TELL1 are needed for the data acquisition of an IT half-station. The TELL1 names correspond to their host names:

▶ *"ITTELL$i": $i is the TELL1 index that matches with the last word of the TELL1 IP address.*

The correspondence between half-stations and TELL1 indexes is defined in the table:

"ST1A"	8,9,10,11,12,13,14.	**"ST1C"**	1,2,3,4,5,6,7
"ST2A"	22,23,24,25,26,27,28.	**"ST2C"**	15,16,17,18,19,20,21
"ST3A"	36,37,38,39,40,41,42	**"ST3C"**	29,30,31,32,33,34,35

Table 10.1: TELL1 to half-stations correspondence.
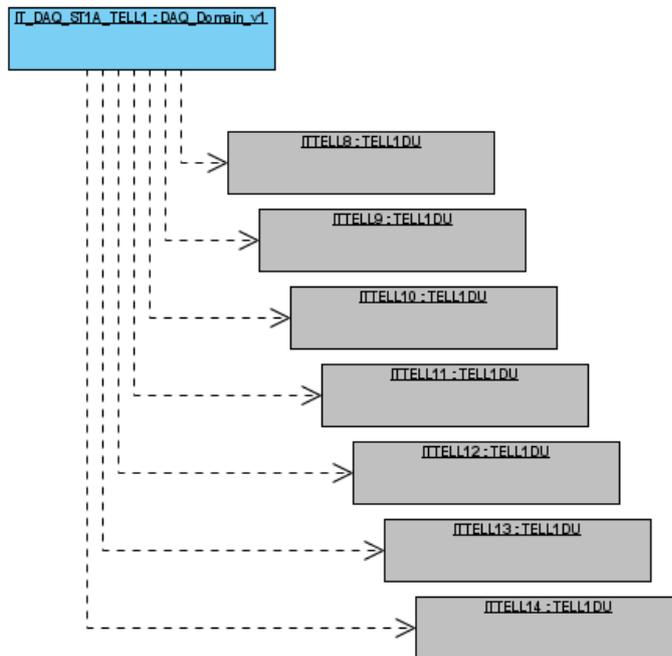
Figure 10.9: TELL1 CU.

10.4.1.4 THE FE BOXES

Each IT detector box is read out by two service boxes. The naming convention for the SBs consists of removing the *"_FE"* and adding:

▶ *"_SB$i": $i ranges from 1 to 4 indicating the SB location starting to count from the beam pipe outwards. Indexes 1 and 2 correspond to the B (Bottom) and C (Cryo) detector boxes and 3 and 4 correspond to the A (Acess) and T (Top) boxes.*
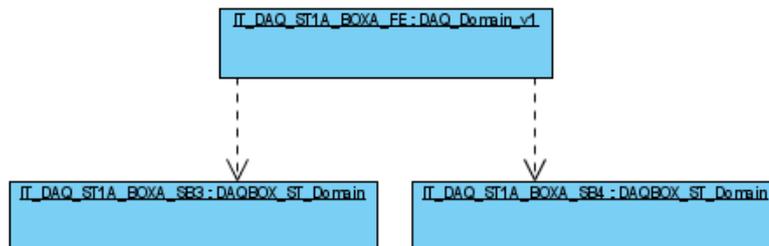
Figure 10.10: FE DAQ IT detector box split into two service boxes.

10.4.1.5 THE FE SERVICE BOXES

Each SB can house either twelve or sixteen DBs and hence read out twelve or sixteen hybrids. This implies that some of the SBs will consist of three or four FE readout partitions, each made of four DBs/hybrids. The two SBs belonging to a detector box sum up a total of seven partitions corresponding to twenty-eight hybrids/DBs.

The naming for the FE partitions (which are implemented as LUs) consists of appending the following to the service box name:

► *"_P$j": $j represents the ECS partition and can range from 1 to 7. Three of the partitions belong to one SB and four to the other, as in the figure.*

The naming convention for the Control Board DUs is:

► *"IT_STi_BOXj_SB$m_CB" for the Control Board DUs:*

  ▪ *$i represents the half-Station ID: "1A", "1C", "2A", "2C", "3A", "3C".*

  ▪ *$j represents the half-Station ID: "A" (access), "B" (bottom), "C" (cryo), "T" (top).*

  ▪ *$m ranges from 1 to 4 indicating the service box position starting from the beam pipe.*
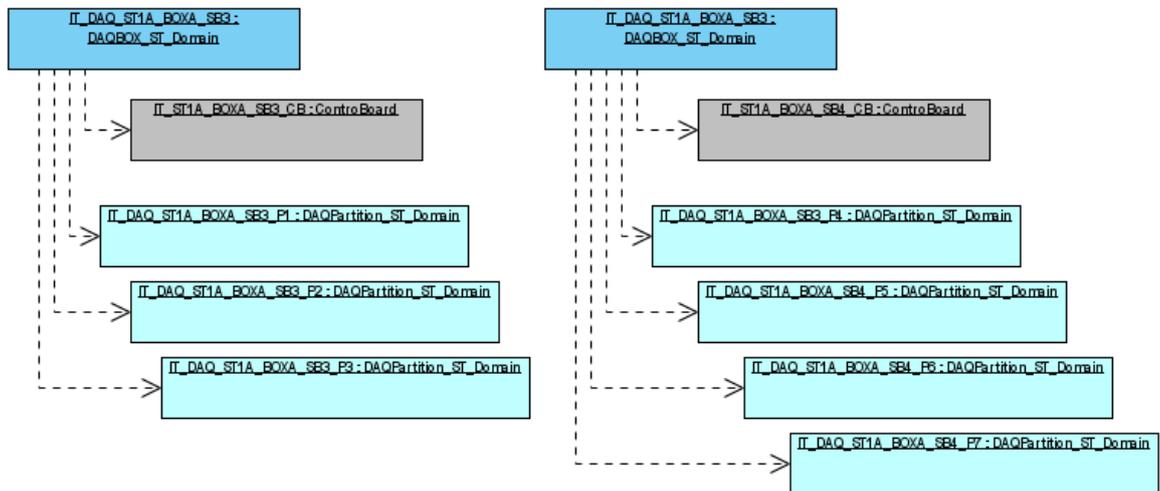
Figure 10.11: FE box sub-division into one CB and various partitions.

10.4.1.6 THE FE PARTITIONS

Each FE partition consists of four DBs and four hybrids.

The naming convention for the Digitizer Boards is the following:

► *"IT_STi_BOXj_SBm_DBn":*

  ▪ *$i represents the half-Station ID: "1A", "1C", "2A", "2C", "3A", "3C".*

  ▪ *$i represents the half-Station ID: "A" (access), "B" (bottom), "C" (cryo), "T" (top).*

  ▪ *$m ranges from 1 to 4 indicating the Svce box position starting from the beam pipe.*

  ▪ *$n ranges from 1 to 16 indicating the slot where the Dig Board is plugged in the Svce Box (see Figure 10.12).*

For the hybrids:

► *"IT_STi_BOXj_Lm_Sn" for the hybrids:*

  ▪ *$i represents the half-Station ID: "1A", "1C", "2A", "2C", "3A", "3C".*

  ▪ *$j represents the half-Station ID: "A" (access), "B" (bottom), "C" (cryo), "T" (top).*

  ▪ *$m indicates the layer location of the ladder in the box. I can be "X1", "U", "V" or "X2".*

  ▪ *$n ranges from 1 to 7 indicating the sector location of the ladder in the box (see Figure 10.12).*

10.4.2 THE DCS DOMAIN

In Figure 10.5 the structure of the DCS domain was first introduced. The DCS has to deal with all the components related to the correct detector operation. This includes monitoring sensors related to temperature, humidity and also the control of the low voltage supply to the electronics. It also contains the supervision of systems like the cooling systems. For this reason there are four sub-domains below DCS: DCS_LV, DCS_TEMP, DCS_HUM and DCS_COOLING. From them, just the cooling CU is deployed in a separate system. It is also important to note that between the top DCS domain and the four sub-domains there is intermediate one. It is needed because the DCS domain is a standard CU type that is common for all LHCb detectors and its behaviour should not be modified. This type does not take into account the needs of the ST regarding to the sequencing of operations when turning on the detector: before the temperature and humidity monitoring is started, the LV must be switched on. It has to be like that because the monitoring and DAQ electronics are implemented in the same boards (CB and DBs) and hence they use the same power sources. In addition, this intermediate state implements a special "standby" state for the FE electronics named *HYBRIDS_OFF* which basically consists on having the monitoring electronics of the SBs powered on, the hybrids off and the monitoring services of the DCS_TEMP and DCS_HUM domain running.
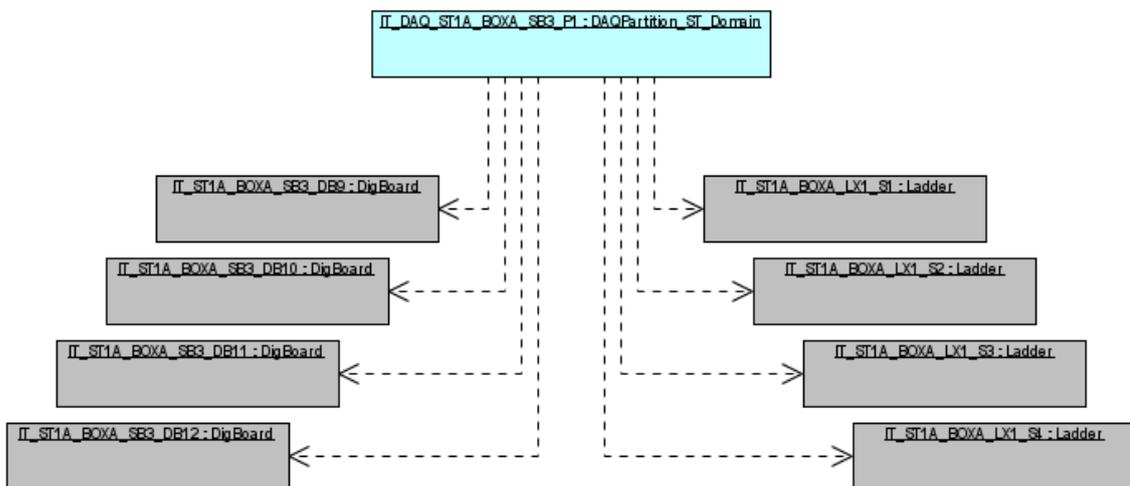
Figure 10.12: IT partition with four DBs and four hybrids.

10.4.3 DCS LV DOMAIN DEPLOYMENT

The DCS_LV domain is part of the DCS. As depicted in Figure 10.13, it is deployed into three systems: *XXECS*, *XXHVDCS* and *XXDAQDCS*. The first of the systems hosts the top level CUs, i.e. the top IT_DCS_LV together with the sub-layer of half-stations. The half-stations node controls the MARATON power supplies and the service box power regulators corresponding to that station. The MARATON control is deployed in the *XXHVDCS* system whereas the regulators control is in *XXDAQDCS*.

The MARATON CU controls all the channels that power a half-station. From this CU hangs the LU representing the two MARATON channels corresponding to a service box.

The top CU deployed in *XXDAQDCS* represents the regulator power control for a detector box and hence is further sub-divided into two service boxes. The SBs are implemented as LUs and control and monitor both the CB and partitions power regulators.

10.4.3.1 THE DCS LV DOMAIN

As the DAQ domain, it is divided into six half-stations using the following naming:

▸ *"IT_DCS_LV_ST$i": $i identifies the half-station: "1A", "1C", "2A", "2C", "3A", "3C" (see Figure 10.14).*

10.4.3.2 THE DCS HALF-STATION LV DOMAIN

Each of the IT half-stations is controlled by the same SPECS link so the 4 service boxes belonging to these partitions must be always powered. If one of them is not powered the SPECS link chain is cut, so it only makes sense that all the MARATON channels that power the half-stations are all on or off. This is why this

domain is subdivided into a MARATON CU and two detector boxes CUs, each one controlling the LV electronics for each of the boxes.

The names of the MARATON and the boxes append the following to the half-station name:

► *"_MTN" for the MARATON CU domain.*

► *"_BOX$" where the value of $ depends on the parent half-station name: if the half-station belongs to the access side (ex, "IT_DCS_LV_ST2A") the possible values for $ are "A" (Access) and "B" (Bottom); if the half-station belongs to the cryo side (ex, "IT_DCS_LV_ST3C") the two values are "T" (Top) and "C" (Cryo).*
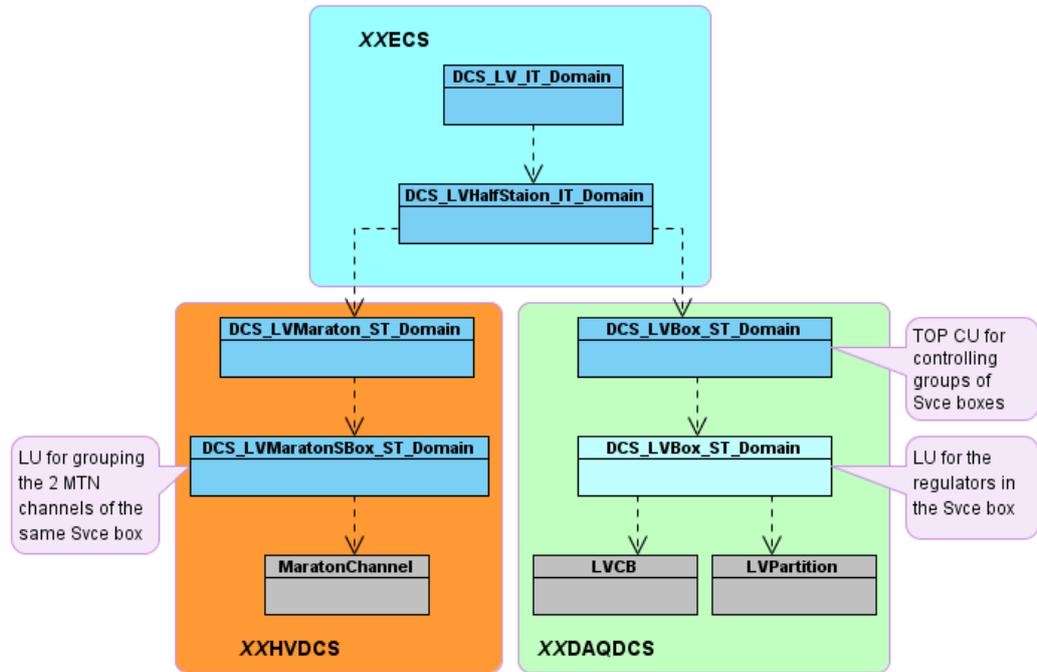
Figure 10.13: DCS low voltage hierarchy deployment.
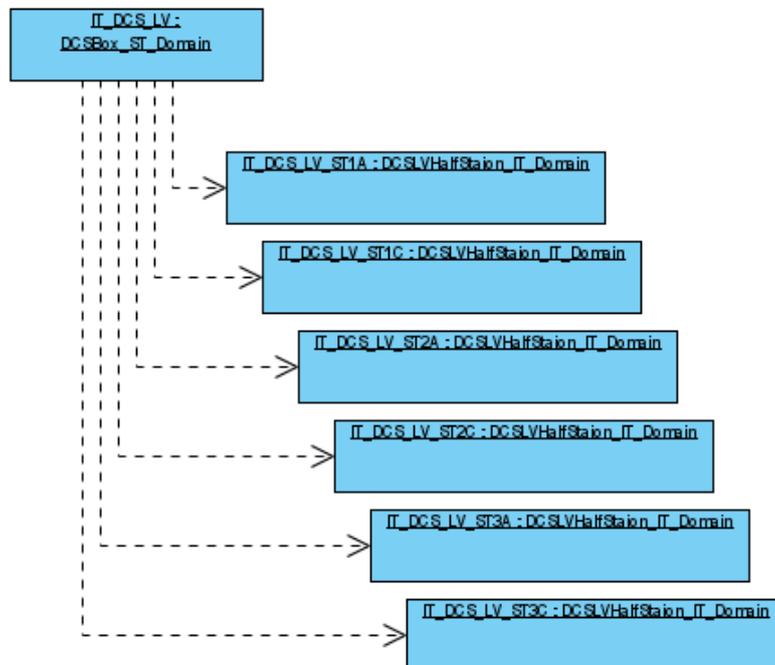
Figure 10.14: IT_DCS_LV domain split up into six half-stations.

- 199 -
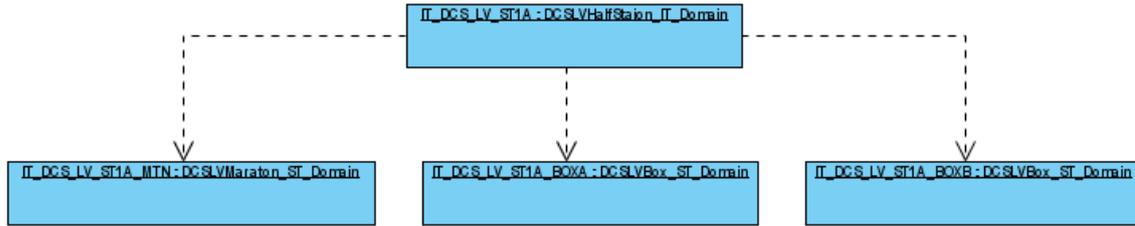
Figure 10.15: subdivision of the half-station low voltage domain.

10.4.3.3 THE DCS MARATON DOMAIN

It controls all the MARATON channels belonging to the same half-station. The LV channels are grouped in pairs of two into service boxes. The service boxes nodes are implemented as LUs.

The naming of the LUs is built from the parent name:

▶ *"_BOXi_SBj" for the LUs. $i represents the box name ("A", "B", "C", "T") and $j ranges from 1 to 4 indicating the Svce box position.*

The DU name for the MARATON low voltage channels is as follows:

▶ *"IT/LV/ST$i/BOX$j/SB$m/Ch$n":*
  - *$i represents the half-station: "1A", "1C", "2A", "2C", "3A", "3C".*
  - *$j represents the box name "A", "B", "C", "T".*
  - *$m is for the service box position (from 1 to 4).*
  - *$n can either be 1 (2.5 V channel) or 2 (5 V channel).*
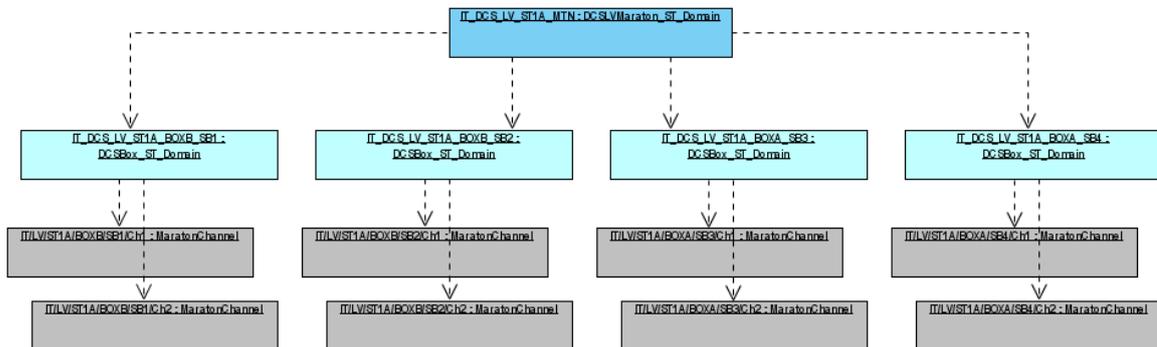
Figure 10.16: MARATON sub-hierarchy.

10.4.3.4 THE IT DCS BOX LV DOMAIN

The detector box is sub-divided into two service boxes. One of the service boxes houses twelve DBs/hybrids and the other one sixteen, which in other words means three or four partitions. In addition, there is a DU per SB of type *LVCB* that carries out the tasks of initializing some parameters in the CB and monitor the CB regulators. The Figure 10.17 shows an example of it.

The naming for the service boxes appends the following to the box names:

▶ *"_SB$i": $i ranges from 1 to 4 indicating the Svce box position. Indexes 1 and 2 correspond to the "B" (bottom) and "C" (cryo) detector boxes and 3 and 4 correspond to the "A" (access) and "T" (top) boxes.*

The DUs naming is:

▶ *"IT_STi_BOXj_SB$m_LVCB" for the Control Board low voltage DUs:*
  - *$i represents the half-station: "1A", "1C", "2A", "2C", "3A", "3C".*
  - *$j represents the box name "A", "B", "C", "T".*
  - *$m is for the service box position (from 1 to 4).*

▶ *"IT_STi_BOXj_SBm_LVPn" for the LV partitions:*

- *$i represents the half-station: "1A", "1C", "2A", "2C", "3A", "3C".*

- *$j represents the box name "A", "B", "C", "T".*

- *$m is for the service box position (from 1 to 4).*

- *$n ranges from 1 to 7 indicating the ECS partition as shown in the figure.*
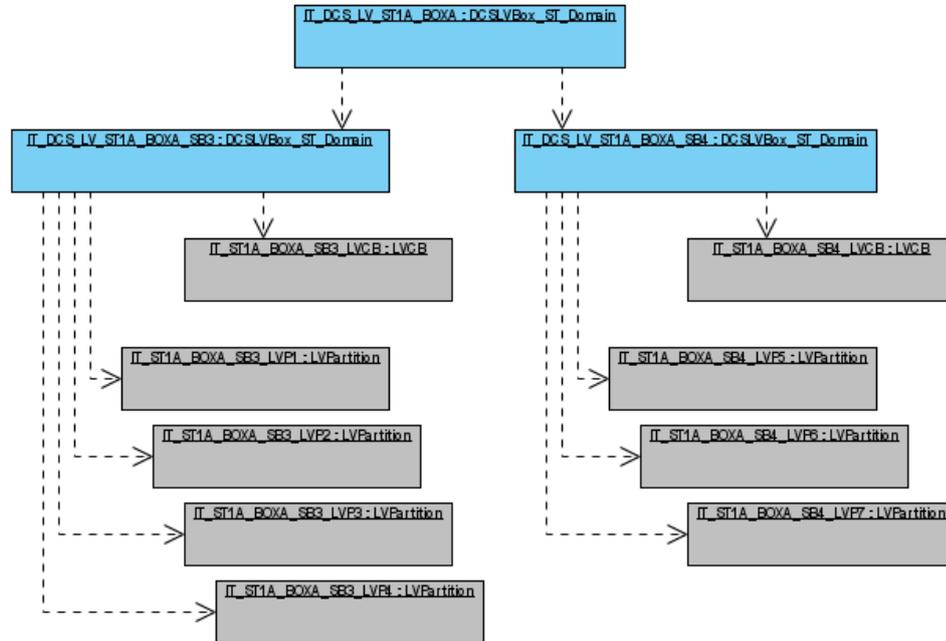
Figure 10.17: IT box low voltage hierarchy.

10.4.4 DCS TEMPERATURE DOMAIN DEPLOYMENT

It follows a structure similar to the low voltage with the exception that there is no sub-split into different systems at the half-station level (Figure 10.18).

At the detector box level hangs a separate DU that represents the readings from the box internal environmental temperature.

10.4.4.1 THE DCS TEMPERATURE TOP DOMAIN

As the others, the first sub-division is into half-stations (Figure 10.19). The naming is similar to the the LV domain.

▶ *"IT_DCS_TEMP_ST$i": $i identifies the half-station: "1A", "1C", "2A", "2C", "3A", "3C".*

10.4.4.2 THE DCS TEMPERATURE HALF-STATION DOMAIN

It consists of two CUs representing the detector boxes (Figure 10.20). The names of the boxes append the following to the half-station name:

▶ *"_BOX$" where the value of $ depends on the parent half-station name: if the half-station belongs to the access side (ex, "IT_DAQ_ST2A") the possible values for $ are "A" (Access) and "B" (Bottom); if the half-station belongs to the cryo side (ex, "IT_DAQ_ST3C") the two values are "T" (Top) and "C" (Cryo).*

10.4.4.3 THE DCS TEMPERATURE BOX DOMAIN

As for the other domains, this node is sub-divided into two service boxes (Figure 10.21). It has the peculiarity that one DU hangs directly from the box level. This DU represents the environmental temperature readings of the detector box. Then, at the SB level, there is a DU that corresponds to the two monitored temperatures in the SB backplane and one on the CB. Finally there are other three/four DUs per SB that represent the four hybrid temperatures of a partition.
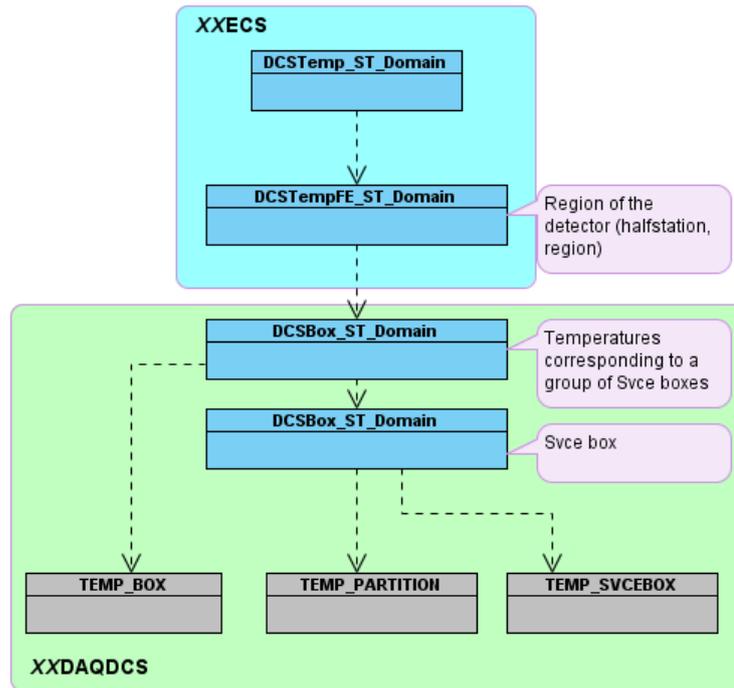
Figure 10.18: DCS temperature domain deployment.
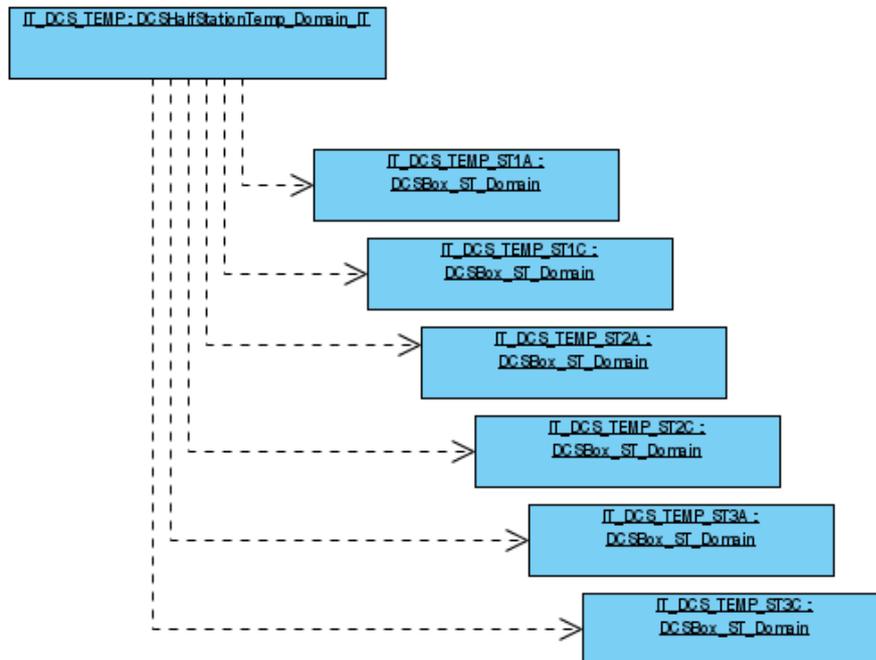
Figure 10.19: DCS temperature domain splitting into half-stations.
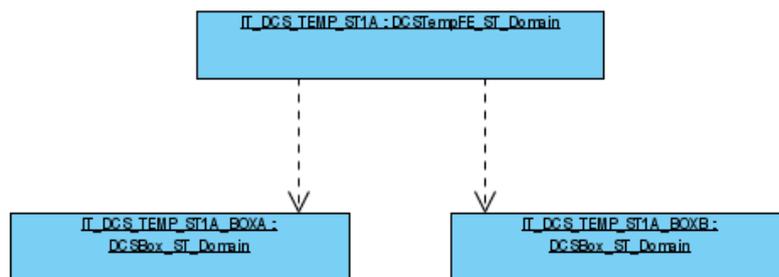
Figure 10.20: Subdivision of the half station low voltage domain into boxes.
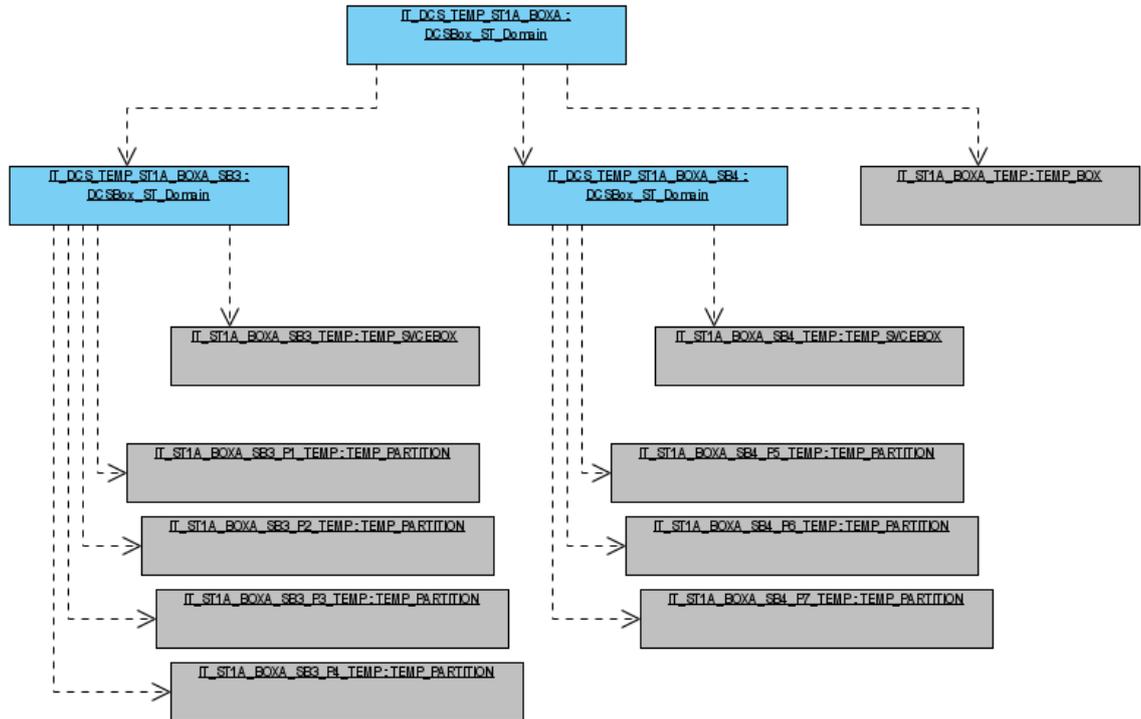
Figure 10.21: IT DCS box temperatures hierarchy.

10.4.5 DCS HUMIDITY DOMAIN DEPLOYMENT

The DCS humidity domain is much simpler than the previous. As there is only a humidity sensor per box, the chosen implementation consists of twelve humidity DUs just below the DCS_HUM domain (Figure 10.23).
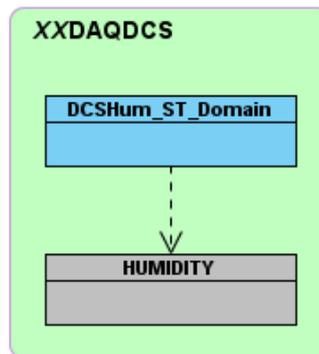
Figure 10.22: DCS humidity deployment.

The naming of the DUs is:

▶ *"IT_STi_BOXj_HUM":*

  ▪ *$i identifies the half-station: "1A", "1C", "2A", "2C", "3A", "3C".*

  ▪ *$j represents the box name "A", "B", "C", "T".*

10.4.6 DCS COOLING DOMAIN DEPLOYMENT

The cooling system control is running in an external computer. A specific project on that computer carries out the control of all the LHCb cooling plants. For each of the systems there is a CU to perform the cooling plant control tasks. Underneath the control CU there is a cooling plant DU and several temperature loops, one per each cooling loop with associated temperature monitoring.

10.4.7 HV DOMAIN DEPLOYMENT

The HV control domain is a four layers hierarchy (Figure 10.25): the top IT_HV, the half-stations level, the detector boxes and finally the CAEN HV channels. The SB level was skipped because the HV distribution is completely unrelated from the FE electronics grouping. The whole HV FSM hierarchy is running in the *XXHVDCS* system.
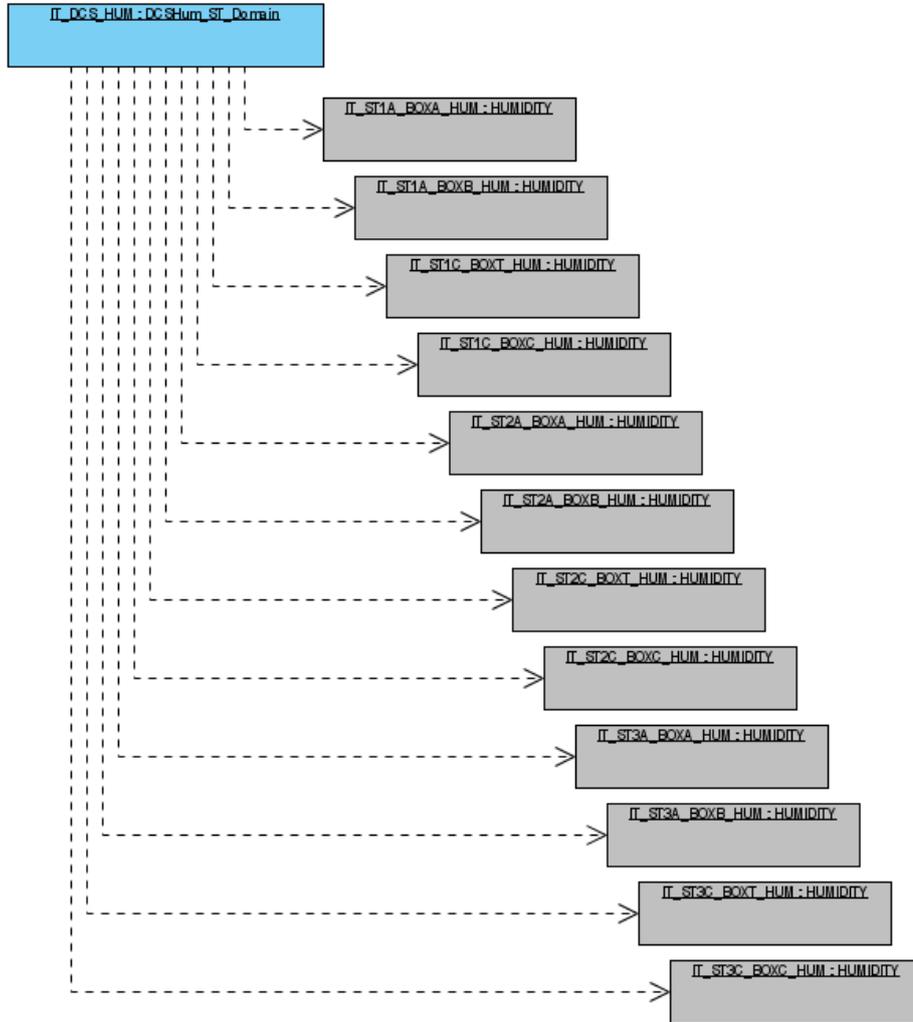
Figure 10.23: DCS humidity domain FSM.
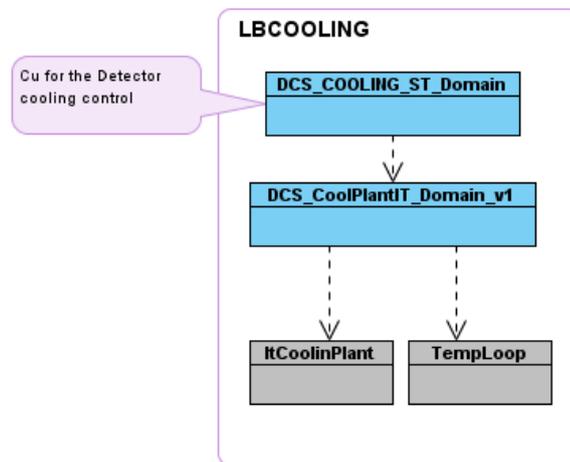
Figure 10.24: DCS cooling deployment.
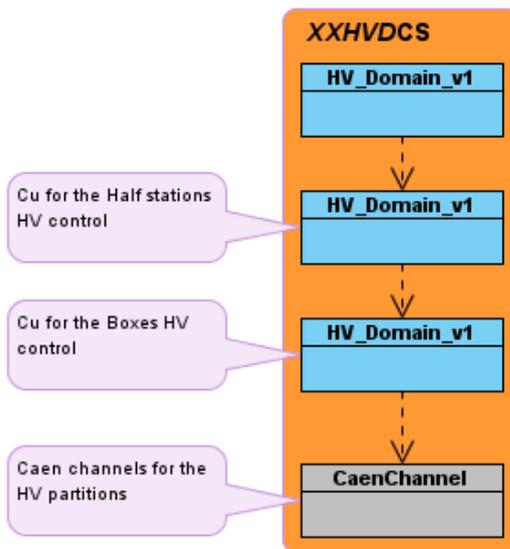
Figure 10.25: IT HV domain deployment.

10.4.7.1 THE HV TOP DOMAIN

The same philosophy is kept, the first subdivision is into half-stations (Figure 10.26) and the used naming is:

▶ *"IT_HV_ST$i": $i identifies the half-station"1A", "1C", "2A", "2C", "3A", "3C".*

10.4.7.2 THE HV HALF-STATION domain

This domain splits off into the two detector boxes (Figure 10.27). Naming:

▶ *"_BOX$" is appended to the parent name. The value of $ depends on the parent half-station name. The possible values for $ are "A" (access) and "B" (bottom) for the access side half-station or "T" (top) and "C" (cryo) for the cryo side.*
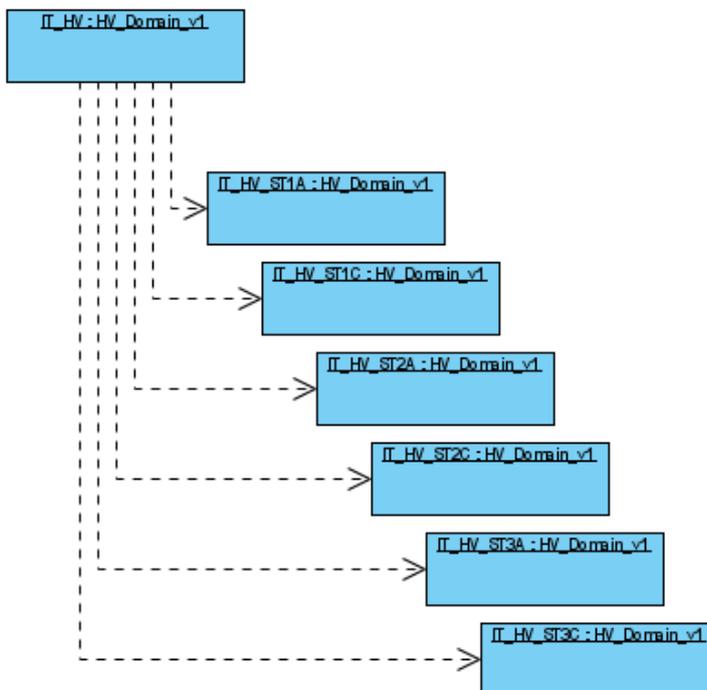
Figure 10.26: HV top domain FSM.
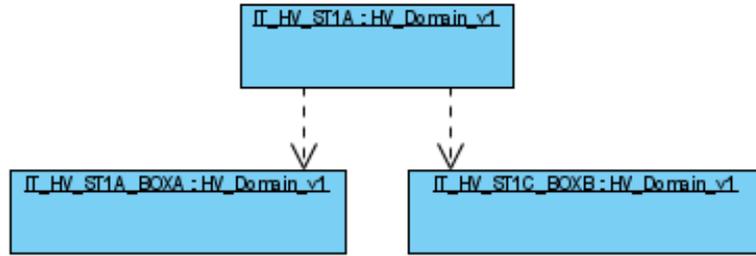
Figure 10.27: The HV box domain FSM.

10.4.7.3 THE HV BOX DOMAIN

In the HV domain this is the last bottom layer. It holds the seven HV channel associated to a detector box. Each HV channel biases a partition of hybrids (Figure 10.28).

The DU names are built as follows:

▶ *"IT/HV/ST$i/BOX$j/P$m" for the MARATON channels.*

- ▪ *$i identifies the half-station: "1A", "1C", "2A", "2C", "3A", "3C".*
- ▪ *$j represents the box name "A", "B", "C", "T".*
- ▪ *$m ranges from 1 to 7 indicating the ECS partition.*

10.5 THE TT HIERARCHY

The TT detector geometry differs from the IT and therefore the control system cannot just be replicated. In fact, the grouping of the different parts of the detector is not as obvious as in the IT (detector ▶ half-stations ▶ boxes ▶ service boxes ▶ partitions). The TT detector is made of two stations ('TTa' and 'TTb') with two sensor layers each (*x1-u* and *v-x2*). The hybrids house four Beetles (instead of three) and the readout service boxes are placed in four towers. Each of the stacks consists of six service boxes and serves a quadrant of the detector. Therefore there is not a matching between service box groups and detector stations as in the IT. In addition there is a limitation imposed on how the readout sector groups map into TELL1s, designed to distribute the data load. Finally, also cable length limitations have imposed restriction on how the grouping of readout sectors match to service boxes.
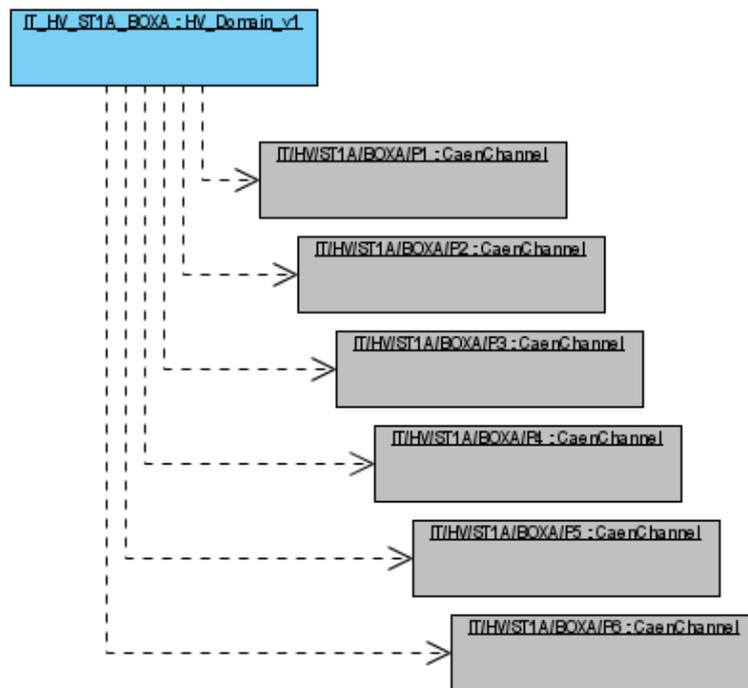
Figure 10.28: IT HV box FSM.

Despite these "obstacles" a logical hierarchical grouping similar to the IT was designed, but with an "imperfect" matching regarding to the control link and power supplies. The subdivision is done into regions: "A" (access), "B" (beam pipe) and "C" (cryo). Each region is sub-split into the "TOP" and "BOTTOM" location of the readout sectors. Finally, the service boxes layer and the partitioning of the hybrids/DBs: detector ▶ regions ▶ location (top, bottom) ▶ service boxes ▶ partitions.

The "imperfect" matching refers to the SPECS link and the power control granularity. Each group of SBs (stack in the TT and half-station in the IT) shares the same SPECS link, and all of the CBs need to be powered in order to keep the link up. In the IT the hierarchical sub-division into half-stations matches perfectly with the SPECS link deployment so single half-stations can be powered on/off. That is not the case in the TT. There are four links in the TT detector, one per detector quadrant, which does not match with the subdivision into regions. This means that there is no way to power on/off single regions without disrupting the SPECS communication of the contiguous region. As a consequence the ECS power on/off the whole detector altogether, although finer control granularity has been implemented for the FSMs which will perform automatic safety actions.

The distribution of the control system has the same shape as in the IT (Figure 10.29).

In order to understand the grouping and being able to build the hierarchy, the following pictures with the read-out sector numbering (Figure 10.30), partitioning of service boxes (Figure 10.31) and the ECS partition numbering (Figure 10.32) are needed[1]. The pictures show the layout of the detector, the regions division and the corresponding layers (*x1*, *u* , *v*, *x2*).
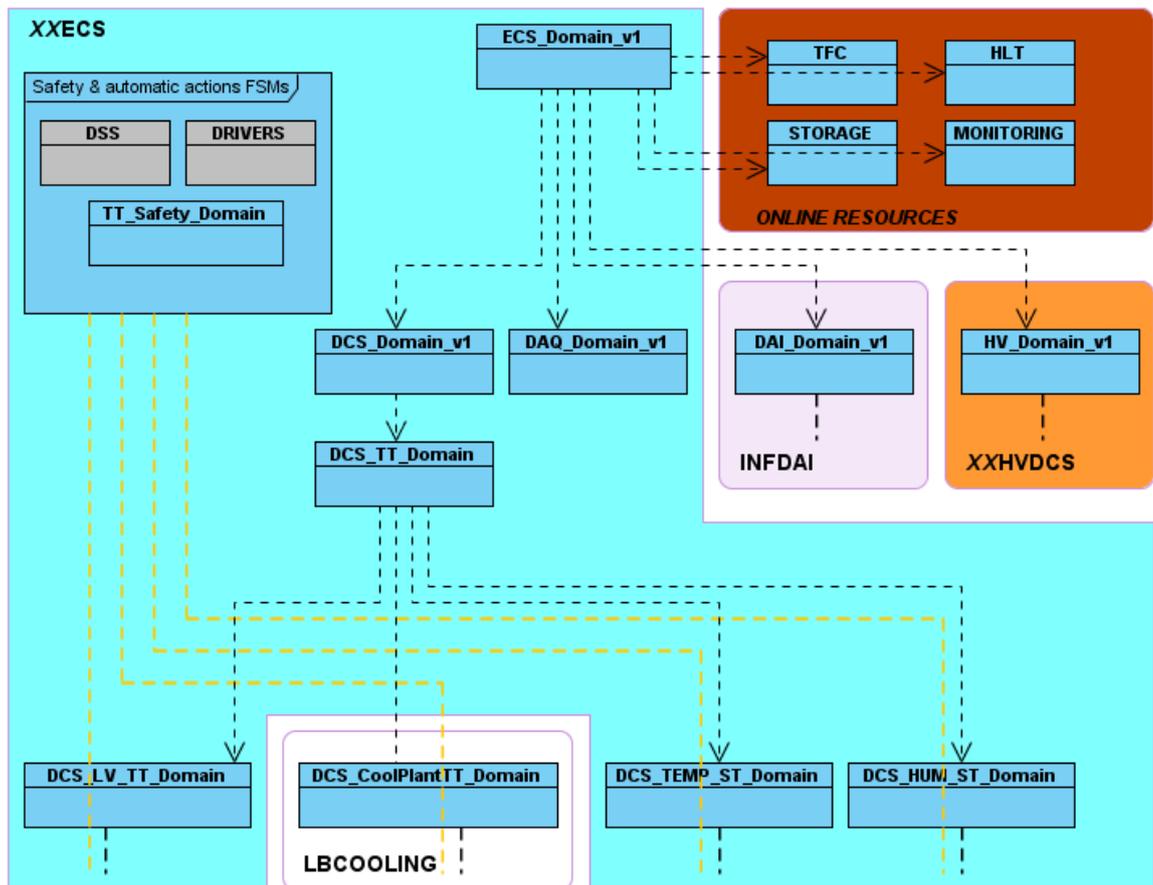
Figure 10.29: Deployment of the TT top nodes.

---

1 The authorship of these schemes corresponds to Olaf Steinkamp, U. Zurich.
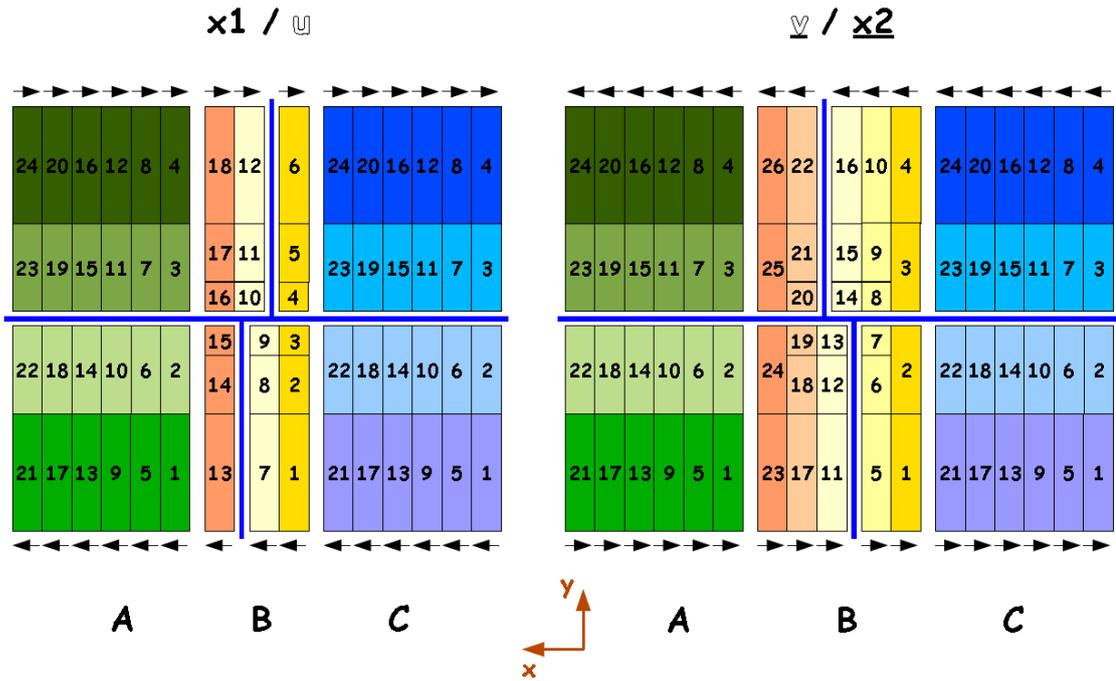
Figure 10.30: TT read-out sector numbering.
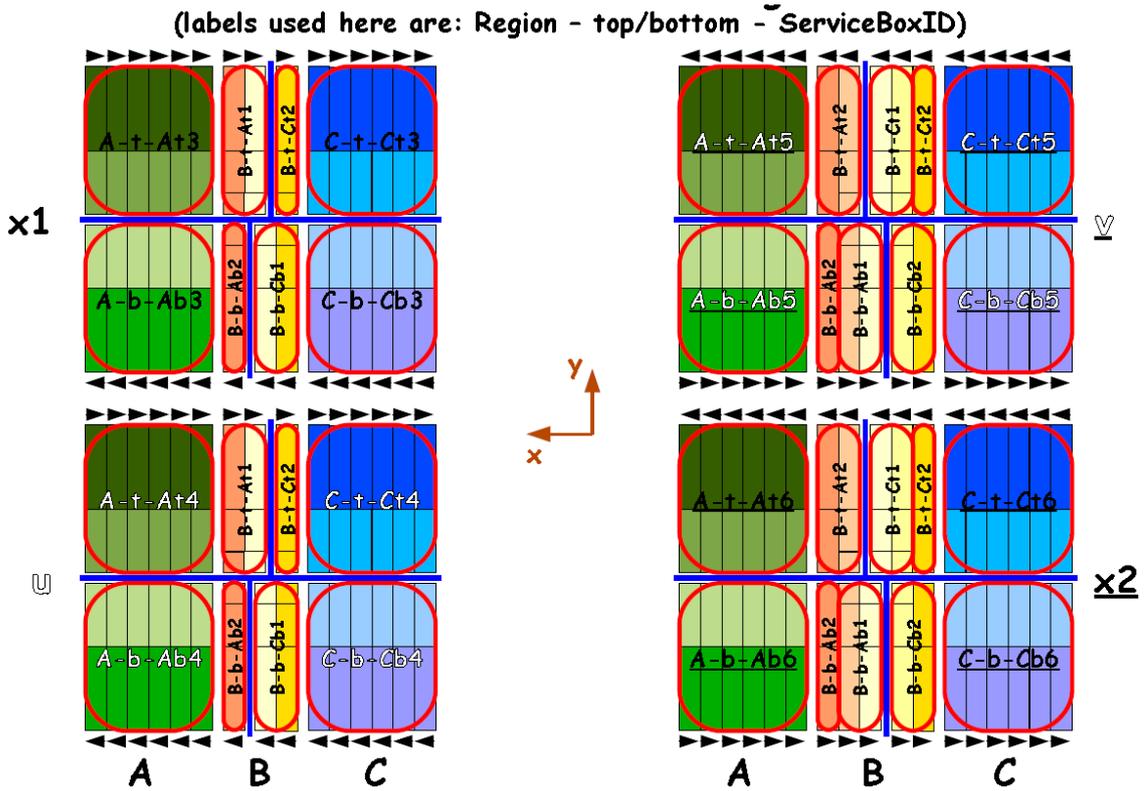
Figure 10.31: TT service box partitioning.

10.5.1 TT DAQ DOMAIN DEPLOYMENT

The deployment scheme is the same as in the IT. The only difference is the meaning of the layers: the top and regions DAQ nodes are deployed in the *XXECS* system (Figure 10.33). The system is down split into TELL1 and FE domains for each of the half-stations. The TELL1 domain is deployed in the *XXDAQTELL* system and consists of the CU that holds the TELL1 DUs. The FE domain, which represents a FE box, is sub-divided into service boxes which are in turn made up of a *ControlBoard* DU and the FE partitions. Each of the partitions is composed by DBs (of type *DigBoard*) and Hybrids (of type *Ladder*).
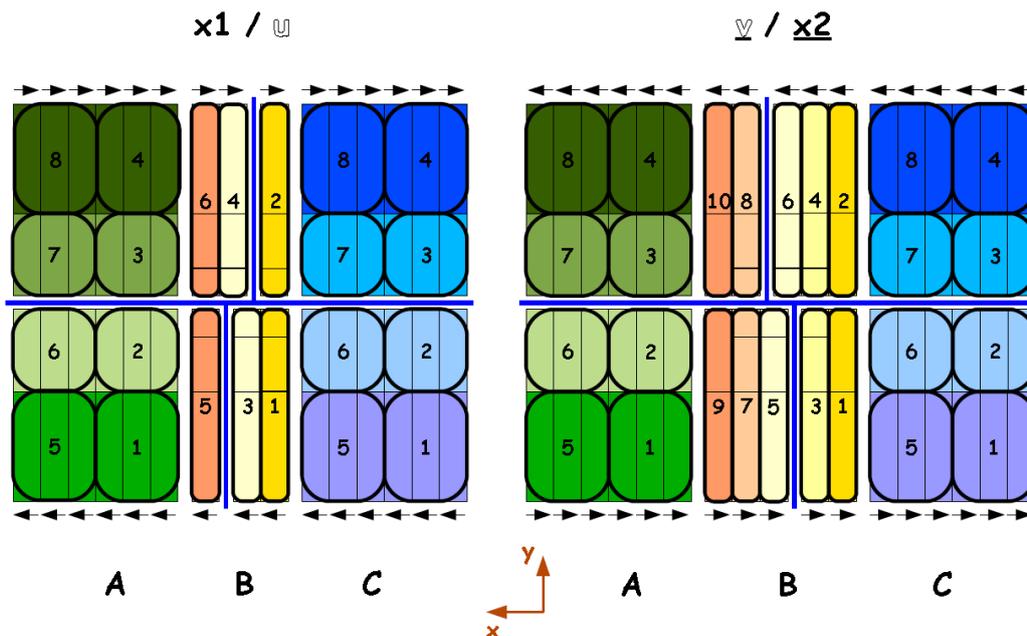
Figure 10.32: TT ECS partition numbering.
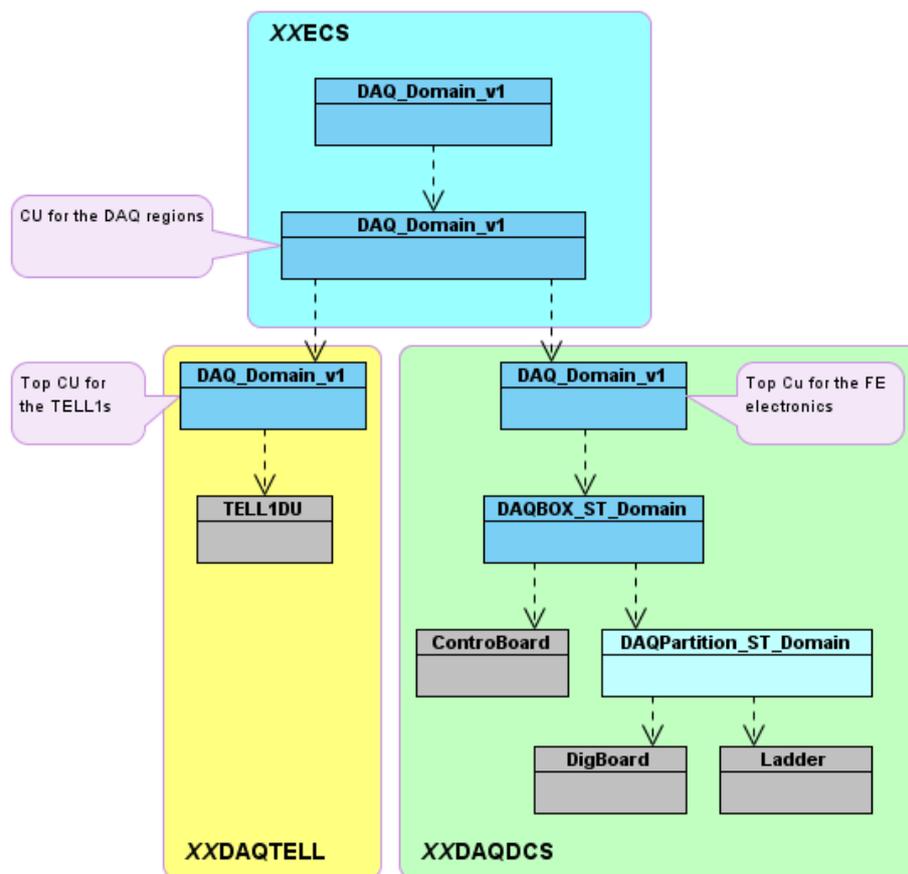
Figure 10.33: TT DAQ deployment scheme.

10.5.1.1 THE DAQ TOP DOMAIN

The DAQ domain is first divided into three regions with the following naming:

▶ *"TT_DAQ_$i": $i identifies the region: "A", access part; "B", beam pipe region; "C", cryo part (see Figure 10.34).*
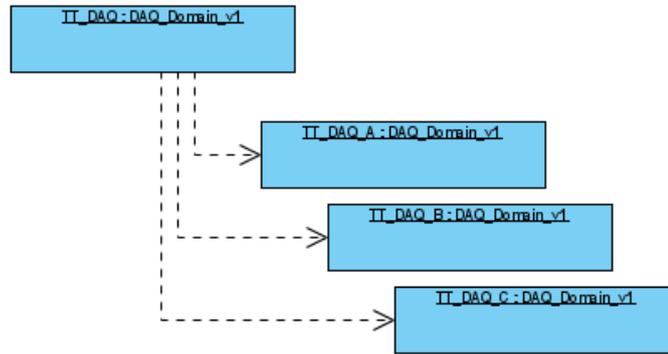
Figure 10.34: IT_DAQ domain split up into three regions.

10.5.1.2 THE REGIONS

Each of the regions is sub-divided in the two corresponding boxes and a TELL1 control domain (Figure 10.35). The naming convention rule for this layer consists of adding a suffix to the parent CU name:

▶ *"_TELL1": for the Tell1s CU domain.*

▶ *"_FE$i": where the values of $i can either be "T" (Top) or "B" (Bottom). This division indicates whether the service boxes are placed in the top or bottom quadrant.*
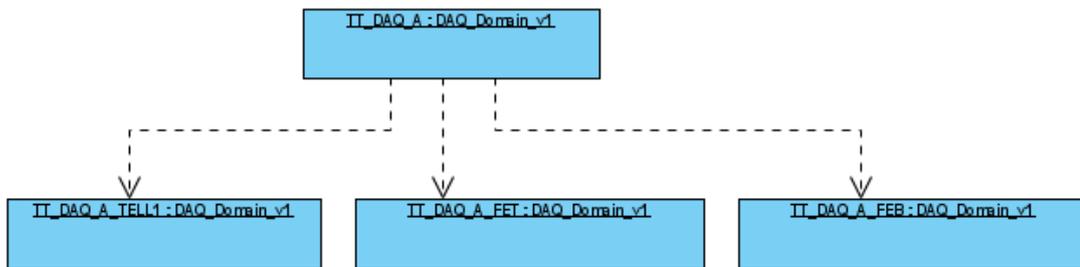
Figure 10.35: Half-station DAQ subdivision.

10.5.1.3 THE TELL1S

16 TELL1 boards are needed per TT region (Figure 10.36). The TELL1 names correspond to their host names:

▶ *"TTTELL$i": $i is the TELL1 index that matches with the last word of the TELL1 IP address.*

The correspondence between half-stations and TELL1 indexes is defined in the table:

"Region A"	33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48
"Region A"	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16
"Region C"	17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32

Table 10.2: TELL1 to half-stations correspondence.

10.5.1.4 THE FE ZONES

Each top-bottom zone is read out by 4 service boxes (Figure 10.37). The naming convention for the SBs consists of adding:

▶ *"_ij": $i indicates the SB tower quadrant:"AT" (access top), "AB" (access bottom), "CT" (cryo top) and "CB" (cryo bottom). $j the service box index in the tower. It ranges from 3 to 6 in the "A" and "C" regions and from 1 to 2 and in the "B.*

Figure 10.36: TELL1 CU for the A region.

Figure 10.37: FE DAQ TT zone split into 4 service boxes.

10.5.1.5 THE FE SERVICE BOXES

The SBs house 12 DBs and hence link to 12 hybrids (Figure 10.38). As an exception 4 SBs only have 11 DBs enabled as they are not needed.

The naming for the FE partitions (which are implemented as LUs) consists of appending the following to the service box name:

▶ *"_Pij": $i represents the ECS partition as numbered in the Figure 10.32 and $j the layer where the group of hybrids belong to.*

The naming convention for the Control Board DUs is:

▶ *"TT_REG$i_$j_kl_CB" for the Control Board DUs:*

- *$i: the region ID: "A" (access), "B" (beam pipe), "C" (cryo).*

- *$j represents the service boxes tower location, "TOP" or "BOT"(bottom).*

- *$k indicates the SBs quadrant:"AT" (access top), "AB" (access bottom), "CT" (cryo top) and "CB" (cryo bottom).*

- *$l: the SB index in the tower. It ranges from 3 to 6 in the "A" and "C" regions and from 1 to 2 and in the "B".*
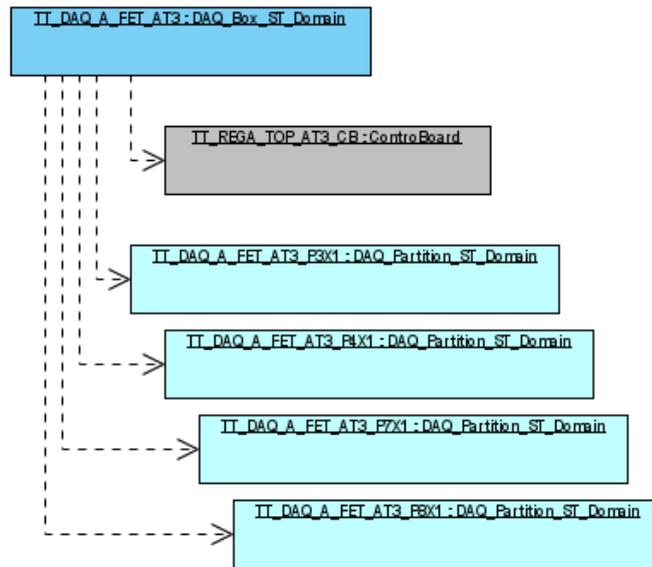
Figure 10.38: FE box sub-division into one CB and various partitions.

10.5.1.6 THE FE PARTITIONS

Each FE partition consists of 3 DBs and 3 hybrids (Figure 10.39).

The naming convention for the Digitizer Boards is the following:

▶ *"TT_REG$i_$j_kl_DB$m" for the Dig Board DUs:*

- *$i represents the region "A" (access), "B" (beam pipe) or "C" (cryo).*
- *$j represents the service boxes tower location, "TOP" or "BOT"(bottom).*
- *$k indicates the SBs quadrant:"AT" (access top), "AB" (access bottom), "CT" (cryo top) and "CB" (cryo bottom).*
- *$l: the SB index in the tower. It ranges from 3 to 6 in the "A" and "C" regions and from 1 to 2 and in the "B".*
- *$m is the slot in the service box where the DB is plugged in.*

For the hybrids:

▶ *"TT_REG$i_$j_Lk_Sl" for the hybrids DUs:*

- *$i represents the region "A" (access), "B" (beam pipe) or "C" (cryo).*
- *$j represents the service boxes tower location, "TOP" or "BOT"(bottom).*
- *$k is the layer where the hybrid belongs to: "X1", "U", "V", X2".*
- *$l the readout sector number as in the figure.*
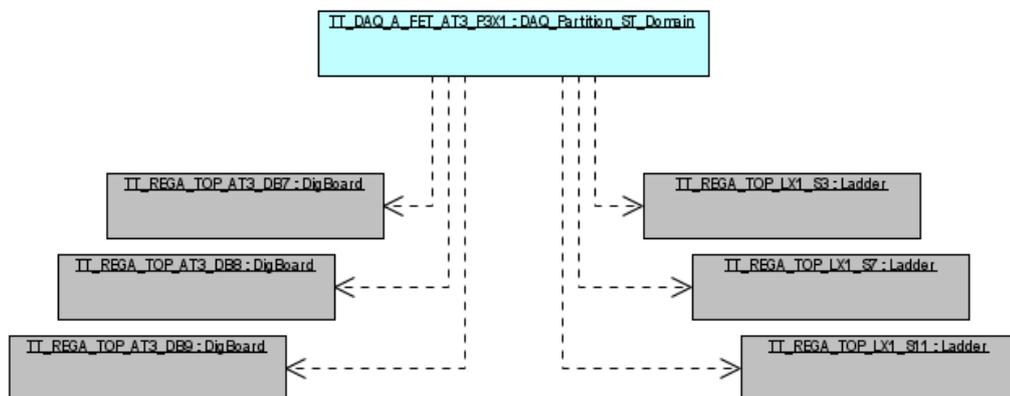
Figure 10.39: TT partition with 3 DBs and 3 hybrids.

10.5.2 THE DCS DOMAIN

As for the IT the DCS domain is subdivided into DCS_LV, DCS_TEMP, DCS_HUM and DCS_COOLING. Between the top DCS domain and the four sub-domains there is an intermediate one needed, because the DCS domain is a standard CU type that is common for all LHCb detectors and its behavior must not be modified. The intermediate CU implements the sequencing of actions: before the temperature and humidity monitoring is started, the LV must be on. The standard DCS domain does not perform a sequential switch on (first LV and then the rest) as the intermediate does.

10.5.3 DCS LV DOMAIN DEPLOYMENT

The DCS_LV domain is part of the DCS. It is deployed into three systems: *XXECS*, *XXHVDCS* and *XXDAQDCS* (see Figure 10.40). The first of the systems hosts the top level CUs, i.e. the top TT_DCS_LV together with the sub-layer of regions. The MARATON power supplies are directly linked from the TT_DCS_LV node are deployed in the *XXHVDCS* system whereas the regulators control is in *XXDAQDCS*.

The MARATON CU controls all the channels for the TT. This CU is split first into quadrants and then into SBs with the two corresponding MARATON channels.

The top CU deployed in *XXDAQDCS* represents the top-bottom zone where the SBs belong to. The zone is sub-divided into 4 SBs that control and monitor both the CB and partitions power regulators.

10.5.3.1 THE DCS LV DOMAIN

It is divided into regions and a MARATON domain (see Figure 10.41). The MARATONs need to be at this layer as the SPECS links do not match with the regions and hence the regions cannot be independently powered off.

The names of the MARATON append the following to the parent name:

▶ *"_MRT" for the MARATON CU domain.*

The names for the regions append:

▶ *"_$i" where $i represents the region "A" (access), "B" (beam pipe) or "C" (cryo).*
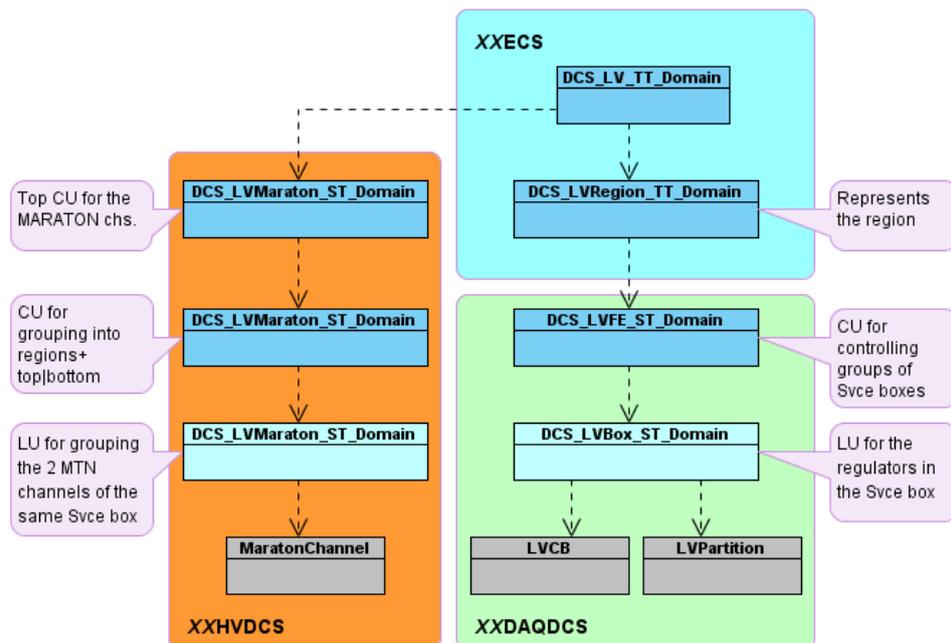
Figure 10.40: DCS low voltage hierarchy deployment.

10.5.3.2 THE DCS REGION LV DOMAIN

Each region is subdivided into the top and bottom zones and each zone in four services boxes as in the Figure 10.42. The name of the CUs is obtained by adding the following to the parent CU:

► *"_$j" where $j can either be "TOP" or "BOTTOM".*

Below the top and bottom zones hang the service boxes nodes (Figure 10.43). To the name of the previously described nodes one has to add the *ID* of the service boxes:

► *"_ij$" where $i indicates the SB tower quadrant:"AT" (access top), "AB" (access bottom), "CT" (cryo top) and "CB" (cryo bottom) and $j the service box index in the tower. It ranges from 3 to 6 in the "A" and "C" regions and from 1 to 2 and in the "B".*
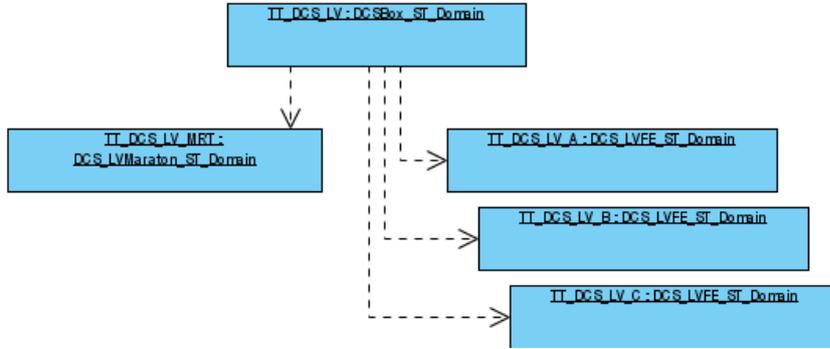
Figure 10.41: TT_DCS_LV domain.
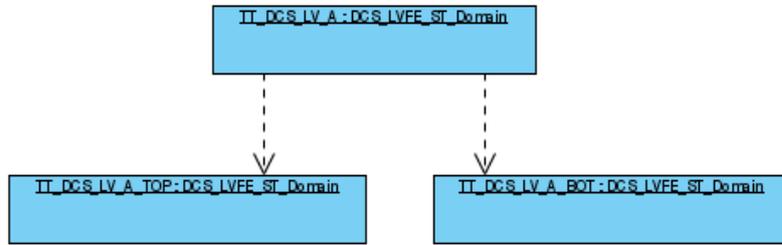
Figure 10.42: Subdivision of the region low voltage domain.
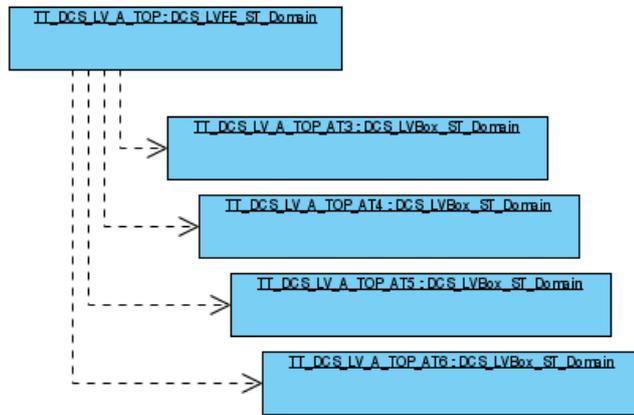
Figure 10.43: Subdivision of the top-bottom zones of the low voltage domain.

10.5.3.2.1 THE TT DCS SERVICE BOX LV DOMAIN

Each of the service boxes houses twelve DBs/hybrids, that is, four partitions (Figure 10.44). In addition there is a DU per SB of type 'LVCB' that carries out the tasks of initializing some parameters in the CB and monitor the CB regulators.

The DUs naming is:

► *"TT_REG$i_$j_kl$_LVCB" for the Control Board LV DUs:*

   ▪ *$i: the region ID: "A" (access), "B" (beam pipe), "C" (cryo).*

   ▪ *$j represents the service boxes tower location, "TOP" or "BOT"(bottom).*

- $k indicates the SBs quadrant:"AT" (access top), "AB" (access bottom), "CT" (cryo top) and "CB" (cryo bottom).

- $l: the SB index in the tower. It ranges from 3 to 6 in the "A" and "C" regions and from 1 to 2 and in the "B".

▶ *"TT_REG$i_$j_kl$_LVP$m$n" for the LV partitions:*

- $i: the region ID: "A" (access), "B" (beam pipe), "C" (cryo).

- $j represents the service boxes tower location, "TOP" or "BOT"(bottom).

- $k indicates the SBs quadrant:"AT" (access top), "AB" (access bottom), "CT" (cryo top) and "CB" (cryo bottom).

- $l: the SB index in the tower. It ranges from 3 to 6 in the "A" and "C" regions and from 1 to 2 and in the "B".

- $m represents the ECS partition as numbered in the figure.

- $n":the layer where the group of hybrids belong to.
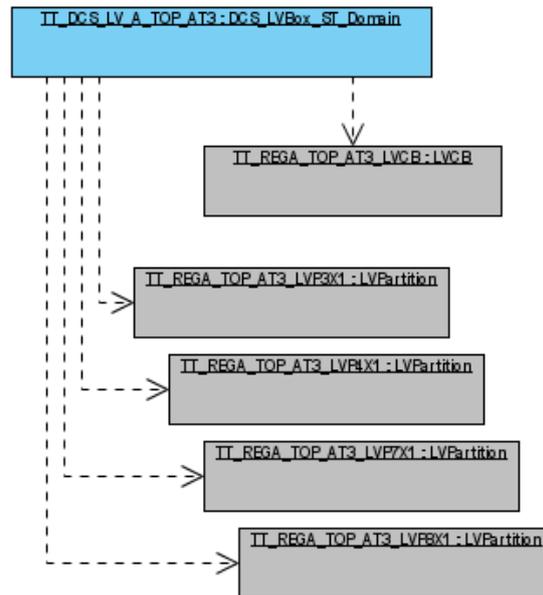
Figure 10.44: TT service box low voltage hierarchy.

10.5.3.3 THE DCS MARATON DOMAIN

This domain controls all the MARATON channels of the detector. It is divided into six domains corresponding to the regions and the top/bottom zones (Figure 10.45).

The naming of the CUs is built from the parent name:

▶ *"_$i_$j". $i represents the region ("A", "B", "C") and $j the zone ("TOP" or "BOT").*

10.5.3.4 THE DCS MARATON ZONE

To each region-zone corresponds four SBs. The LV channels are grouped in pairs into service boxes. The service boxes nodes are implemented as LUs (Figure 10.46).

The naming of the LUs is built from the parent name removing the suffix "A_TOP" (for example) and adding:

▶ *"_ij" where $i corresponds to the service boxes towers quadrant ("AT", "AB", "CT" and "CB") and $j to the service box index in the tower. $j ranges from 3 to 6 in the "A" and "C" regions and from 1 to 2 and in the "B".*

The DU name for the MARATON low voltage channels is as follows:

▶ *"TT/LV/$i/$j/km/Ch$n" for the MARATON channels.*

- $i represents the region ID: "A" (access), "B" (beam pipe) and "C" (Cryo).

- $j represents the Box ID: A (access), B (bottom), C (cryo), T (top).

- ▪ *$k indicates the SB tower quadrant:"AT" (access top), "AB" (access bottom), "CT" (cryo top) and "CB" (cryo bottom).*

- ▪ *$m: the service box index in the tower. It ranges from 3 to 6 in the "A" and "C" regions and from 1 to 2 and in the "B".*

- ▪ *$n can either be 1 (2.5V channel) or 2 (5V channel).*
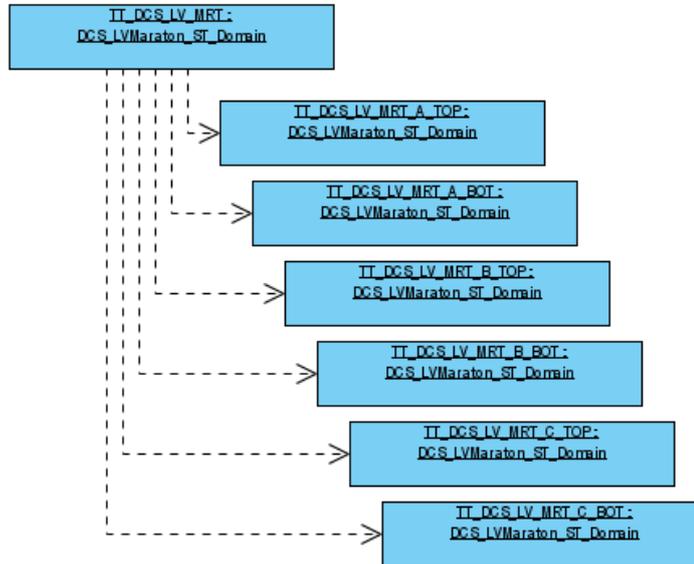
Figure 10.45: TT MARATON domain.
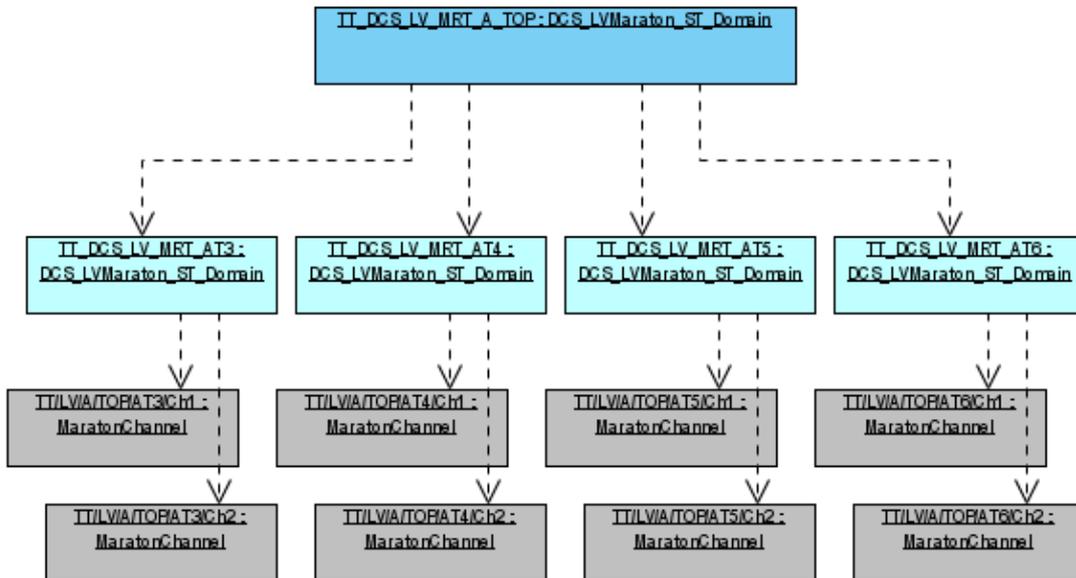
Figure 10.46: MARATON zone sub-hierarchy.

10.5.4 DCS TEMPERATURE DOMAIN DEPLOYMENT

It follows a structure similar to the low voltage with the exception that there is not a MARATON domain (Figure 10.47). It differs from the IT in the point that the temperatures of the box hang from the SB level.

10.5.4.1 THE DCS TEMPERATURE TOP DOMAIN

As the others the first sub-division is into regions (Figure 10.48). The naming is similar to the the LV domain.

- ▶ *"TT_DCS_TEMP_$i": $i identifies the region: "A", "B" or "C".*

10.5.4.2 THE DCS TEMPERATURE REGION AND ZONE DOMAIN

The region is divided into the top and bottom zones and the zones into service boxes (Figure 10.49). The names of the zones and service boxes append a suffix to the corresponding parent name.

Zones:

▶ *"_$i": can either be "TOP" or "BOTTOM".*

Service boxes:

▶ *"_ij$" where $i indicates the SB tower quadrant:"AT" (access top), "AB" (access bottom), "CT" (cryo top) and "CB" (cryo bottom) and $j the service box index in the tower. It ranges from 3 to 6 in the "A" and "C" regions and from 1 to 2 and in the "B".*
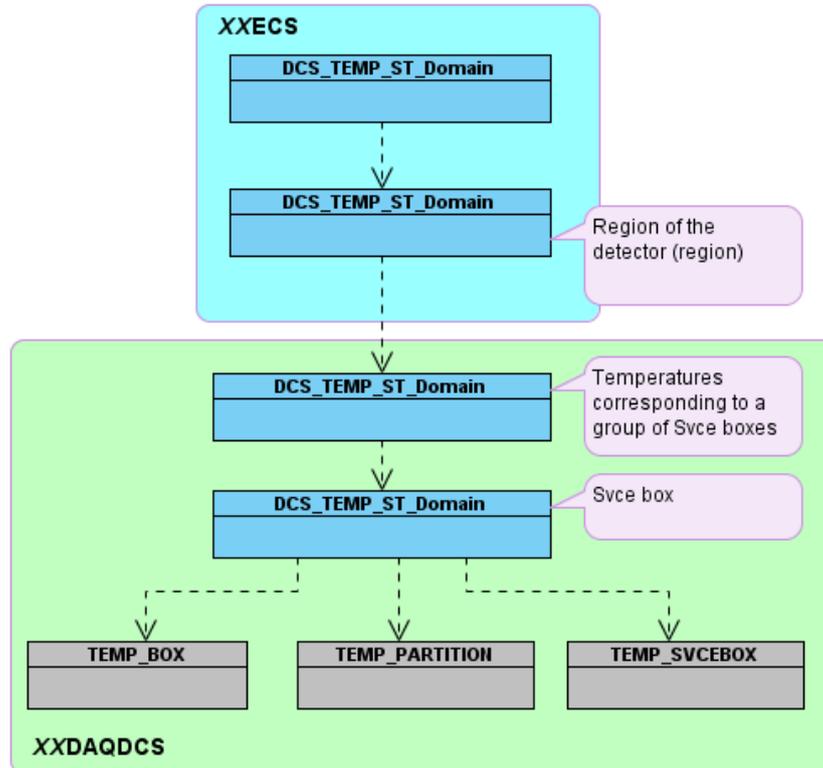
Figure 10.47: DCS temperature domain deployment.
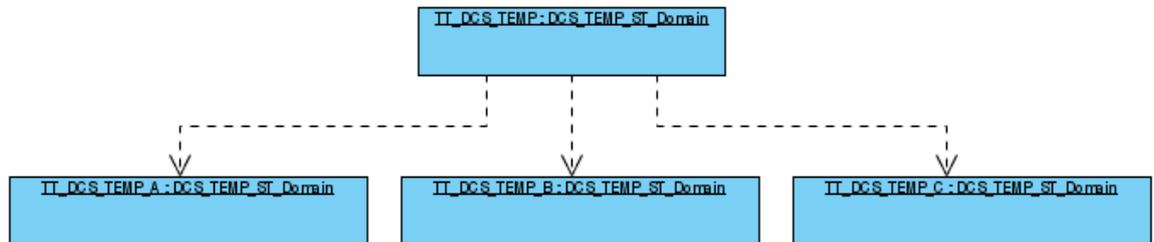
Figure 10.48: DCS temperature domain split into regions.
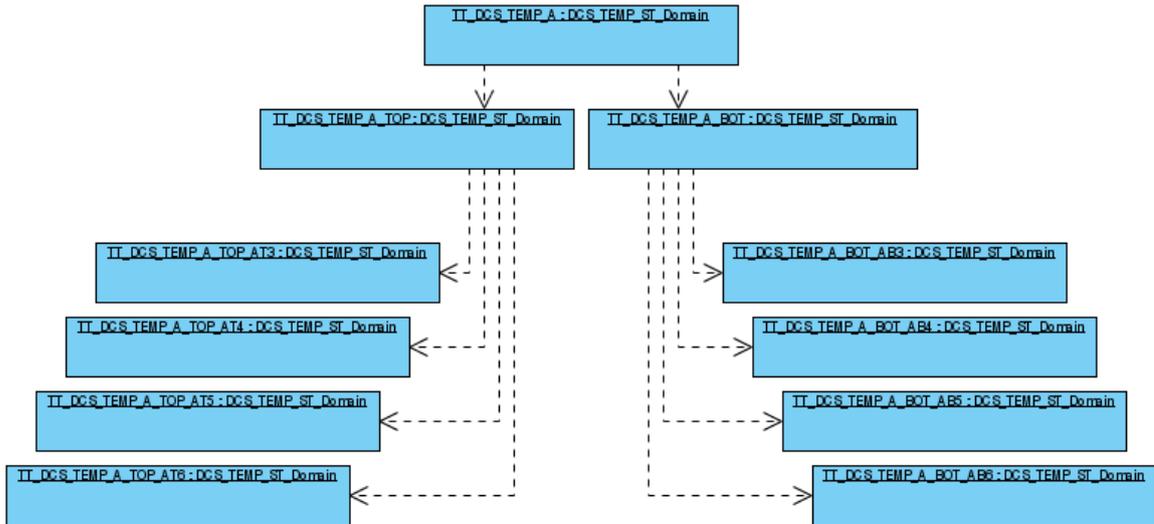
Figure 10.49: Subdivision of the region down to zones and service boxes.

10.5.4.3 THE DCS TEMPERATURE SERVICE BOX DOMAIN

The service box CU for the temperatures comprises a DU that corresponds to the two monitored temperatures in the SB backplane and one on the CB, the 4 DUs representing four hybrid temperatures of a partition and in some cases a DU for some temperature readings of PT1000 placed inside the detector box (Figure 10.50).

DU naming:

▶ *"TT_REG$i_$j_kl$_TEMP" for the Control Board LV DUs:*

  ▪ *$i: the region ID: "A" (access), "B" (beam pipe), "C" (cryo).*

  ▪ *$j represents the service boxes tower location, "TOP" or "BOT"(bottom).*

  ▪ *$k indicates the SBs quadrant:"AT" (access top), "AB" (access bottom), "CT" (cryo top) and "CB" (cryo bottom).*

  ▪ *$l: the SB index in the tower. It ranges from 3 to 6 in the "A" and "C" regions and from 1 to 2 and in the "B".*

▶ *"TT_REG$i_$j_kl$_P$m$n_TEMP" for the LV partitions:*

  ▪ *$i: the region ID: "A" (access), "B" (beam pipe), "C" (cryo).*

  ▪ *$j represents the service boxes tower location, "TOP" or "BOT"(bottom).*

  ▪ *$k indicates the SBs quadrant:"AT" (access top), "AB" (access bottom), "CT" (cryo top) and "CB" (cryo bottom).*

  ▪ *$l: the SB index in the tower. It ranges from 3 to 6 in the "A" and "C" regions and from 1 to 2 and in the "B".*

  ▪ *$m represents the ECS partition as numbered in the figure.*

  ▪ *$n": the layer where the group of hybrids belong to.*

▶ *"TT_REG$i_$j_$kTEMP" for the box temperatures:*

  ▪ *$i: the region ID: "A" (access), "B" (beam pipe), "C" (cryo).*

  ▪ *$j represents the service boxes tower location, "TOP" or "BOT"(bottom).*

  ▪ *$k indicates the temperature sensor IDs. There aren't any sensors in the B region.*

Sensor IDs:	
"Access top"	A23-C3; A20-A16-A12-C4
"Access bottom"	A19-A15-A11-C2; C1
"Cryo top"	A24-C7; A22-A18-A14-C8
"Cryo bottom"	A21-A17-A13-C6; C5

10.5.5 DCS HUMIDITY DOMAIN DEPLOYMENT

The DCS humidity domain is much simpler than the previous (Figure 10.51). The eight humidity sensors are encapsulated in one CU, the DCS_HUM domain (Figure 10.52).

The naming of the DUs is:

▶ *"TT_ij_CkHk_HUM":*

- ▪ *$i indicates the SB quadrant:"AT" (access top), "AB" (access bottom), "CT" (cryo top) and "CB" (cryo bottom).*
- ▪ *$j: the SB index in the tower. It ranges from 3 to 6 in the "A" and "C" regions and from 1 to 2 and in the "B".*
- ▪ *$k is the index of the humidity and associated temperature sensor, which ranges from 1 to 8.*

10.5.6 DCS COOLING DOMAIN DEPLOYMENT

As for the IT, the cooling system control is running in an external computer. A specific project on that computer carries out the control of all the LHCb cooling plants. For each of the systems there is a CU to perform the cooling plant control tasks. Underneath the control CU there is a cooling plant DU and several temperature loops, one per each cooling loop with associated temperature monitoring.
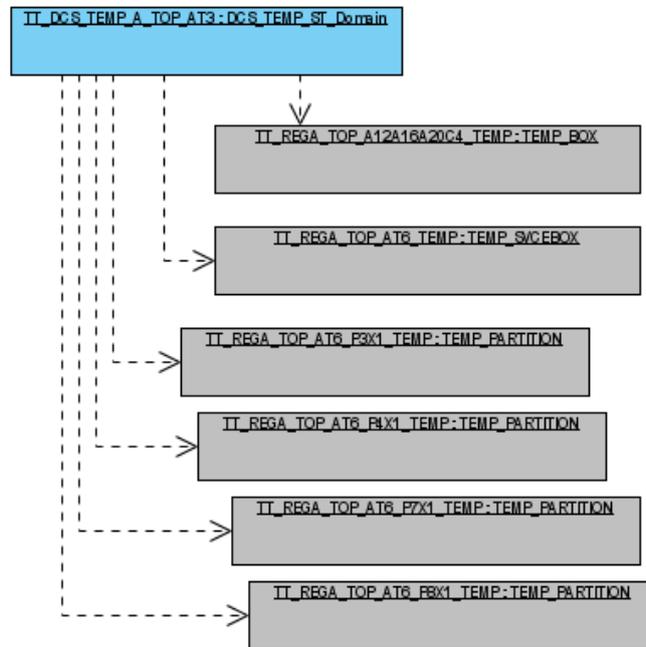
Figure 10.50: TT DCS service box temperatures CU.
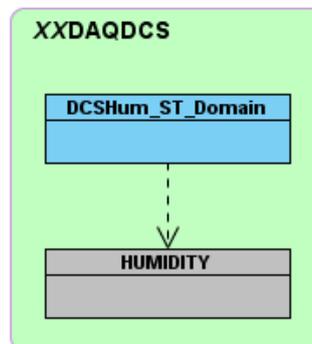
Figure 10.51: DCS humidity deployment.
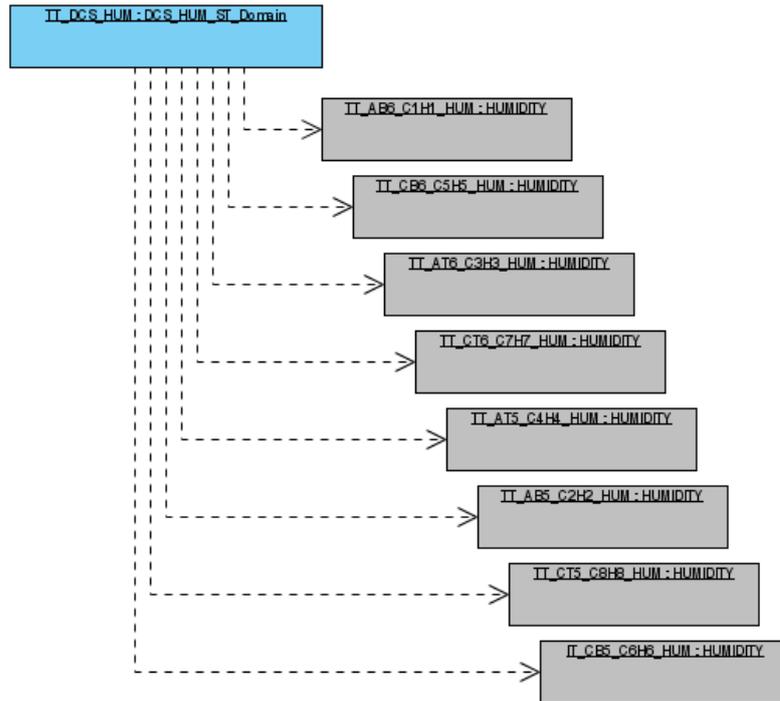
Figure 10.52: DCS humidity domain FSM.
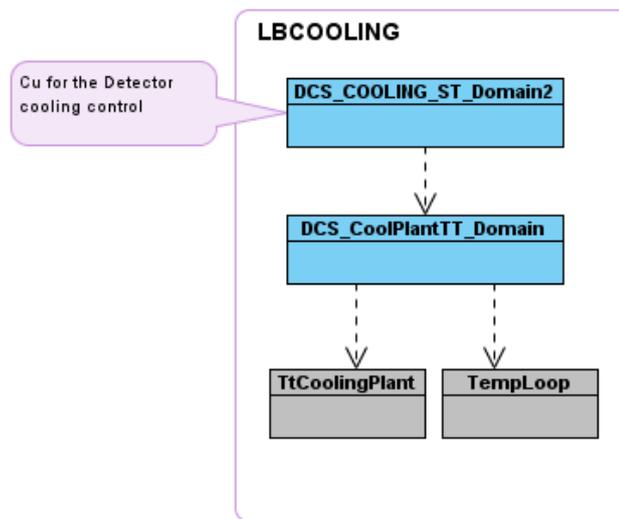
Figure 10.53: DCS cooling deployment.

10.5.7 HV DOMAIN DEPLOYMENT

The HV control domain is made of four layers (Figure 10.54): the top *TT_HV*, the regions level, the top/bottom zones, the service boxes and finally the CAEN HV channels. The SB level was skipped because the HV distribution is completely disjointed from the FE electronics grouping. The whole HV FSM hierarchy is running in the *XXHVDCS* system.

10.5.7.1 THE HV TOP DOMAIN

The same philosophy is kept (Figure 10.55), the first subdivision is into regions and the used naming is:

▶ *"TT_HV_$i": $i identifies the region: "A", "B" or "C".*
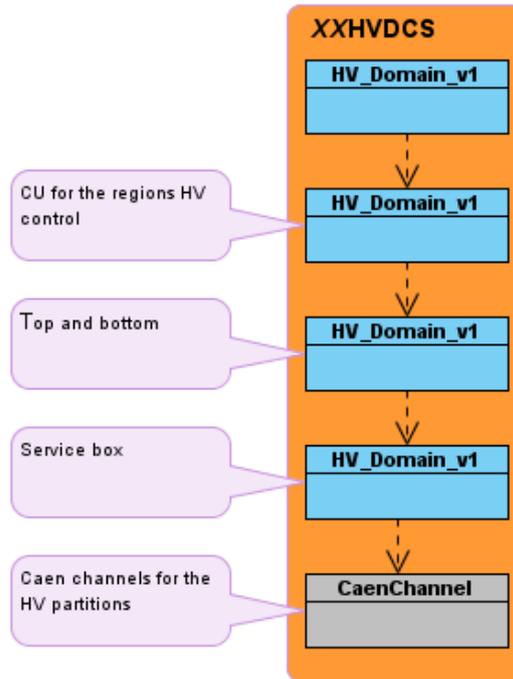
Figure 10.54: TT HV domain hierarchy layers.
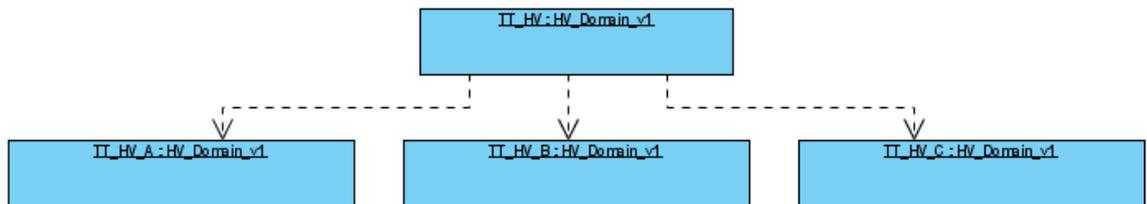
Figure 10.55: HV top domain FSM.

10.5.7.2 THE HV REGION DOMAIN

Each region is subdivided into the top and bottom zones and each zone (Figure 10.56). The name of the CUs is obtained by adding the following to the parent CU:

▶ *"_$j" where $j can either be "TOP" or "BOTTOM".*
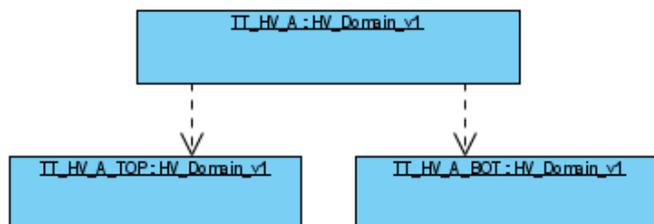
Figure 10.56: The HV region domain FSM.

10.5.7.3 THE HV TOP/BOTTOM ZONE DOMAIN

Below the top and bottom zones hang the service boxes nodes (Figure 10.57). To the name of the previously described nodes one has to add the id of the service boxes:

▶ *"_ij$" where $i indicates the SB tower quadrant:"AT" (access top), "AB" (access bottom), "CT" (cryo top) and "CB" (cryo bottom) and $j the service box index in the tower. It ranges from 3 to 6 in the "A" and "C" regions and from 1 to 2 and in the "B".*
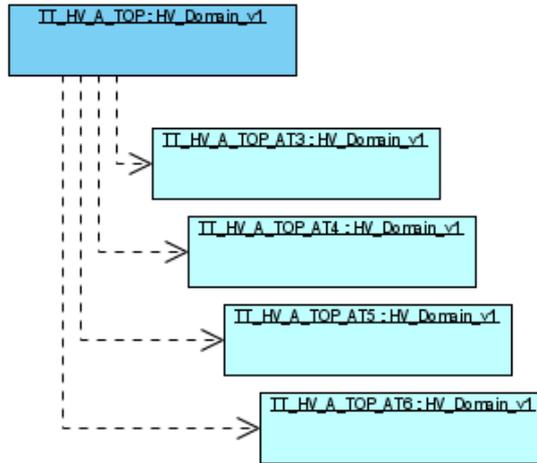
Figure 10.57: The HV top/bottom zone domain.

10.5.7.4 THE HV SERVICE BOX DOMAIN

The HV domain at the Top/Bottom level consists of a variable number of HV channels. In the "A" and "C" regions each HV channel powers three readout sectors (a partition) while in the "B" region the ratio is one to one.

The DU names are built as follows:

▶ *"TT/HV/$i/$j/kl/HVPmn[s$r]" for the CAEN channels.*

■ *$i: the region ID: "A" (access), "B" (beam pipe), "C" (cryo).*

■ *$j represents the service boxes tower location, "TOP" or "BOT"(bottom).*

■ *$k indicates the SB tower quadrant:"AT" (access top), "AB" (access bottom), "CT" (cryo top) and "CB" (cryo bottom).*

■ *$l: the service box index in the tower. It ranges from 3 to 6 in the "A" and "C" regions and from 1 to 2 and in the "B".*
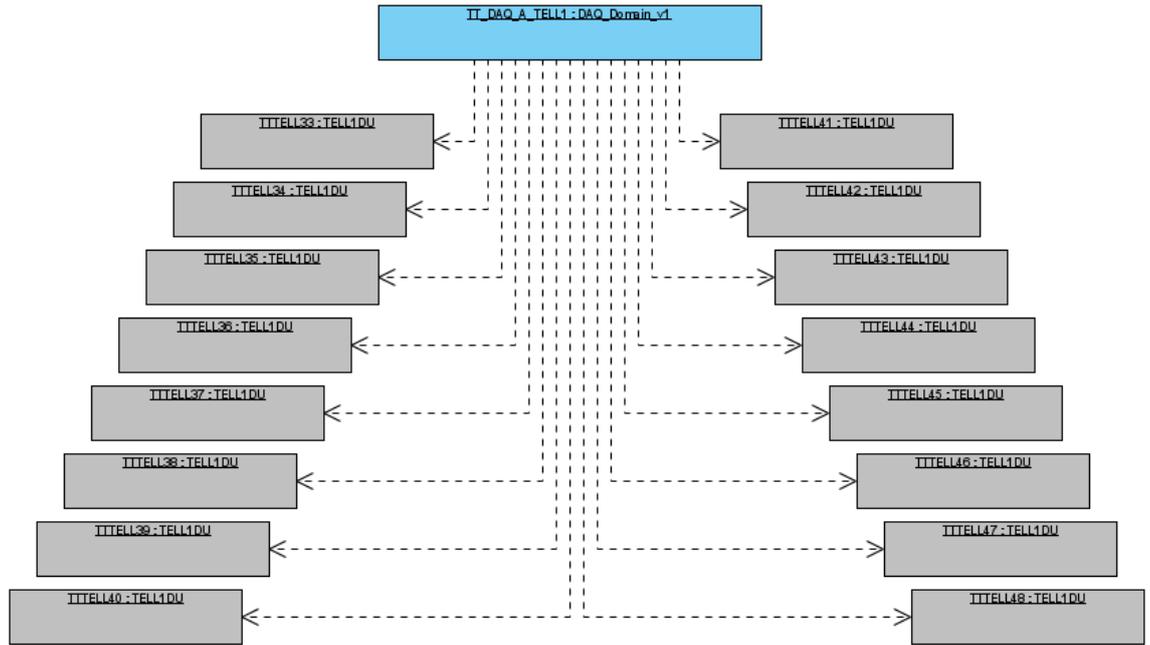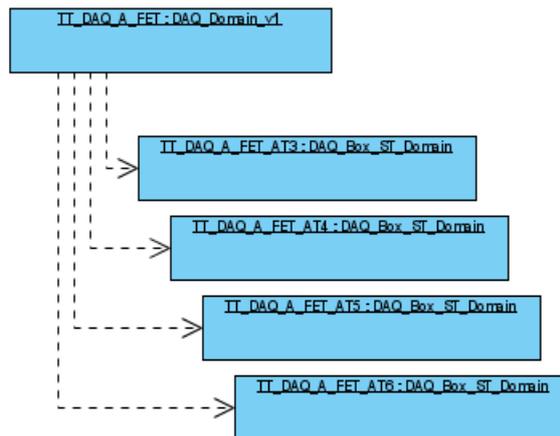
■ *$m represents the ECS partition as numbered in the Figure 10.32.*

■ *$n: the layer where the group of hybrids belong to.*

■ *[s$r]: it is just for the "B" region where each read-out sector is biased by a single HV channel. $r is the read-out sector index as in the Figure 10.30.*
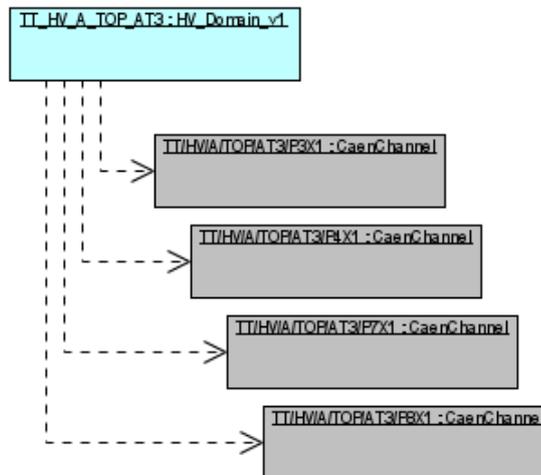
Figure 10.58: TT HV service box LU.

10.6 DETECTOR CONFIGURATION

The detector configuration settings are structured in the so-called recipes. A recipe is associated to a specific activity of the detector together with the command that is to be executed. For instance, if the detector needs to be configure for cosmic data taking and the activity is called *COSMICS*, for the *Configure* command (which is the one for configuring the DAQ devices) and recipe called *COSMICS/Configure* must exist. The recipe includes all settings for all the devices which need to be configured.

In the following table we describe the commands and corresponding domain which apply a recipe.

HV	DCS	DAQ	TFC
GoStandBy1	SwitchOn	Configure	Configure
GoStandBy2	SwitchHybridsOff		
GoReady			

The next tables describe briefly the contents of the recipe for each of the pairs domain/command and the corresponding DU type.

HV	
	CaenChannel
GoStandBy1	Set the channel to 5V and the current limits
GoStandBy2	Set the channel to a voltage in which the sensors are still not depleted and also the current limits
GoReady	Set the channel to a nominal voltage where the sensors are biased and the current limits

DCS_LV			
	MaratonChannel	LVCB	LV_PARTITION
SwitchOn	Set the LV current limits	Defines the CtrlBoard voltages which are being monitored	Defines which hybrids are powered on and monitored.
SwitchHybridsOff	Set the LV current limits	Defines the CtrlBoard voltages which are being monitored	NOP (the hybrids will be powered off)

DCS_TEMP			
	TEMP_BOX	TEMP_SVCEBOX	TEMP_PARTITION
SwitchOn	Defines which of the 4 temperature sensors are enabled	Defines which of the 3 temperature sensors are enabled	Defines which of the partition temperature sensors are enabled
SwitchHybridsOff	Defines which of the 4 temperature sensors are enabled	Defines which of the 3 temperature sensors are enabled	Defines which of the partition temperature sensors are enabled

DAQ				
	ControlBoard	DigitizerBoard	Ladder	TELL1
Configure	Defines the timings for the Ctrl Board	Defines the settings of the GOLs	Defines the Beetle settings	Defines all the TELL1 parameters

TFC	
	Odin
Configure	Defines the Odin run parameters

A recipe for the activity *DEFAULT* must exist for each DU type. Whenever the specific recipe for the corresponding activity does not exist this one will be loaded. In principle this recipe has to configure the electronics for normal data taking.

The recipe naming grants the possibility to build staggered activity names. In case the specific name does not exist the next root activity name is used, otherwise the *DEFAULT* is loaded. For example, if the activity to be loaded is *PHYSICS/BEAM1*, first the *PHYSICS/BEAM1/Command* recipe is tried, if it does not exist *PHYSICS/Command* is tried and eventually *DEFAULT/Command*.

In this chapter we will only present the set of recipes exist for the ST normal operation. When running under the LHCb scope usually the *DEFAULT/Command* is used and in case new special settings are needed they are built on-demand basis.

The following table summarizes the set of ST standard activities and hence recipe names. The 'Y' symbol indicates that specific settings for that activity must exist in the corresponding DU type.

		CaenChannel	MaratonChannel	LV_PARTITION	LV CB	TEMP_BOX	TEMP_SVCE BOX	TEMP_PARTITION	COOLING	TELL1	ControlBoard	Ladder	DigitizerBaord	Odin
DEFAULT		Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ST_TEST														Y
	VFS100											Y		Y
	VFS400											Y		Y
	VFS1000											Y		Y
	VFS100_TP											Y		Y
	VFS400_TP											Y		Y
	VFS400_TPALL											Y		Y
	VFS1000_TP											Y		Y
	ORX			Y						Y		Y	Y	-
	ADC_TIMING											Y	Y	

The description of the recipe is as follows:

DEFAULT	The default hardware settings for the normal data taking
ST_TEST|VFS100	VFS set to 100mV; tpSelect mask to "0"; no testpulse from the odin
ST_TEST|VFS400	VFS set to 400mV; tpSelect mask to "0"; no testpulse from the odin
ST_TEST|VFS1000	VFS set to 1000mV; tpSelect mask to "0"; no testpulse from the odin
ST_TEST|VFS100_TP	VFS set to 100mV; tpSelect mask to "1 every 8th"; Testpulse from the odin
ST_TEST|VFS400_TP	VFS set to 400mV; tpSelect mask to "1 every 8th"; Testpulse from the odin
ST_TEST|VFS400_TPALL	VFS set to 400mV; tpSelect mask to "1 all channels"; Testpulse from the odin
ST_TEST|VFS1000_TP	VFS set to 1000mV; tpSelect mask to "1 every 8th"; Testpulse from the odin

ST_TEST|ORX	Ladders not powered; GOLs in test mode; odin "not needed"
ST_TEST|ADC_TIMING	Use Vfs=400 and testpulse. Used for performing ADC scans

10.7 PARAMETER SCANS

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

For this purpose the top FSM node, IT_ECS, implements an iterative procedure which consists of sending a special *Step* command together with the iteration index parameter *STEP_NR* to the DAQ domain for each loop iteration. In each iteration a predefined number of auto-triggers (or calibration signals) are sent by the RS to the DAQ electronics and when all the triggers have been dispatched a new iteration start with the *N_ITER* parameter incremented. The number of iterations ('steps') as well as the number and type of triggers can be defined in the top node user interface panel. Depending on what parameter we want to scan in the readout electronics, the corresponding DU that holds the parameter information must implement the *Step* command, get the *STEP_NR* parameter value and apply the parameter settings based on the *STEP_NR* value and eventually also on the activity name.

As an example, for doing testpulse scans, the *Step* command must be decoded and implemented in the *ControlBoard* DU because it consists of time delaying the *calib0* signal in the Delay25 chip. The resolution of the Delay25 chip is 0.5 ns, so the applied delay in each of the iterations can be computed as $STEP\_NR \times K$, where K is a multiplier in case we want to do delay steps coarser than 0.5 ns. The RS must be configured to send calibration triggers and the number of steps has to be set according to the value we have predefined in the code for K and the time window we want to scan.

CHAPTER 11

THE ST ECS ERROR HANDLING AND DETECTOR SAFETY

ABSTRACT

This chapter describes the automatic operation sequences that the ECS implements in order to operate the ST detector in a safe and reliable way. First, the start-up sequences to set the detector in a well known state are described. In addition the possible ST detector states based on the status of the different detector elements will be defined. The states are of crucial interest for the safe operation of the detector when the LHC beam is ready. Besides, the ECS automatic actions taken in case of abnormal status of any of the detector elements are reported. Next, the operational alarm ranges and the error recovery mechanisms are mentioned.

"Bebo para hacer interesantes a las demás personas"

Groucho Marx

11.1 AUTOMATION AND DETECTOR SAFETY

When speaking about automation we cover two different aspects of the control system process:

1. Sequencing of actions in order to set the detector in a well known running state.
2. Automatic error recovery and preventive actions.

The sequencing of actions is performed at two different levels depending on the complexity of the action to be taken and on the level of abstraction it is related to. There are high level actions such as powering on the electronics before starting the temperature monitoring, which are implemented in the CUs (Control Units); and low level commands at the DU (Device Unit) level, e.g. resetting and initializing the DCUs before starting the monitoring. The high level actions are commanded by the FSM (Finite States Machines) logic while the commands at the DU level are executed by the PVSS scripts. Automatic sequencing is basically intended for the detector start up.

Automatic reaction to abnormal situations is needed in order to avoid damaging the detector. In addition, the detector should show a coherent state based upon the status of the different elements of the system and hence avoid non allowed operations. An example of the latter would be that in case the any of the IT low voltage supplies goes to off, the DCS domain will display the correct state (*OFF*), but the DAQ CU corresponding to the control of that specific box should also be aware of the abnormal situation an update its state accordingly (go to *NOT_READY*). As the control system is organized from the top layer by domains (DCS, DAQ, HV, etc.) and there is no direct interconnection between low level nodes, a way to correlate the state of CUs corresponding to the same ensemble of equipment is needed. A parallel tree that replicates the structure of the detector hardware is the only way to take the proper logical actions.

Finally we discuss the DSS signals. They are published via DIP so they can be subscribed by the ECS. This information shall be used by the ECS to update the detector state in the case of a DSS action and also to perform any preemptive action on the hardware if needed.

11.1.1 OPERATION SEQUENCES

The overall start-up sequence of the ST detector from top to bottom is described here. The logical sequence of operations is described at different levels, starting from the top IT control CU down to the final DUs. It is assumed that all the software services (middleware, PVSS ...) are in running state and the execution of all commands or operations are error checked. If any error occurs the system goes to *ERROR* state and a meaningful message describing the problem is displayed in the alert screen.

1. IT top: the first action to be taken when running the detector stand-alone is to allocate the online resources (TFC, storage...). Next, the DAI (Data Acquisition Infrastructure) must be set in *READY* state in order to go on with the system start-up. The DAI controls and monitors the fan trays and the TELL1, HV and LV crates in the barrack. In addition it supervises the racks' environmental status. Due to a hardware feature of the HV system, for the IT the step to follow is to set the HV to *STANDBY1*. The next step is switching on the DCS (LV, temperature, humidity and cooling). Afterwards the DAQ (FE, TELL1 and Hugin boards) and the online resources (TFC, HLT, storage and monitoring) are configured and go to *READY* state. When done, the HV is set to the desired value and as the last action the DAQ and online elements are set in *RUNNING* state, so the system is definitely taking data. All these actions must be done manually but there is quite automatic sequencing behind for each of the domains. We describe next the ones specific for the ST.
2. DCS: in the case of the IT detector the first is to check that the HV is in the right state. Then the C_6F_{14} cooling system is checked to be in *READY* state otherwise the hybrids must not be switched on. The next step is to power on the low voltage and the corresponding monitoring of all the LV related signals (voltage and current values, over-current and inhibit regulators signals). Once the LV is *READY*, the temperature and humidity monitoring are also started. In section 11.1.1.1 there is a more comprehensive description.
3. DAQ: the TELL1 and the FE electronics are programmed in parallel, but in the FE side the CBs have to be configured before the rest of the elements. A more detailed explanation can be found in 11.1.1.2.

11.1.1.1 THE DCS SEQUENCE

Upon the reception of a *SwitchOn* command, the DCS domain is started up and goes to *READY* state. The initial step for the IT is to verify that the HV is in one of the following states: *STANDBY1*, *STANDBY2*, *READY*, *RAMPING_STANDBY1*, *RAMPING_STANDBY2* and *RAMPING_READY*. Otherwise the DCS will

stay in *NOT_READY*. Then the cooling plant is checked to be in *READY* state so the LV can be safely powered on. The plant will only display this state when the coolant circulates through the circuit loops at the pre-set temperature. After that the LV can be switched on followed by the temperature and humidity monitoring.

In order to start-up the LV sub-domain, first the MARATON channels are switched on. Once this is done the CBs are powered so the embedded SPECS slaves. The sequence of operations that follows is:

1. Cleanly initialize the SPECS communication system by resetting the SPECS masters.
2. Then the Control Board (CB) in the Service Box has to be initiated by resetting and initializing the SPECS, TTCrq, DCUs and Delay25 chip registers. Then the SPECS slave has to be configured to work synchronous with the LHC clock coming from the TTCrx.
3. When the CB is ready, the monitoring of the CB regulator voltages and over-current signals is also started.
4. Then the partitions of DBs and hybrids are powered up. A dead time of hundreds of milliseconds between the power up of DBs and hybrids is implemented in order to avoid power surges caused by a sudden switch on of the entire system.
5. The DCUs of the DBs are initialized and the monitoring of the Backplane regulators voltages and OCM signals started.

The DCS temperature sub-domain is started after the LV since this task is carried out by specific channels in the Control Board and Digitizer boards DCUs, which need to be powered beforehand. The function of the *SwitchOn* command in the temperature DUs is to start up the temperature monitoring services.

The humidity domain is finally initiated once the LV and temperatures are on. As for the temperature domain, the LV must be on *READY* state first, otherwise the humidity biasing and read-out electronics might not be powered up. The temperature readings are needed to calculate the relative humidity value inside the detector boxes.

A special command, *SwitchHybridsOff*, permits to set the detector in a safe state (*HYBRIDS_OFF*). In this state the detector is partially powered, the CBs and DBs are on, whereas the Hybrids are off, and temperatures and humidity are being monitored (*READY* state). This is useful in case there is no need to have the Hybrids on and therefore keep the LV power supplies consumption lower.

11.1.1.2 THE DAQ SEQUENCE

The DAQ devices configuration may take a while as it implies loading all the parameters corresponding to the defined run type (*PHYSICS*, *CALIBRATION*, etc.) The TELL1s, Hugins and the FE are configured in parallel, but at the service box (SB) level the CBs are configured first: it entails configuring the TTCrx, setting the right delays for the fast control signals and enabling them. Once the CB is *READY*, the hybrids and DBs belonging to the same service box are all configured in a concurrent way.

After all DUs are *READY*, they all have loaded the right operational parameters and the only missing step is setting them to *RUNNING*. The transition from *READY* to *RUNNING* needs to be fast, so it performs some simple operations in the hardware such clearing the CB TTCrx counters and enabling the optical links inputs in the TELL1s.

11.1.2 AUTOMATIC ACTIONS AND DETECTOR SAFETY AT THE ECS SCOPE

Automatic actions on the ST FSMs are needed to grant the detector safety. As the frequency of abnormal situations is expected to be low, the granularity of the control system automatic reactions will be at the half station level for the IT and at the quadrant level for the TT. A control node per half-station/quadrant will perform the corresponding automatic actions and display the equipment state. These nodes will be grouped under a top node that handles the overall state of the detector. The reason for this structure is also driven by the fact that one SPECS link corresponds to a half-station or quadrant and this is the minimal entity that may be powered off.

Table 11.1 shows the truth table with the detector states depending upon the status of the different parts. For simplicity, the *NOT ALLOWED*, *ERROR* and *EMERGENCY OFF* states definition have been summarized each one in one single row, detailing below the state definition as a foot note. The table shows a distinction between LV MARATON and LV HYBRIDS, but in the actual implementation there isn't a dedicated FSM DU type for controlling the hybrids LV regulators. The powering of the hybrids is controlled by the DUs representing the LV partitions. These DUs not only handle the hybrids but also the powering of the DBs, so

the switch on/off command affects to both. The special command *SwitchHybridsOff* will permit to switch off only the hybrids and keep the DBs powered so the temperature monitoring of the hybrids also remains running. The hybrids temperature monitoring is performed by the DCU devices housed in the DBs, that is the reason why they need to be powered. The state of the DU in that case will display *HYBRIDS_OFF* and is upwards to the IT_DCS_TOP node. The IT_DCS will display *NOT_READY* since its type is common to all sub-detectors and *HYBRIDS_OFF* is not envisaged as a possible state. Under these circumstances any attempt to program the DAQ will end up reporting an error message.

The Table 11.1 describes the possible states in which the detector may be in.

	HV	LV MARATON	LV HYBRIDS	COOLING	TEMP	HUMIDITY
READY	READY	READY	READY	READY	READY	READY
OFF	ST1 || OFF	OFF	OFF	OFF	OFF	OFF
SAFE FOR INJECTION	ST1 || OFF	READY	READY || OFF	READY	READY	READY
SAFE FOR SHUTDOWN	ST1 || OFF	READY	READY || OFF	READY	READY	READY
NOT ALLOWED	X	READY	READY	Not READY	X	X
NOT ALLOWED*	Not OFF	Not OFF	Not OFF	Not OFF	Any OFF	Any OFF
ERROR **	Any in ERROR	Any in ERROR	Any in ERROR	Any in ERROR	Any in ERROR	Any in ERROR
EMERGENCY OFF ***	Any in EM_OFF	Any in EM_OFF	Any in EM_OFF	—	—	—
NOT READY	Not any of the previous states.					
(X means state not relevant) * It is not allowed that the detector is not powered off and either the TEMPERATURE or HUMIDITY monitoring not running ** The detector will be in ERROR if any of the domains is in that state *** The detector will be in EMERGENCY OFF any of HV, LV MARATON or LV HYBRIDS in that state						

Table 11.1: Truth table of the detector states.

The safety of the detector is handled from two different perspectives, i.e. there will be two sets of state machines that will implement the needed mechanism to guaranty the detector integrity:

1. FSMs for the intrinsic safety of the detector independently of the accelerator status, that is without taking into account whether particles passing through or not. It usually has to do with over-temperatures, over-voltages…
2. FSMs for the detector safety with respect to the beam conditions, i.e. the status of the detector elements needs to be such that particles' injection can be performed without damaging the detector. This usually has to do with the detector biasing.

11.1.2.1 INTRINSIC SAFETY

In order to implement reliable FSMs it is important to keep the number of possible states to a minimum. This also eases the integration under the overall LHCb control tree. We explain next a way to implement the detector safety state machines using the LHCb DCS domain as template.

To guarantee that the detector is running in a safe way, the HV status only matters in case the temperature monitoring is not running. In order to assure that the detector is not mishandled when the HV is on, for instance, by opening any of the detector boxes, the expert operator has to verify beforehand the status of the system, i.e., check that the HV is *OFF*. This way we get rid of the *SAFE FOR SHUTDOWN* state and simplify the state machines.

In the following table the *OK* state means that it is operating in safe mode. This has to be the default state independently of the LHC machine status. It must be pointed out that *OK* doesn't mean that the detector is ready for data taking as the hybrids could be off. The fact that the detector is ready for data taking is handled by the main control tree and in case the hybrids are off the DAQ domain cannot be configured and thus can never go to *READY* or *RUNNING*.

	HV	LV MARATON	LV REGULATORS	COOLING	TEMP	HUMIDITY
OK	NOT ERROR	READY	READY || HYBRIDS_OFF	READY	READY	READY
OFF	ST1 || OFF	OFF	OFF	OFF	OFF	OFF
NOT ALLOWED	NOT ERROR	READY	READY	Not READY	X	X
NOT ALLOWED*	Not OFF	Not OFF	Not OFF	Not OFF	Any in OFF	
ERROR **	Any in ERROR					
EMERGENCY OFF ***	Any in EM_OFF			—	—	—
NOT READY	Not any of the previous states.					

Table 11.2: Truth table for the detector intrinsic safety.

The *OFF* state is not a desirable situation for the detector because it implies that the cooling plant is not working and this can have a impact in the silicon sensors aging after irradiation.

The *NOT ALLOWED* states are situations that should never happen. On one hand, the front-end hybrids should never be powered if the cooling plant is not in *READY* and on the other hand the temperature and humidity monitoring has to be running whenever the other domains are not off. In principle, when for any unknown reason one of these situations happens and automatic action that sets the system in a safe state is triggered. In addition, as the FSMs work like Mealy state machines, the *NOT ALLOWED* state does not need to be coded. As soon as any of the conditions that lead to this state takes place the corresponding automatic action will set the system into a safe state:

1. If the cooling plant is *OFF* or in the HV system should be set to *OFF* and the LV system to *HYBRIDS_OFF*.
2. If the temperature is OFF or NOT READY the HV and LV systems of the corresponding half-station or quadrant will be powered off after a delay of 60 seconds. The reason for this delay is to avoid triggering an automatic action during a normal switch off, where the humidity and the temperature are sequentially switched off before the low voltage.
3. If the temperature of the box or the temperature of the hybrids is in ERROR state, the corresponding part of the system may be set to *HYBRIDS_OFF*. If the temperature in the backplane or in the CB is in ERROR the corresponding part may be set to *EMERGENCY OFF*.
4. If the LV system is in *ERROR* state, the corresponding region of the system will be switched off with the *EMERGENCY OFF* command.

In case of an *ERROR* state, the corresponding part of the system may be set to *OFF* or *EMERGENCY OFF*.

The *NOT READY* state should be considered as a combination that is not foreseen and hence not good for the detector. It should only happen at the initialization of the FSMs or when doing manual operations in single elements of the system. It implies that the operator has to take care of the detector until a safe state is granted (normally *OK*).

As a result of the previous explanations only five states are needed: *OFF*, *NOT READY*, *OK*, *EMERGENCY OFF* and *ERROR*.

11.1.2.2 SAFETY WITH RESPECT TO THE LHC BEAM CONDITIONS

The *SAFE FOR INJECTION* is the only state that has to do with the status of the LHC machine. It is a common feature that the HEP detectors need HV for biasing the sensors and a high density of particles passing through while they are biased may be a cause of damage. For this reason the particle injection stage is one of the critical moments and the detectors should be kept un-biased until the beam is stable. In principle, the machine injection will take place only if all detectors are in safe state. Biasing is the last step just before data-taking and will take place when the DAQ is ready and the particles are already passing through.

A parallel control unit at the LHCb scope implements the logic to decide whether it's safe to bias the different LHCb detectors. HV rules for each sub-detector are defined in a table for each of the defined LHC states:

- 'INJECTION': the beam is going to be injected. The HV is set to *STANDBY1* and the LV of the FE hybrids is on.
- 'RAMP': the beam energy is being ramped up. The HV is set to *STANDBY1* and the LV of the FE hybrids is on.
- 'PHYSICS': the beam is stable and normal data taking may proceed. The HV is ramped up to the nominal HV and the HV hybrids are on.
- 'PHYSICS ADJUST': restricted readjustments of the beam can take place. The HV is kept at nominal HV as in 'PHYSICS'.
- 'ADJUST': the beam will be readjusted and could be dumped.
- 'DUMP': the beam is going to be dumped. The HV is set to *STANDBY1* and the LV of the FE hybrids is on.
- 'NO BEAM': there is no beam. The HV can be at any state and the LV as well.
- 'MACHINE DEVELOPMENT': any kind of activity can go on in the machine. The HV is set to *OFF* and the LV to *HYBRIDS_OFF*.
- 'EOF': special state to force the detector to go to a safe HV state, as to mark the end of a fill.

For each of these LHC states each sub-detector defines the wanted state for the HV. Besides it is possible for the sub-detectors to define whether they want permit sending actions to the HV from the LHCb central control panel. If sending actions is not permitted then it is the responsibility of the sub-detectors to put the HV to the right state. When the current state of the sub-detector HV matches with the expected in the table the control logic decides that the system is *OK* for going on with the corresponding LHC operation.

The sub-detectors also have the option to choose as wanted state '_ANY_', which means that any HV state from the sub-detector is fine for the corresponding LHC state. This is useful for the 'NO BEAM' periods, i.e. there is full freedom while the LHC is off.

Ready for injection is a safety issue so the ownership of the system is bypassed so that the command is sent unconditionally.

Only the Operation and Run Coordinators are allowed to change this table after discussion with the SD responsible.

11.1.3 DSS INTERCONNECTION

The Detector Safety System (DSS) is a separate system that acts independently of the ECS in order to assure the detector safety [79]-[81]. The system will usually power off the racks of the IT or TT equipment and/or triggers a CO_2 extinguisher system. Although it is a completely stand-alone system the signals generated by the DSS are published as DIP services and they are gathered by a dedicated DU that will be part detector intrinsic safety. Whenever any DSS action is triggered the DU will display an *ERROR* state and the alarm screen will describe the raised error.

11.1.3.1 THE EXISTING DSS SIGNALS: SENSORS AND ACTUATORS

The DSS equipment protection can be divided in three parts: the first is for the detector itself (with thermo switches, the signal from the cooling PLC, the Sniffer lines); the second for the racks equipments (electronics, power supplies) which is 'common' for all the LHCb racks (Smoke detection, Thermo switch, water leak detection, CO_2 extinguisher system); the third corresponds to the 'Beam Approach Monitoring System' to warn about the Inner Tracker in contact with the beam pipe. In addition to these input signals from different sensing elements, there are output signals to cut the electrical power, open CO_2 flushing valves or switch off the HV.

The following DSS signals exist for the detector:

1. Thermo-switches for detector boxes and service-boxes. Thermo-switches are installed in the Detector Boxes (two per quadrant in TT and one per detector box in the IT) and glued on the cooling block of each Service Box. The DSS cuts the power of the detector in case of overheating.
2. Mixed Water Cooling for SBs: the service boxes are cooled using the LHCb common mixed water system. An alarm is triggered in case of problem with the mixed water cooling (no flow or

temperature above 25 ºC for more than 5 s.). For the TT there is also a flow switch that detects whether there is water flow in the SBs circuit. DSS cuts the electrical power of the equipment concerned.

3. Water Leak Detection: a detection cable is installed below the TT service boxes.

4. Nitrogen Flushing for the Detector: the supply of the TT detector with nitrogen is measured by two flow switches. If the Nitrogen stopped for longer than 30 minutes the CCC/Infrastructure (Cern Control Center) need to be called for this problem. If the situation is not re-established before 8 hours the cooling plant temperature needs to be increase. The DSS will just send and alarm for the operator to take an action.

5. C_6F_{14} Cooling PLC Signal: in case of problem with the detector cooling plant (C_6F_{14}), the controlling PLC provides an open contact and after a delay of 30 s the power of the detector is cut.

6. High Voltage door switches: to protect the sensors from sudden HV disruptions, doors with switches are installed in front of the patch panels. The HV PS will be powered off in the event of this signal.

7. Smoke Sniffer System for the detector: one signal is given by the Sniffer System PLC in case of smoke detection around the detectors. It will trigger a power cut.

8. Water Mist System above Outer Tracker: even if the system could be activated automatically by DSS, for the moment it is only triggered manually with 2 valves in UX85-B (after the chicane where the bottles are installed). DSS receives 3 signals from the system: the status (valves position, water pressure...), the pressure of all bottles and the 'Veto' valve. The two first signals are only to monitor the system. When the Veto valve open signal is received, DSS switches off the power of all sub-detectors.
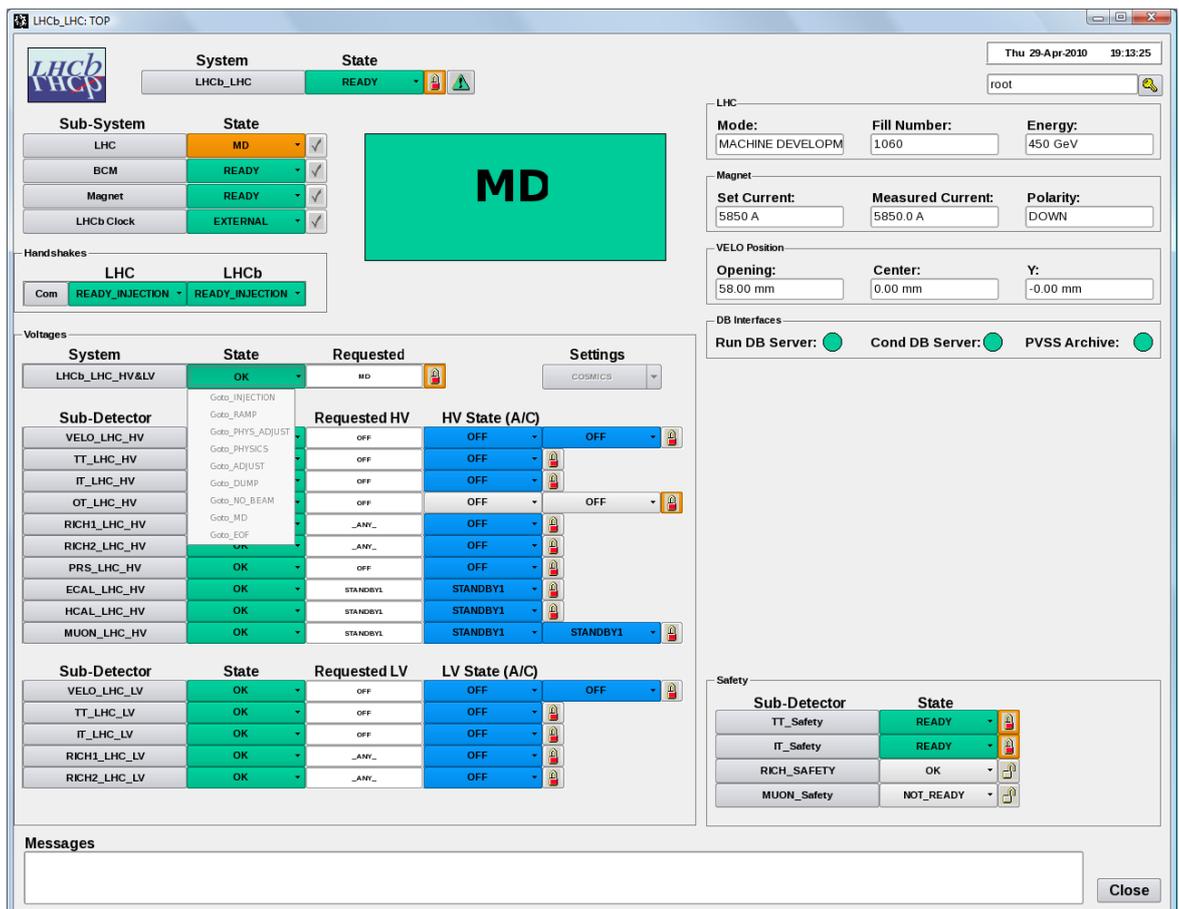
Figure 11.1: The LHCb-LHC panel showing the status of the HV for the "MACHINE DEVELOPMENT" LHC state.

The following DSS items concern the racks with the electronic equipment. A 'turbine' (air cooling unit) with a smoke detector and a thermo switch is installed in each ST rack in the counting house behind the concrete wall. The racks in the experimental area have no turbine nevertheless they are also equipped with a thermo switch and a smoke sensor.

1. Racks thermo switch: the DSS cuts the electricity in the racks if one thermo switch opens.
2. Racks smoke detection: the smoke detection PLC gives one contact to DSS per smoke detector and the action in case of alarm is powering off the rack. If the thermo-switch also signaled an over-temperature it means that there's fire and the CO_2 extinguisher system is activated together with the barrack sirens.
3. Racks water cooling: if the mixed water cooling goes above 25 ºC or there is not water flow for a certain space of time the power of the racks is cut off.
4. Racks water leak detection: water leak detection cables installed in the floor cover all the racks placement but for the TT ones in the balcony. In case of water leak detection the valves of the circuit are closed and electrical power is cut for the equipment concerned.

The tables below show a summary of the different DSS signals, the action delay and the action taken. The first one corresponds to the detector related signals and the second to the racks.

With respect to the IT beam Approach Monitoring System, it is installed on the detector boxes to have an alarm if an IT detector box gets too close to the beam pipe. DSS does not take any action but is used to trigger an alarm and inform people in case of contact between the Beam Approach Monitor System and the beam pipe.

DSS signal	Delay	Action
Thermo-switches Detector box	30 s	Power racks off
Thermo-switches Service boxes	30 s	Power racks off
Svce boxes Mixed water temperature over 25ºC	30 s (TT) and 0 s (IT)	Power racks off
Mixed water flow TT service boxes	30 s	Power racks off
TT service boxes water leak	30 s	Power racks off and cut mixed water
Detector nitrogen flow in TT box	30 s	
C_6F_{14} Cooling not OK	30 s	Power racks off
HV patch-panel door open	30 s	Switch HV off
Smoke sniffer close to the detector	30 s	Power racks off
Water mist system	0 s	Power racks off

Table 11.3: Detector DSS signals.

DSS signal	Delay	Action
Mixed water temperature over 25ºC more than 10 s.	0 s	Power racks off
Mixed water flow during 600 s.	0 s	Power racks off
Water leak in IT, OT, MUON and RICH2 racks in cavern	0 s	Power racks off and cut mixed water
Water leak in the counting room false floor	0 s	Power racks off and cut mixed water
Thermo-switches racks D3	30 s	Power racks off
Thermo-switches racks cavern	30 s	Power racks off
Smoke detection in racks	30 s	Power racks off
Smoke detection in racks + thermo-switch	10 s	Power racks off + CO_2 release

Table 11.4: Racks DSS signals.

11.2 ST DETECTOR SAFETY FSMs IMPLEMENTATION

The Figure 11.2 and Figure 11.3 show the parallel control trees which implement the detector intrinsic safety procedures for IT and TT respectively. The dark blue color corresponds to CUs, the light blue to LUs and the grey box represents the DU that provides the DSS information.

There is a top CU that summarizes the overall status and takes high level actions and then a geographical subdivision of the detector into half-stations or quadrants, also implemented as CUs. The DSS DU hangs directly from the top node, as well as the humidity, cooling plant control and other DUs intended to monitor the status of several relevant software entities. A subset of LUs hung from the half-station (quadrant) CUs.

This domain oriented subdivision links to the respective nodes of the main control tree, i.e. each of the LUs of the safety tree connects to the corresponding CUs, LU or DUs of the ECS tree (see Figure 11.4):

- 'HV': connects the HV CU that controls the channels corresponding to that half-station (quadrant).
- 'TEMP box': links to the DUs which read the interior box temperatures of the half-station (quadrant).
- 'TEMP Svce boxes': connects to the DUs used for the temperatures monitoring of backplanes and CB in the same half-station (quadrant).
- 'TEMP hybrids': links to the PT1000 sensors on the hybrids.
- 'LV MARATON': connects to the LUs that control the MARATON channels that power the service box in the same half station (quadrant).
- 'LV Regulators': connects to the DUs which control the LV regulators which power the hybrids and digitizer boards in the same half station (quadrant).
- 'Humidity': links to the CU that holds the humidity sensors installed in the detector.
- 'Cooling plant': the C_6F_{14} cooling plant.

Next we do a bottom to top description of the FSMs behavior.

11.2.1 THE DSS DEVICE UNIT

The underlying idea behind the DSS DU is to power OFF the equipment in a gentle way before the DSS system triggers a sudden power off of the racks. The DSS info is available in the ECS via the DIP protocol, so this DU implements the subscription of the different signals. In the DSS most of the actions power off the equipment within a reasonable delay, time enough for the safety FSM to power off beforehand. The DSS DU implements the FSM states depicted in the Table 11.5.

READY	The DU is actively monitoring the status of the DSS signals and they are all fine.
ERROR	Any DSS signal has been triggered and a DSS action will take place within the programmed 10 or 30 s delay.
NOT_READY	The DIP communication with the DSS is not established properly.

Table 11.5: DSS DU states.
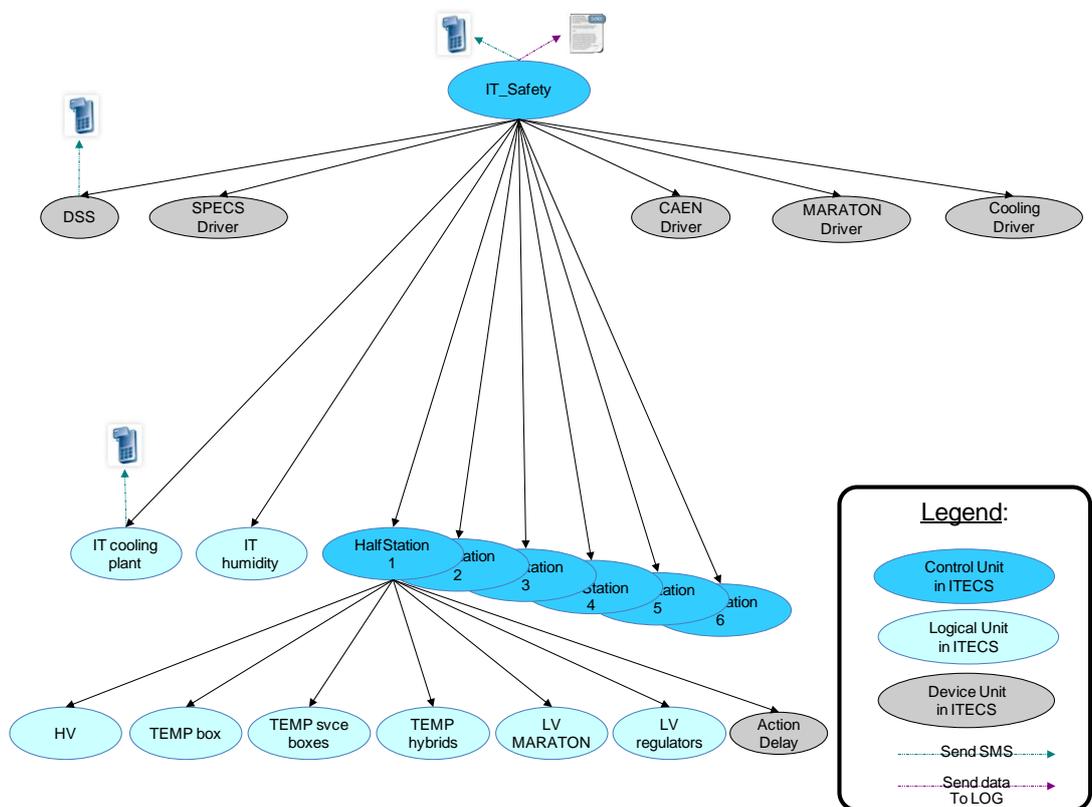
Figure 11.2: IT safety hierarchy.
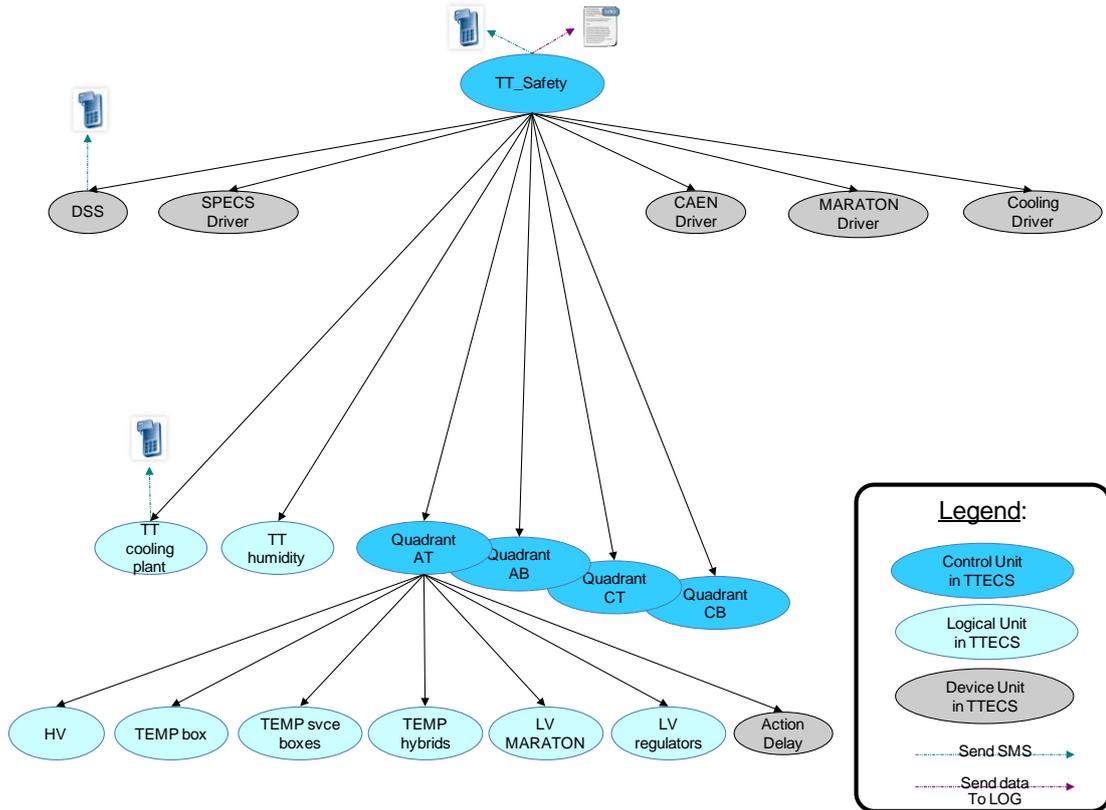
Figure 11.3: TT safety hierarchy.
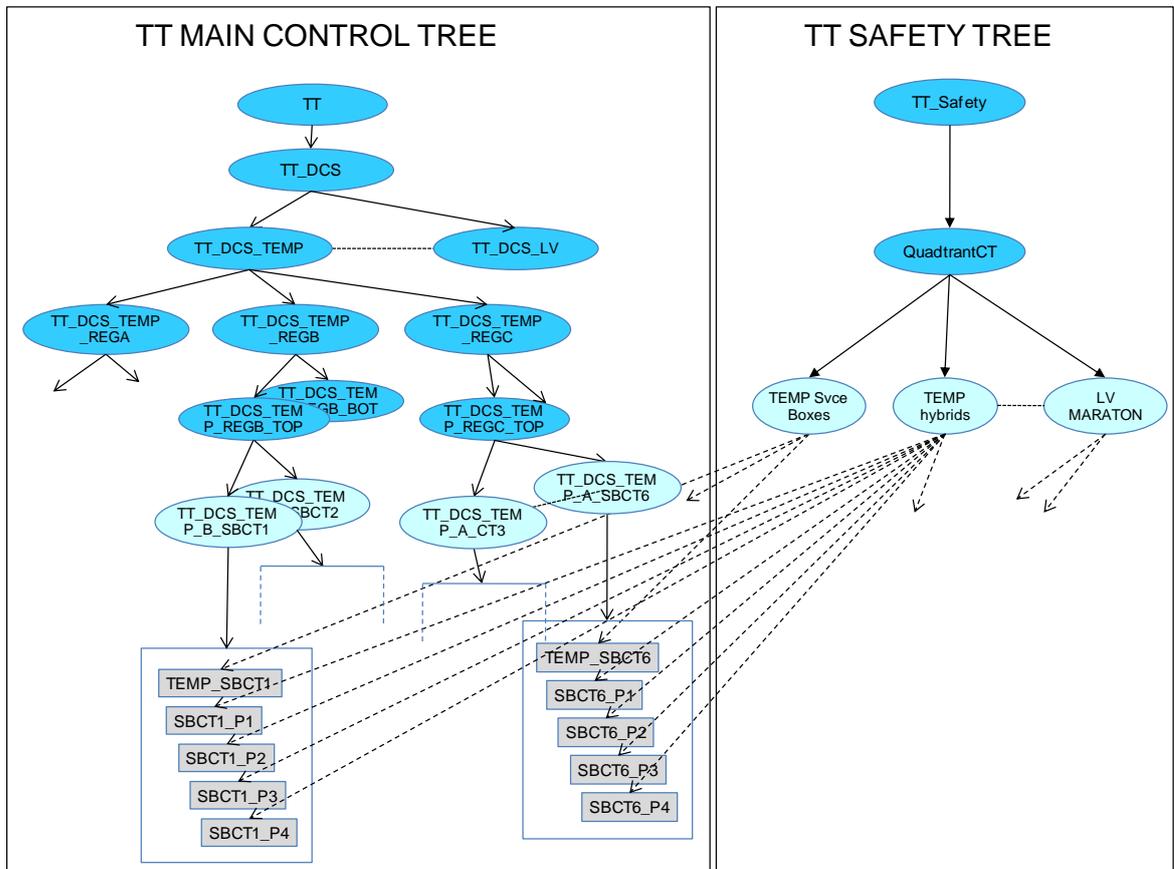
Figure 11.4: Example of interconnection between the main and safety control trees. It only shows how the SB and hybrid temperatures match in the two different hierarchies.

11.2.2 THE SPECS, OPC AND COOLING DRIVERS STATUS

In addition it is crucial to introduce some additional safety measures to check that some relevant software entities are running, namely:

- The SPECS DIM-server and DIM-client. If any is not running this DU will be sent to *NOT_READY*.
- The status of the CAEN and Wiener MARATON OPC servers. This DU will be sent to *NOT_READY* if the servers are not running or stuck.
- The status of the PVSS manager that provides control and supervision for the cooling plant parameters.

These dedicated DUs hang from the top IT and TT safety node. An alert message will be sent to the main LHCb alert screen in case any of these elements stops running properly.

The FSM corresponding to these DUs only need two states:

READY	The software entity (SPECS Server/Client, the OPC server or the Cooling manager) is properly running.
NOT_READY	The software unit is not running.

Table 11.6: States associated with OPC, SPECS and Cooling DUs.

11.2.3 THE FSMs' STATES DEFINITION AND TREE DESCRIPTION

As explained before just five states are necessary for the CU and LUs in the safety tree: *OFF*, *NOT_READY*, *OK*, *EMERGENCY_OFF* and *ERROR*. The result is that the FSM definition for the CUs is the same as for the DCS domain (Figure 11.5), with the only difference that the *OK* is replaced by *READY*. The table describes the states of the FSMs for all the nodes but the LV partitions. To cope with the *HYBRIDS_OFF* state of the partitions and make it easily visible in the safety tree the behavior of the "LV regulators" LU adds the *HYBRIDS_OFF* state (Table 11.8).

The states in the tables are defined in terms of the state of the children. It is important to point out that the *READY* state means for the top nodes that the detector is powered on, with all the supervision services running properly and all the monitored parameters show that the detector is running in a safe way. This implies that the HV may either be on or off, as well as the hybrids. *OFF* means that the detector is not powered at all, although the DSS signals, the SPECS masters and slaves, the Cooling, the CAEN OPC and the Wiener MARATON OPC are still being monitored.

Additionally, in the states definition one can easily find incoherence between the *ERROR* and *EMERGENCY_OFF* states. A CU can have children in both of the states, so was decided to resolve this ambiguity by giving higher priority to the *EMERGENCY_OFF*.

The hierarchy of state machines for all the safety LUs and CUs has already been depicted. The only nodes that perform automatic actions are the half-station/quadrant and top. The actions described in the following tables are only implemented for the *READY* state of the corresponding node. By the nature of the system itself, whenever the detector is operated and the DCS is *READY* these FSMs should also automatically reach the *READY* state. The system has to work as a whole in order to get take data. Any operation of the system that does not leave the safety FSMs in *READY* state should not be allowed; hence the operator needs to make sure about it so automatic actions are "enabled".
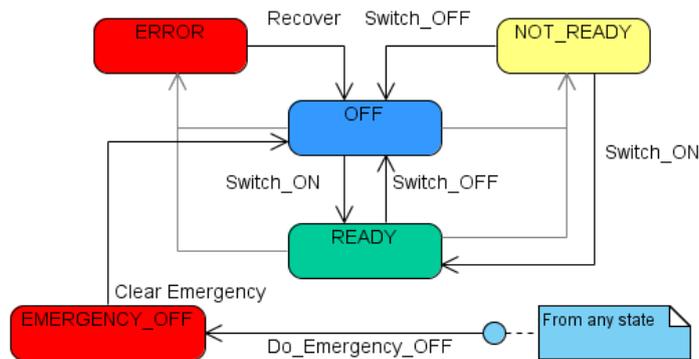
Figure 11.5: FSM definition for the ECS automation.

OFF	All the children are OFF or RAMPING_OFF (HV).
READY	All the children are in READY or in the following HV states: STANDBY1, STANDBY2, RAMPING_STANDBY1, RAMPING_STANDBY2 or RAMPING_READY.
ERROR	Any of the detector elements is in ERROR.
EMERGENCY_OFF	Any of the detector elements is in EMERGENCY OFF.
NOT_READY	The system is in a transitional state or not properly initialized.

Table 11.7: ST states description for the CUs and LUs in the safety tree.

OFF	All the children are OFF or RAMPING_OFF (HV).
READY	All the children are in READY.
HYBRIDS_OFF	All the children are in HYBRIDS_OFF and READY.
ERROR	Any of the detector elements is in ERROR.
EMERGENCY_OFF	Any of the detector elements is in EMERGENCY OFF.
NOT_READY	The system is in a transitional state or not properly initialized.

Table 11.8: ST states description for the "LV regulators" LU.

HALFSTATION_SAFETY | QUADRANT_SAFETY

- When LV MARATON or LV regulators in *ERROR* ⇨ LV *ForceOff*:

  It powers off the MARATON channels and sets the DUs which control the regulators, temperatures and humidity to *OFF*. This command also clears fake alarms caused by the MARATON being off and stops the monitoring of the affected DUs.

- When the TEMP BOX or TEMP Hybrids in *ERROR* ⇨ LV *SwitchHybridsOff*:

  This means that the temperature inside the box is either too high or there is a hardware problem related to the temperature readings. In this case the hybrids are switched off while the DBs are kept on. This command sets the LV Partitions into *HYBRIDS_OFF* state and hybrid's temperature monitoring continues running.

- When the TEMP Service Box in *ERROR* ⇨ LV *EmergencyOff*:

  The backplane or CB temperature is either too high or there is a problem with the readings. The safest in this case is powering off the MARATON channel, thus the whole half-station is set to the *EMERGENCY_OFF*.

- When the TEMP BOX or TEMP hybrids or TEMP Svce Box is *OFF* or *NOT_READY* for more than 60 s ⇨ HV *GoToOff* & LV *EmergencyOff*:

  If the temperature monitoring is not running, it is not safe to keep the detector powered so it must be switched off. We use the *EmergencyOff* since this is the only way to disentangle why everything went off, as this action is not preceded by any error status in any DU. The 60 s time is needed for the normal switch off of the system. The sequential switch off (temperatures, LV regulators and LV MARATON) takes usually takes around 30 s so the 60 s latency is needed to avoid false *EMERGENCY OFF*.

Table 11.9: Actions at the half-station/Quadrant scope.

IT_SAFETY | TT_SAFETY

- When the DSS in *ERROR* ⇨ HV *GoToOff* & LV *EmergencyOff*:

  This action immediately powers off the LV and tries to ramp-down the HV. The ramping down will be accomplished on time as long as the ramping rate is fast enough to react within a couple of seconds.

- When the COOLING PLANT is *OFF* or ERROR ⇨ HV *GoToOff* & LV *SwitchOffHybrids*:

  If the cooling plant is or goes to *OFF* the hybrids must be powered off. The temperature monitoring will continue running.

- When the SPECS driver is *NOT_READY* ⇨ LV *SwitchOff*:

  If either the SPECS server or client is not working properly the MARATONs LV will be switched off as the monitoring is not running.

Table 11.10: Actions at the top level.

In order to have these FSM's actively working the control of the CUs must be taken. A PVSS control script takes automatic control of the safety FSMs 15 s immediately after they are "alive" and not taken by any user. This control script is set to "Run always" mode so if any expert operator needs to take stand-alone control of these FSMs it has to force release the control of the CU from the expert menu and take the control for himself within 15 s., otherwise the FSMs will be taken again by the control script.

The panel associated with the safety top node (Figure 11.6) displays a table with the overall status of the different devices involved in safety issues. The table shows the number of elements in a given state for each of the sub-domains (half-stations in the IT and quadrants in the TT) so it is quick to trace the source of any automatic action. The panel also shows a log with the last 50 automatic actions triggered, sorted in reverse time order, and with the possibility to save this log to a file. Each entry of this log includes the action taken, the children status and the time stamp. Finally there is a table where the CUs which are 'not enabled' (that is in DEAD state) are shown. This last table is important because the children of those CUs will not be included in the summary table and are not being monitored, so if any CU appears in this table is must be a priority to fix it.

Figure 11.6: IT Safety panel showing the status of the DUs.

The status of the safety FSMs will be displayed in a panel in the LHCb central console so the shift operator can monitor the status of the safety FSM and call ST piquet if the status is different from *READY*.

11.2.4 EXPERT MANUAL CONTROL OPERATIONS

If an expert wants to perform manual operations on the system such as switching off the temperature monitoring when the system is running, he has to make sure to disable the corresponding elements in the safety tree. Otherwise, it might be possible that one if of the previously defined conditions is fulfilled and an automatic action triggered.

The *Clear_Emergency* command must be sent from the DCS node to recover from the *EMERGENCY_OFF* state. The same applies to the *Recover* command for the *ERROR* state.

11.3 SILICON TRACKER ALERTS

The alert panel displays relevant information about problems occurring in the system. Either measurements going off the expected limits or DUs going to error because of failure to execute a command will display an alert with a relevant message in the alarm screen.

Next is the list of measured parameter with associate alarms. Their current default ranges and alert messages is the following:

Parameter	Alert status	Alert range IT	Alert range TT	Alert message
Hybrid Temperature [ºC]	OK	-12<T<30	-12<T<30	OK
	WARNING	-25<T<-12 || 30<T<60	-25<T<-12 || 30<T<60	Temperature (HIGH | LOW) in hybrid
	ERROR	T<-25 || T>60	T<-25 || T>60	Temperature very (HIGH | LOW) in hybrid
Box Temperature [ºC]	OK	-15<T<30	-15<T<30	OK
	WARNING	-15<T<-20 || 30<T<45	-15<T<-20 || 30<T<45	Temperature (HIGH | LOW) in Detector Box
	ERROR	T<-20 || T>45	T<-20 || T>45	Temperature very (HIGH | LOW) in Detector Box
Backplane Temperature [ºC]	OK	10<T<40	10<T<40	OK
	WARNING	0<T<10 || 40<T<60	0<T<10 || 40<T<60	Temperature (HIGH | LOW) in Backplane
	ERROR	T<0 || T>60	T<0 || T>60	Temperature very (HIGH | LOW) in Backplane
Control Board Temperature [ºC]	OK	10<T<45	10<T<45	OK
	WARNING	0<T<10 || 45<T<60	0<T<10 || 45<T<60	Temperature (HIGH | LOW) in Control Board
	ERROR	T<0 || T>60	T<0 || T>60	Temperature very (HIGH | LOW) in Control Board
Detector Box humidity [%RH]	OK	H<15%	H<15%	OK
	WARNING	15%<H<20%	15%<H<20%	Dew-point (HIGH | LOW) in Detector Box
	ERROR	H>25%	H>25%	Dew-point very (HIGH | LOW) in Detector Box
Detector Box dew-point margin [ºC]	OK	ΔT>10	ΔT>10	OK
	WARNING	10<ΔT<5	10<ΔT<5	Dew-point margin LOW in Detector Box
	ERROR	ΔT<5	ΔT<5	Dew-point margin very LOW in Detector Box

Table 11.11: Environmental parameters.

Parameter	Alert status	Alert range IT	Alert range TT	Alert message
Caen channel Current [A]	OK	I<6	I<50	OK
	WARNING	6<I<8	50<I<100	Current HIGH in Caen Channel
	ERROR	I>8	I>100	Current very HIGH in Caen Channel
Ch1 Maraton Current (full Svce Box) [A]	OK	I<25,2	I<24,2	OK
	WARNING	25,2<I<27,7	24,2<I<26,4	Current HIGH in Maraton Channel
	ERROR	I>27,7	I>26,4	Current very HIGH in Maraton Channel
Ch1 Maraton Current (not full Svce Box) [A]	OK	I<18,8		OK
	WARNING	18,8<I<20,5		Current HIGH in Maraton Channel
	ERROR	I>20,5		Current very HIGH in Maraton Channel
Ch2 Maraton Current (full Svce Box) [A]	OK	I<10,2	I<9,4	OK
	WARNING	10,2<I<10,9	9,4<I<9,8	Current HIGH in Maraton Channel
	ERROR	I>10,9	I>9,8	Current very HIGH in Maraton Channel
Ch2 Maraton Current (not full Svce Box) [A]	OK	I<8,1		OK
	WARNING	8,1<I<8,8		Current HIGH in Maraton Channel
	ERROR	I>8,8		Current very HIGH in Maraton Channel

Table 11.12: Power supplies parameters.

Parameter	Alert status	Alert range IT	Alert range TT	Alert message
2.5V Control Board Regulator [V]	OK	2,3<V<2,7	2,3<V<2,7	OK
	WARNING	2,15<V<2,3 || 2,7<V<2,85	2,15<V<2,3 || 2,7<V<2,85	Voltage (LOW || HIGH) in 2.5V CB Regulator
	ERROR	V<2,15 || V>2,85	V<2,15 || V>2,85	Voltage very (LOW || HIGH) in 2.5V CB Regulator
5V Control	OK	4,7<V<5,2	4,7<V<5,2	OK

Board Regulator [V]	WARNING	4,5<V<4,7 || 5,2<V<5,5	4,5<V<4,7 || 5,2<V<5,5	Voltage (LOW || HIGH) in 2.5V CB Regulator
	ERROR	V<4,5 || V>5,5	V<4,5 || V>5,5	Voltage very (LOW || HIGH) in 2.5V CB Regulator
3.3V Control Board Regulator [V]	OK	3,1<V<3,5	3,1<V<3,5	OK
	WARNING	2,9<V<3,1 || 3,5<V<3,6	2,9<V<3,1 || 3,5<V<3,6	Voltage (LOW || HIGH) in 2.5V CB Regulator
	ERROR	V<2,9 || V>3,6	V<2,9 || V>3,6	Voltage very (LOW || HIGH) in 2.5V CB Regulator
2.5V (REFERENCE) Control Board Regulator [V]	OK	2,4<V<2,6	2,4<V<2,6	OK
	WARNING	2,4<V<2,25 || 2,6<V<2,75	2,4<V<2,25 || 2,6<V<2,75	Voltage (LOW || HIGH) in 2.5V CB VREF
	ERROR	V<2,25 || V>2,75	V<2,25 || V>2,75	Voltage very (LOW || HIGH) in 2.5V CB VREF
2.5V Ana Hybrid Regulator [V]	OK	2,3 < V < 2,7	2,3 < V < 2,7	OK
	WARNING	2,15<V<2,3 || 2,7<V<2,85	2,15<V<2,3 || 2,7<V<2,85	Voltage (LOW || HIGH) in 2.5V Hybrid Ana
	ERROR	V<2,15 || V>2,85	V<2,15 || V>2,85	Voltage very (LOW || HIGH) in 2.5V Hybrid Ana

Table 4.13: Regulators voltage measurements

A panel for enabling and disabling alarms and also for modifying all the previous alarm ranges will be available. It is expected that operational ranges of some of the parameters may change on time due to aging in the components. They may also need to be modified due to a failure in the equipment or because any special intervention in the detector needs to be done.

11.4 LV MONITORING AND CURRENT LIMITS

In our experiment the LV power supplies are located 'far away' from the detector (several tens of meters). The LV power supplies draw high currents over long distance cables and the voltage drop along the cables may exceed the voltage on the load itself (electronic circuit). In case of a short circuit on the equipment or a ground fault, the increase in current would not be large enough to trigger the classical circuit breaker protection of the power supply. Therefore there is a risk of fire.

The only effective protection of our LV circuits is to carefully monitor the currents and set a threshold few percents above the nominal known value and trigger an automatic action (power off) in case the threshold is exceeded. LV currents are important safety parameters. If they are not properly controlled the equipments cannot be left powered without continuous human supervision.

The MARATON LV current consumption is controlled and limited at three different levels:

1. At the hardware level by tuning properly the potentiometers in the MARATON channels power box.
2. A second one at the RCM (remote controller module) level, just settable connecting to it via USB.
3. A third level also in the RCM that can be set either with the USB or from PVSS via OPC. This setting is part of the LV channel recipe and the way to fine tune the current limit from the ECS.

In addition, and very important, for the cases 2 and 3, different actions can be taken at the RCM level in case the current limit is reached: "Ignore failure", "Switch the channel off", etc. All channels are configured to automatically switch off in case of failure.

The current limits for the different levels are respectively set to 125%, 120% and 110% with respect to the nominal current consumption.

The nominal and maximum terminal voltage must also be set. They are defined in the tables in appendix E. The setting depends on the voltage drop in the cable, so in the cables length and whether on the SB is completely filled with digitizer boards (half of the IT Service boxes will host 12 boards and the other half 16).

11.5 RECOVERY MECHANISM FOR FAULTY READINGS

In order to ensure the stability of the system the measurements must be reliable, otherwise we can be triggering an automatic action in the event of a fake situation.

The sources of bad readings can be many, but in the ST two have clearly been identified:

1. It has been found that the SPECS link which drives the communication between the CBs and the SPECS master in the counting room can get stuck and lead into permanent bad readings. As the

source of this problem has never been very well understood a mechanism to recover from it has been developed.

2. Oscillating temperature readings due to a malfunctioning of the DCU in the TT digitizer boards.

In addition, there could be other sources of bad readings, that is the reason why a reading retry mechanism has been implemented independently of the known problems. This mechanism is implemented at two different levels

1. A first error-check and retry at the lowest software level in the 'SpecsServer'. The SPECS protocol provides some mechanisms to check whether the data link has somehow been corrupted. The only way to overcome from this is performing a SPECS master soft-reset. The 'SpecsServer' checks whether the lowest level libraries report any kind of error and repeats the reading if needed.

2. Higher level checks in the PVSS side. In this case we look for transmission errors and check the measurement value. The transmission errors refer to SPECS link transmission errors and to errors from the 'SpecsServer' to the PVSS client. Checking the measurement value means testing whether the value is inside the normal operation ranges defined in the alarm settings. In any of the cases the recovery process is the following:

    a. Check whether the SPECS link is ok by reading the slaves date register. If the reading fails a SPECS master soft-reset is performed and the reading of the date register retried. If the problem persists the DU is set to *ERROR* and an alarm message displayed in the alert screen. If the reading of the date register succeeds we go to the step (b).

    b. Repeat the measurement reading N times (currently N=5). If after N times the reading is not within the operational range an error will be displayed in the alert screen and the corresponding DU will go to *ERROR*. This procedure provides robustness against single error readings, but it could also lead into CPU overload if for any reason there is a big bunch of bad readings. This could be the case if the TFC system fails to provide the clock, so all the readings would be corrupted.

11.6 SMS MESSAGING

In case any abnormal situation occurs in the system such the 'Safety FSMs' going to *ERROR*, the problem is reported straightaway to the ST piquet and experts via SMS. This capability is implemented with a PVSS control script called "stSMSSafety.ctl" which is set to "Run Always".

This control script is always monitoring the safety top node, the DCS_TEMP and DCS_HUM top nodes and sends a short but meaningful message in the next situations:

1. When the Cooling state goes to *NOT_READY*. The SMS will be sent each 15 minutes if the *NOT_READY* state persists.

2. When the Safety top node changes to *NOT_READY* a 2 minutes delay is started and after that the state is checked again. Only if the *NOT_READY* state persists a SMS is sent.

3. If the DSS DU goes to a state different from *READY* an SMS is sent. If the DSS state is *NOT_READY* the SMS will report that the connection with the DSS DIP manager is not working. If the DSS state is *ERROR* then a DSS alert is raised and the SMS message will include the alert text.

4. When the DCS_TEMP node goes to a state different of *READY* and the Safety top node is in *READY* state a SMS is sent.

5. When the DCS_HUM node goes to a state different of *READY* and the Safety top node is in *READY* state a SMS is sent.

Additionally, in each of previous cases an email is sent to experts with the information recorded in the safety log. An email is also sent when the safety tree is put back to *READY* from any of the mentioned situations.

CHAPTER 12

SUMMARY OF RESULTS AND PROJECT DEVELOPMENT

ABSTRACT

The aim of this chapter is to summarize the way this project has been developed in a temporal basis, from the beginning to the moment of the writing. We will briefly describe the work done and provide some results together with some conclusions of the lessons learned and the knowledge acquired. The ECS work and troubleshooting is not covered in this chapter due to its software nature, which would devote a long writing to describe all the problems and solutions adopted along the way. Furthermore, this issue is still ongoing and in continuous evolution due to the nature of High Energy Particle Physics experiments which require often new developments, functionalities and implementations.

"Es mejor morir de pie que vivir de rodillas"

Emiliano Zapata

12.1 PROJECT DEVELOPMENT

The work of this thesis has been strongly committed to the development of the control system and the commissioning of the ST detector since the very beginning.

One of the first tasks was to follow the PVSS and Framework courses at CERN. These courses provided with some knowledge of how the PVSS SCADA works and also an overview of the available software tools for the control system development. In some sense, the availability of software tools oriented the selection and design of the detector hardware as it was the case for commercial equipment: for instance, offering an OPC interface was asset in the moment of choosing the equipment. As the learning of PVSS took place in a quite early stage of the framework development, I have actively collaborated in the improvement of the software, both in the testing of the software features, the debugging and in the definition of some tools' requirements.

After that I joined the task of testing some devices for radiation tolerance as reported in [19]. The idea behind was to verify that some electronic devices could be used in the electronics placed close to the detector and have to work in an environment with radiation. The VCSEL, ADC and differential amplifier intended for the DAQ electronics passed the test. A slow multichannel ADC for the monitoring electronics did not. This left us in the situation in which all the needed elements for the data acquisition read out electronics were successfully evaluated for radiation tolerance but not those for the "slow control". The main missing element was an ADC for reading the temperature and humidity sensors. This was solvent by the CERN microelectronics group, which developed a radiation-hard multi-channel ADC initially thought for the CMS experiment. For the signal conditioning the same differential amplifier as for the DAQ could be used.

After the radiation tests the design of the HV system commenced. Two HV multi-channel modules from different vendor were evaluated, both from the software and hardware point of view. The CAEN system was chosen as the most adequate solution. In terms of I/V performance and modularity both the ISEG and CAEN solutions were quite similar, but the CAEN system offered more configurability features. From the software point of view the CAEN OPC server showed to be more flexible and stable. The fact that many more detectors intended to use similar CAEN equipment also determined the decision.

The design of LV system was also initiated by then. The studies performed depicted the Wiener MATARON power system as the most suitable. The idea of powering the front-end electronics with radiation-hard linear LV power regulators was present since the beginning due to granularity voltage distribution issues. Commercial power supplies would provide the bulk power for the regulators, so a system with modules able to provide large currents was preferred. In terms of system architecture two options were studied: place the power supplies in the "safe" area (behind the concrete wall that encloses the experiment) and reach the FE electronics with long and high cross-section cables (~60 m); and a distributed power system such as the MARATON. The MARATON power supplies are basically composed by an AC/DC rectifier and a module controller placed far from the high magnetic field and radiation environment and the DC/DC power converter modules in the hostile environment. The power transmission from the AC/DC module to the DC/DC modules in the cavern is delivered at a reasonably elevated voltage (~380V) which minimizes the amount of current that needs to be transferred and hence the need of long thick cables and power losses due to the cables resistance. This has a considerable impact in the system costs as the price of the copper has increased significantly in the latest years making the first option less attractive. In addition, the voltage regulation is usually more problematic over long distances, that's the reason why the MARATON system is more suitable since the DC/DC conversion stage is done closer to the detector (~5-8 m).

The next stage in the project was to define the technical needs for the ST control system. Some first thoughts on the hardware needs were presented in an EDR (Engineering Design Review) held at CERN by Feb 2004. It was concluded from that review that a controller card for the ST FE electronics had to be built. The development of this board became one of the main responsibilities I had to carry on. In parallel came the need to develop a system to perform temperature cycling tests in the detector modules that we were going to build for the IT. At this point it became essential to reinforce the knowledge in electronics issues such as the development of high speed PCBs (trace impedance control, LVDS, etc.), grounding, shielding and power distribution. A lot of bibliography was read and studied among books and articles [82]-[85]. All this knowledge provided a good basis for the development of the Control Board and the temperature cycling system control electronics as well as for the LV and HV systems implementation.

The CB development, production and commissioning has been one of the tasks that took me most of the time. This part of the work has been developed together with the installation of the first laboratory readout setups using most of the final electronic elements. One of the readout setups was needed to test all the IT detector modules in the temperature cycling system. Another readout setup was used for the firsts PVSS developments, Control Board tests and troubleshooting and for testing the finally assembled IT detector

boxes. Putting all the elements together was crucial in order to test all the CB functionalities and the correct functioning of the ST detector as a whole before doing the final installation in the experimental area, that is why this part of the work posed a great effort. Along the way many problems and bugs were found in the several elements of the system. Details are in paragraph 12.1.5.

The commissioning of the system in the experimental area has been another lengthy task that occupied quite a bit of my time mainly due to all the different problems found during the installation and because most of the PVSS development was performed during the system the detector installation.

In the next paragraphs I describe in more detail many of the previously mentioned issues, problems that rose up during the project lifetime as well as some results. Other matters related to the design of the system have already thoroughly described in other chapters of the thesis or documented in separate technical notes.

12.1.1 THE TEMPERATURE CYCLING ELECTRONICS

A temperature cycling system to perform ageing and quality tests of the IT readout modules had to be built. A thermally isolated box (Figure 12.1) was developed by the 'University of Santiago de Compostela'. The temperature cycling system was based on Peltier heat exchange elements and allowed for temperature cycles between -5ºC a 40ºC and controlled humidity below 1%. For this system an electronics controller and power box and low level libraries to interface via computer were developed. The schematics can be found in appendix Q. This controller box (Figure 12.2) provides:

- The powering and control of 2 Peltier cells (~12 V and ~24 A).
- Monitoring of temperatures and humidity.
- Control for a fan to homogenizing the temperature inside the box.
- Control of a gas flow switch to inject N_2 in the box and keep the humidity low.
- A parallel bus interface for computer control via 'Labview'.

Figure 12.1: Temperature cycling box.
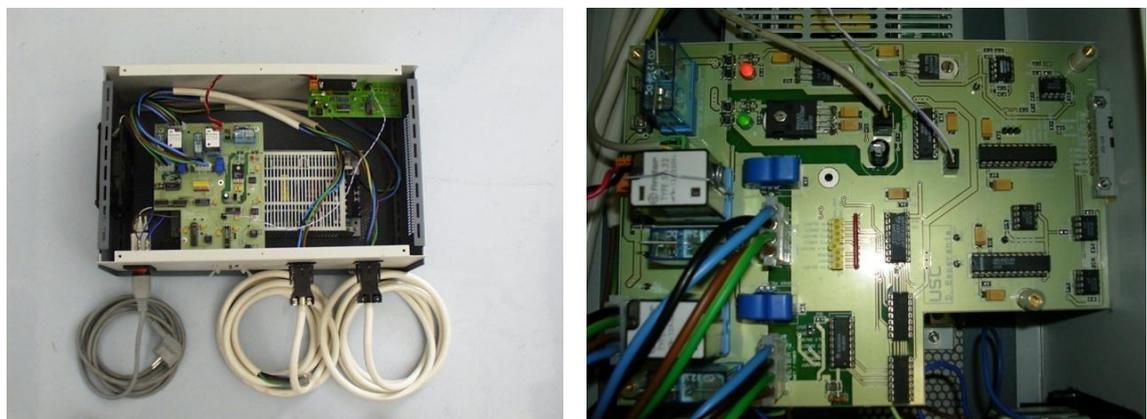
Figure 12.2: Temperature cycling electronics custom made box (left) and zoomed view of the controller card (right).

12.1.2 THE MODULE PRODUCTION AND QUALITY ASSESSMENT

Together with the temperature cycling electronics a readout setup with prototypes and ultimate hardware elements of the final system used in the experiment was developed:

- A prototype Control Board.
- The final IT readout modules: readout hybrids + the silicon sensors.
- Prototype Digitizer Boards: digitizing and optical transmission boards.
- A prototype service box backplane.
- The TELL1s. Used for receiving the optical data and transmit it via Gigabit Ethernet without any kind of processing the DAQ computer.
- The Readout Supervisor for generating all the fast control signaling.
- The computer with the SPECS master controller board and the GBE receiver board for DAQ.
- All the cabling for LV, HV, control and DAQ following the final grounding and shielding scheme.

An application to control the DAQ electronics and perform the data acquisition of the modules under test was developed. This setup was used for performing electrical quality tests on all the readout modules being built. The Figure 12.3 depicts the scheme of the setup.

We performed the module testing, burn-in and temperature cycling of all modules in the thermally isolated box. The measurements included IV-curves of the silicon sensors, monitoring of the leakage currents during the temperature cycling as well as readout of the modules using the internally provided Beetle test-pulse in order to spot broken (open/shorted) channels. Pinholes on the sensors were found by illuminating the unbiased sensors with a light bulb. From Figure 12.4 to Figure 12.7 we show samples of the tests and results obtained when testing the IT modules.
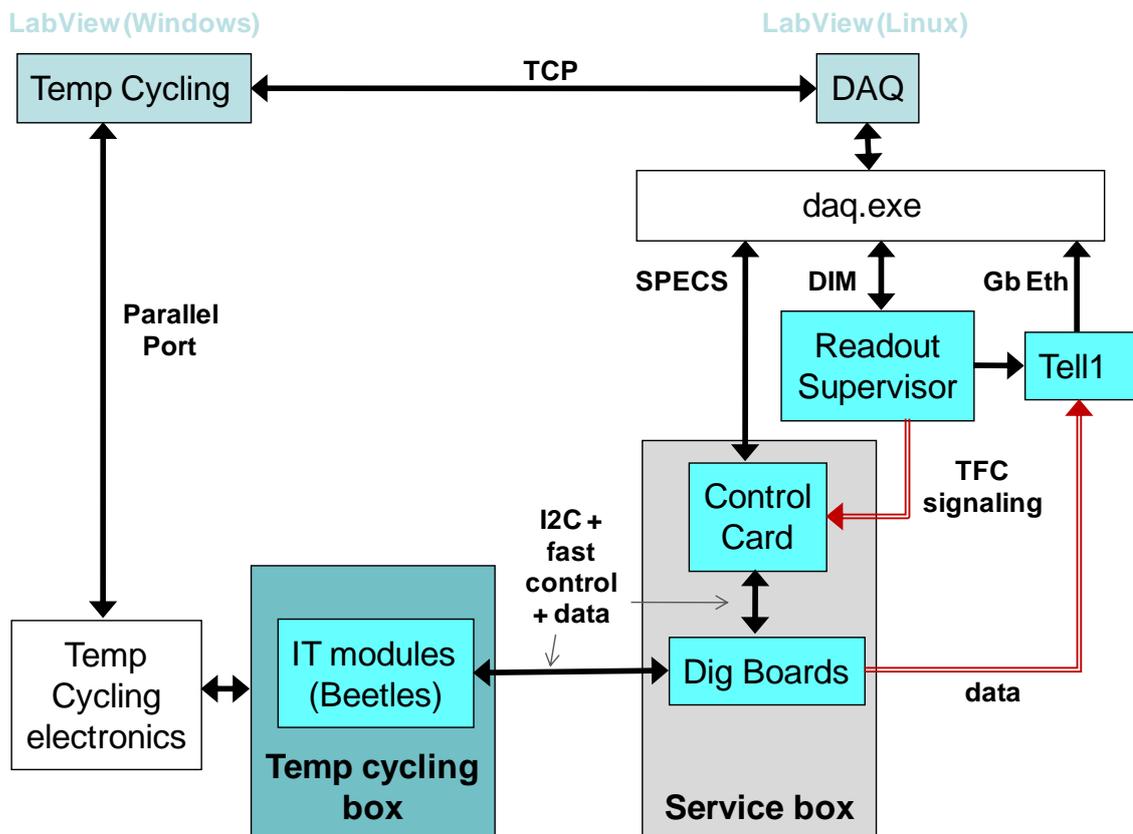
Figure 12.3: IT burn-in test-stand scheme.
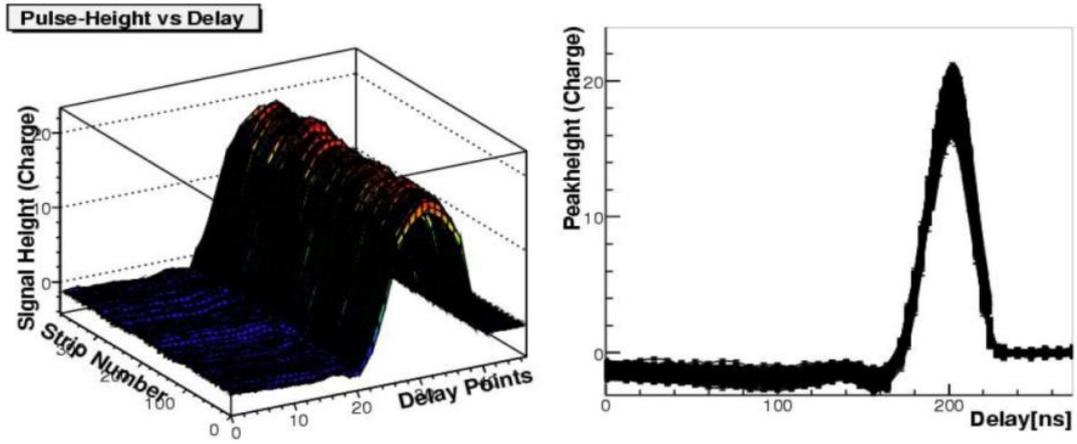
Figure 12.4: Sample of one sensor test-pulse delay scan for a module with all the channels operational. It represents the response of the sensor after signal amplification and shaping by the Beetle for different delays of the test-pulse signal. The test-pulse emulates the charge deposited by an ionizing particle in a biased sensor.
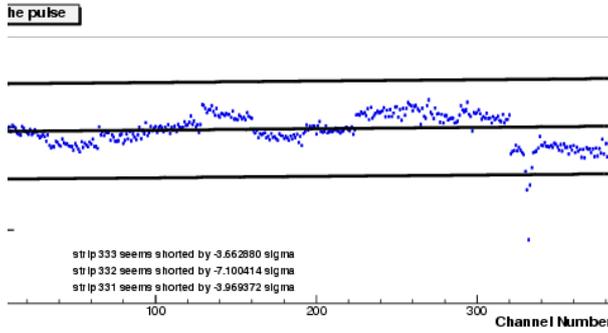
Figure 12.5: Channel scan data: pulse amplitude sampling at the maximum of the signal. Shorted channels can be spotted by their lower amplitude due to the higher acumulated capacitance of the shorted strips.
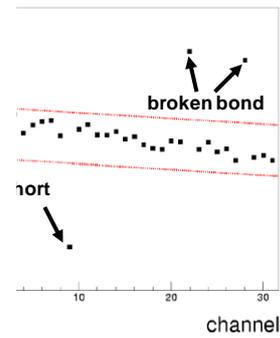
Figure 12.6: Appearance of shorted and open channels when pulsing them due to their higher an lower capacitance respectively.
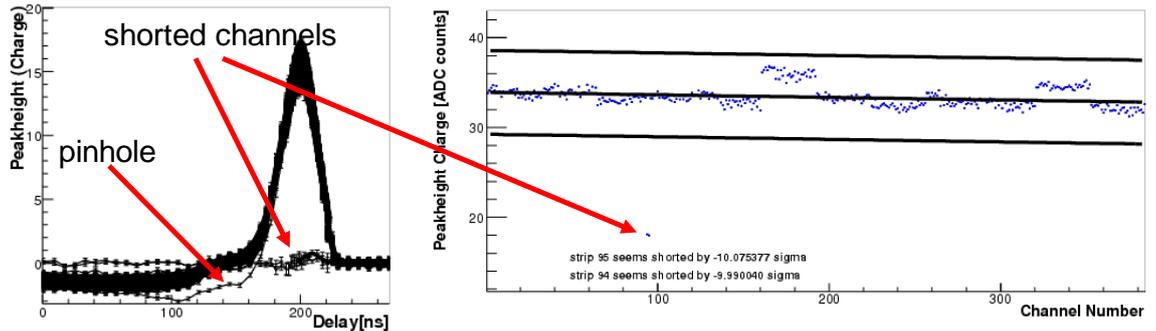
Figure 12.7: Spotting shorted channels and pinholes together in a test-pulse delay scan and channel scan.

12.1.3 THE CONTROL BOARD DEVELOPMENT AND PRODUCTION

The CB design commenced by mid-end 2004 with the production of prototype boards to test several of the devices: SPECS, DCUs and Delay25 chip (Figure 12.8). Some of them were hand-made and other manufactured. Many of the devices functionalities were assessed with these test boards. For the SPECS slave we could not go through many tests as the availability of samples was very limited and the FPGA firmware was still very simple and being developed.
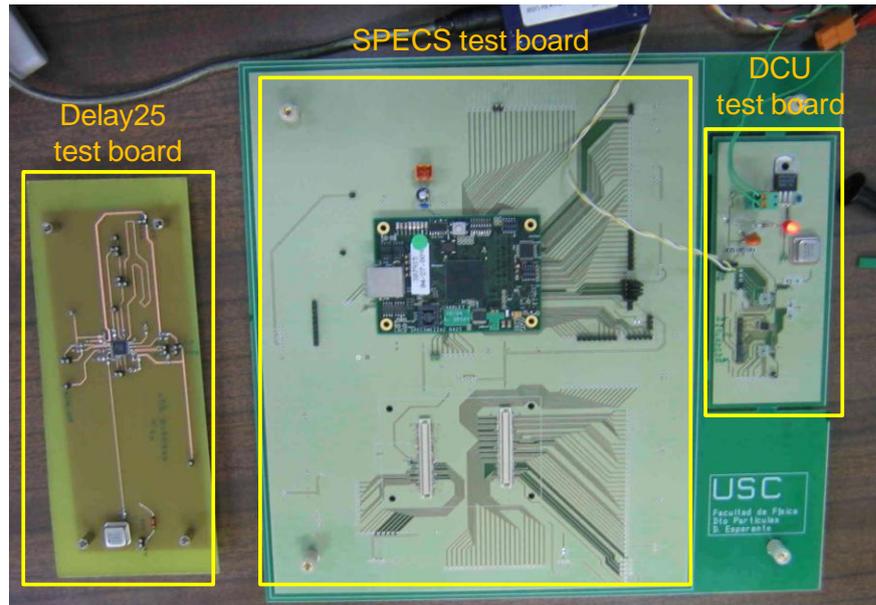
Figure 12.8: Test boards for the DCU (leftmost), SPECS (middle) and Delay25 chip (right).

After playing around with some of the devices that were going to be used in the CB we decided to develop a first version of the board. The first step before doing the final production of all the control boards was to assess in the lab that the control board fulfilled all the needed requirements for the front-end control and monitoring tasks. In the readout setup all the fast control and slow control capabilities were assessed by running the whole readout chain with the final or semi-final detector elements and also by measuring all the signals with the oscilloscope.

The CBs development went through a revision of three versions. Most of the problems were spotted in the first version and hence fixed with the second one, namely:

1. *Board size:* due to a miscommunication when defining the size of the board it happened that the CB was built 2 mm shorter in the two dimensions so it fit properly in the slot rails of the service box.

2. *Crossed lines:* due to a design mistake two of the TTC lines were swapped. It was fixed by rerouting them in the CB-V2.

3. *SPECS bugs:* when testing all the SPECS functionalities in the CB it was found that some of them were either not properly implemented, wrongly documented or they just were not implemented. This caused a major concern in the LHCb collaboration since the SPECS interface is used LHCb-wide. This issue eventually triggered the organization of a SPECS electronics review in which I participated as one of the reviewers. Among the main bugs and miss-implementations were: the TTC channel B decoder; the direction of the SPECS signals for daisy-chaining several SPECS slaves in the same bus; the use of ferrites in high magnetic field environments; the shield to ground connection in the SPECS master and slave boards; the non radiation tolerant 40 MHz internal clock generator; some I^2C communication limitations in relation to the working voltage and maximum frame length; signal integrity issues with the 40 MHz clock; the internal DCU gain; triple voting in the registers implementation in order to provide radiation robustness against single event upsets. Most of the firmware bugs in the slave FPGA were fixed before the mentioned review so the reviewers could assess their correct functioning; the hardware issues were solved in the next version of the master and slave boards.

4. *Clock distribution problem:* one of the most serious bugs in the CB-V1 was a signal integrity issue with the 40 MHz clock signal distribution. The figures below depict very nicely the problem. The 40 MHz clock generated by the TTCrq needs to be distributed to FE electronics (DB and Beetles) and also to the SPECS slave so it can decode the Channel B commands coming from the TTCrx. The line carrying the clock signal was impedance controlled and with impedance matching at the source since the TTCrx output driver is not able to source more than 10 mA. It had a length of about 15 cm in order to reach the 2 SPECS slaves in the CB. In addition a LVDS driver placed close to the source re-transmitted the signal to the FE. We found that the signal at the source, where the LVDS driver is placed, had the shape as plotted in Figure 12.9 when the SPECS slaves were not plugged in the CB. The signal already looks unacceptable enough, but it becomes worse when one of the

SPECS is plugged in (Figure 12.10), due to the fact that the track is not impedance controlled in the SPECS slave causing and impedance mismatch. I decided to shorten the line by cutting it off to 10 cm and feeding only one of the slaves with the 40 MHZ clock and the results measuring at the source without and with SPECS slave plugged in are shown in the Figure 12.11 and Figure 12.12. The plot in Figure 12.13 shows the nice shape of the 40 MHz clock at the LVDS transmitter placed close the TTCrq after redesigning the clock transmission tree. At the SPECS ends the signal had the same shape.
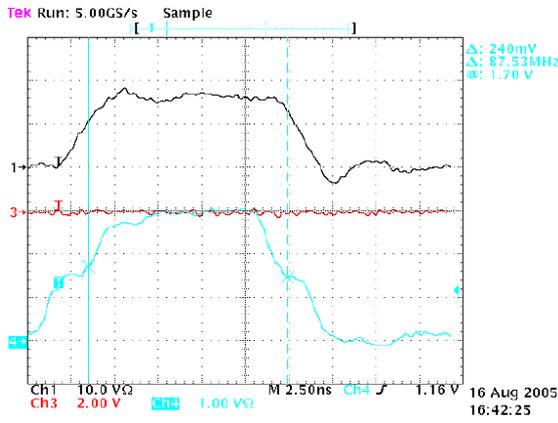
Figure 12.9: Original 40 MHz clock (black) and at the TTCrq output (blue) without SPECS slaves plugged in.
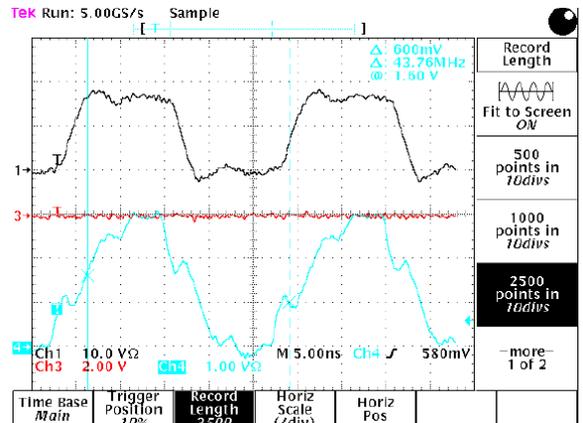
Figure 12.10: Original 40 MHz clock (black) and at the TTCrq output (blue) with one SPECS slaves plugged in.
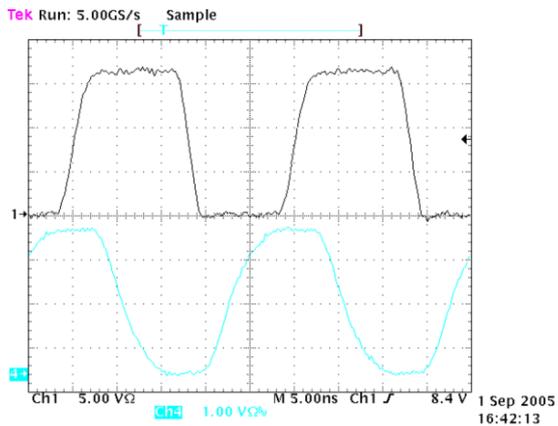
Figure 12.11: Original 40 MHz clock (black) and at the TTCrq output (blue) without SPECS slaves plugged in and line shorten to 10 cm.
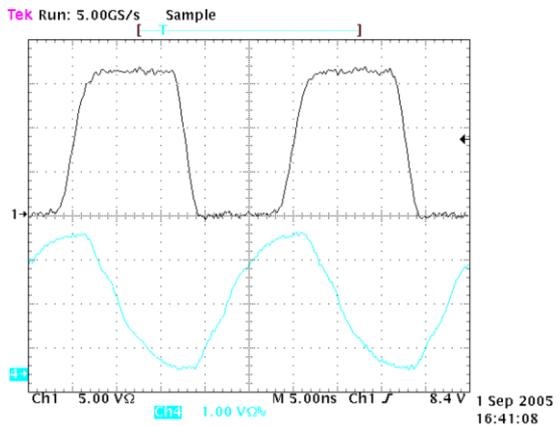
Figure 12.12: Original 40 MHz clock (black) and at the TTCrq output (blue) with one SPECS slaves plugged in and line shorten to 10 cm.
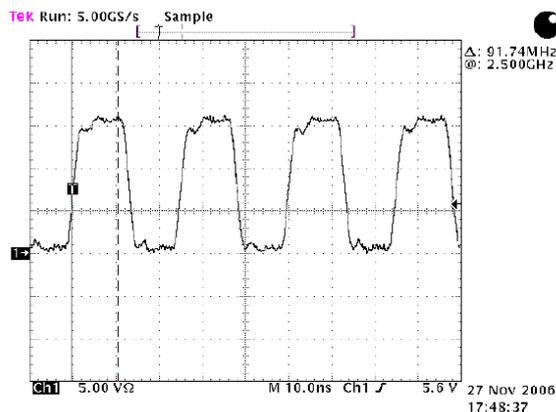
Figure 12.13: The 40 MHz clock at the LVDS input after redesigning the clock distribution.

5. *The PT1000 sensor signals conditioning circuitry:* in the first design of the CB the biasing of the PT1000 temperature sensors that had to be monitored via CB was being done with the 10 and 20 μA current sources implemented in the DCUs. This would generate very low signals, in the range of 10 and 20 mV respectively. As a consequence a signal conditioning stage with gains around 100 and a very low noise figure was needed. In addition, these current sources were found to be not calibrated. This first design never worked out since a redesign of this stage was a better solution. A wheatstone bridge for biasing the sensor was used in the second version of the CB, solution that also needed lower gain in the amplifying stage and showed a good performance.

6. *Beetles I^2C:* due to nature of the I^2C signals (open drain) and the huge length of the lines (4 lines of up to 9 m length in parallel hanging from the same CB I^2C I/O) from the CB to the Beetle chips, we discovered that the effective line capacitance was rather high to be driven with 3K9 Ω pull-up resistors. The rise time of the I^2C signal was to slow to reach the 2.5 V level required by the Beetles (see Figure 12.14). In addition, over and under-shoot was observed in the signals falling edge. The raise time problem was solved by reducing the values of the pull-up resistors to 1K2 Ω and pulling then against to 3.3 V (see Figure 12.15). It is also important to mention that the latest version of the Beetle chip implemented a schmitt-trigger comparator compatible with 5 V for the I^2C I/O. This feature permitted using I^2C signals with low ramping speeds and coping with the small signal oscillations due to transmission line reflections.
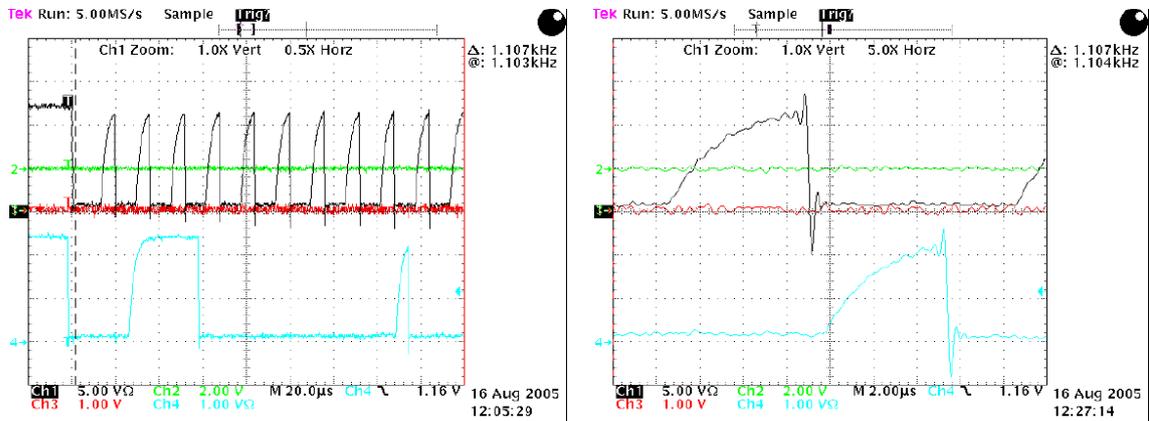
Figure 12.14: The top figure left shows the SCL (black) and SDA (blue) I^2C signals with the 3K9 resistor. The SCL signale hardly reaches the 2 V level. The picture on the right is a zoom of the signals on the left.
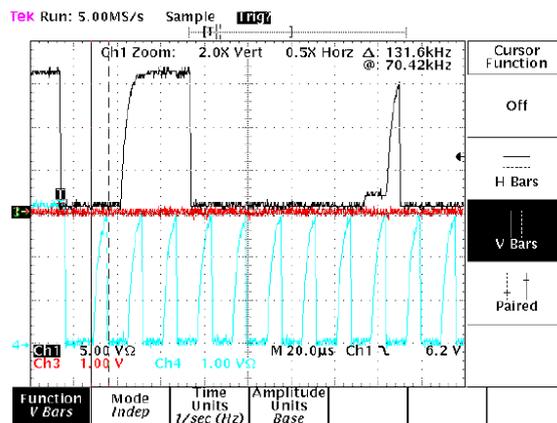
Figure 12.15: The SDA (black) and SDA (blue) signals with 4x6 m cables plugged in and 1K1 resitor pulled-up to 3 V. The step just before the second pulse in the SDA signal is due to the Beetle I^2C acknowledge, i.e. the signal is pulled to ground so the Beetle sinking 3 mA, the maximum limit defined in the I^2C standard and thus the I/O CMOS transistor going into saturation. That's the reason why we decided to use a 1K3 pull-up.

In the version 2 of the CB no design errors were identified, but some changes were performed in order to improve some features and provide extra capabilities:

1. *Layout and component placement:* some modifications were done in order to improve the signal and power routing.

2. *LVDS receivers:* in order to allow for a higher CMRR (common mode rejection ratio) between the SPECS master, located in the counting room 60 m away from the slave in the CB, and also between CBs in different service boxes, the former LVDS receiver (DS90LV047) was replaced with a new one (SN65LVDS32B) with higher CMRR and also tested for radiation tolerance.

3. New additional frontal connector: at some point before launching the production of the final version of the CB it was decided that it would be convenient that the CB featured a small frontal connector providing several input lines for digital signals and also power and ground lines. This would be used to read the detector signals which indicate if the IT detector boxes are in contact with the LHC beam pipe. Due to several reasons, the use of this connector for that end was discarded; however it became very convenient to plug one of the end of the termination cable that had to be installed due to a problem related to the new LVDS receiver as explained in 12.1.5.

4. SPECS add disabling of test-pulses: at this stage it was requested to the SPECS firmware designer to implement the capability of disabling the L0 reset and calibration output signals of the SPECS channel B decoder in order to locally control the their transmission to the FE electronics.

The version 3 of the CB is the one finally installed in the experiment, but it was not absent of some troubleshooting that did not have to do with the board design but with other factors:

1. *Assembly mistakes in CB-V3:* the assembly of the components by the company was not properly done in the first three pre-production boards. They made many mistakes such as assembling components with the wrong size, some components were not assembled as they didn't buy the right ones, they mistook the units of the components positioning files (they considered millimeters instead of mils), they built the solder paste deposition mask wrong (see next slide) and some resistors values were wrong. In order to fix the problem I decided to pass them a completely assembled board so they could use it as a guide.

2. *The glitch in the LVDS receiver for the daisy-chaining:* it was found late during the testing of the Ctrl Boards that the SN65LVDS32B LVDS receiver misbehaved under some circumstances. A glitch in one of the communication lines from the SPECS slave to the master appears just after the end of a transmission going from the 2nd slave in the CB (Figure 12.16). The glitch blocks the master and a reset is required. The SPECS master can only "see" the glitch coming from the 2nd CB in the chain when more than two CBs concatenated since the differential inputs had to be biased. It cannot be seen in coming from the following CBs in the chain since it gets filtered as it passes through all the LVDS drivers/receivers. The reason why we did not see it in the 1st CB is still a mystery, but it was addressed to a timing issue in the SPECS master. After some tests and measurements we realized that a chip failure was the best candidate, and after testing some of the assembled chips in some bare PCBs we found that it was a chip bug. We decided to buy some chips coming from different manufacturing batches and all of them showed the bug but the ones that have as chip inscription LVDS32B instead of LVDS32 as all the other ones. The LVDS32B were by chance the same ones that are assembled in the SPECS slaves and the in principle have been the ones tested for radiation. Replacing the devices was the adopted solution. Anyhow it should be kept in mind a backup solution of "filtering" the glitch in the SPECS master firmware since it is not know how the ageing could affect the device. The manufacturer of the device was contacted about this feature but no answer was received.

3. *The TTCrq modifications:* the TTCrq mezzanine board built at CERN had to be slightly modified so it could be used for the purpose of the ST. The modifications consisted in modifying some resistors and removing the non radiation tolerant regulator housed on it. Not all the TTCrqs were modified properly so they didn't provide the right functionality. This problem was easily identified by visual inspection and fixed manually by putting the correct resistors in the right place.

Once the control boards were produced and assembled three specific tests were performed. The first one consisted of a 24 h. burn-in of the CBs. Six CBs were daisy chained and continuous readings of the inboard devices were performed. In addition, as the CBs were not plugged to a SB and not all the functionalities could be properly tested, the devices and lines were just "pulsed" as if they were in normal operation. After the burn-in we proceeded to calibrate all the DCU channels used for temperature and humidity monitoring and thus checked that the signal conditioning and slow control ADCs worked fine. The last test consisted in checking that all the CB functionalities related to the data acquisition and LV regulators control worked correctly. For that purpose the CB was plugged into a final service box together with a test detector box with 14 hybrids in. The following tests were performed:

1. Test all the slow control issues in a fully equipped Svce Box:
   a. Regulators inhibit signals.
   b. Regulators over-current signals.
   c. Check all the I^2C lines by programming all the Beetles and GOLs and DCUs plugged to the service box.
2. Perform a data acquisition with the whole readout chain and sending different TFC signals. The readout data was later analyzed and checked to be fine.
3.
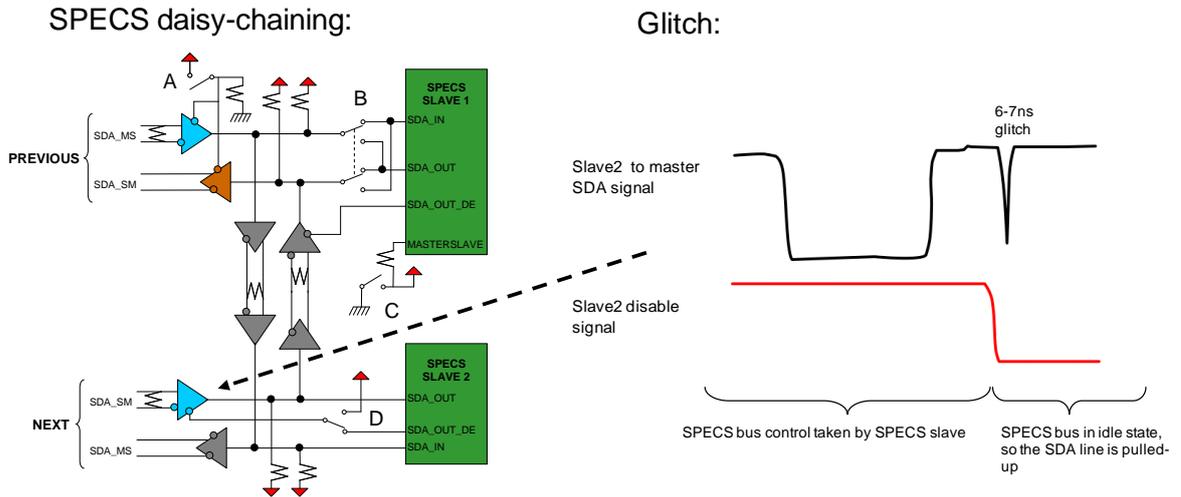
Figure 12.16: On the left is shown the schematic view of CB SPECS daisy-chaining circuitry implemented in the CB with the SN65LVDS32B LVDS receiver colored blue. On right is depicted the glitch generated by the LVDS receiver when it passes from active to idle state.

A failure in any CB would be spotted by any the previously mentioned tests, as it was the case for several boards. The Figure 12.17 is a photo of a batch of fully operational CBs lying in a table on the lab after passing all the tests.

Figure 12.17: Production batch of CBs.

12.1.4 NOISE HUNTING

This section is devoted to one of the problems that kept us struggling for some months during the IT modules production and DAQ electronics commissioning in the lab. Several of us got involved in the troubleshooting of this problem and I would like to acknowledge especially to Helge Voss, the person who dedicated a huge part of his working time to sort the source of the problem out.

In Figure 12.18 we see the S/N distribution for one ladder and the pattern of the common mode subtracted noise for one Beetle port. The measurements were performed using the Beetle test-pulse charge injector, which via an AC coupling injects charge in the sensor strips. The S/N peaking height is in average ~10. The noise has a non flat pattern as we would expect from a pure Gaussian noise distribution for all the channels, leading to channel dependant S/N ratio. This posed a great worry since we were far from the results from the test beam for the same kind of sensors, i.e. a S/N of ~16.

As a first step to understand the problem we studied the impact of the Digitizer Board sampling time in the S/N ratio. Figure 12.19 shows the results of an ADC timing scan for different Beetle channels without common mode subtraction. As the common mode noise was not subtracted the S/N is between 8 and 10, slightly lower that the average 10 we found in the previous figure. Moreover, the timing scan is quite irregular for most of the channels and does not show the proper shape we expected (channel 25 like). We took pedestal data with four different timings as we can see in the right side of the Figure 12.20. The different timings are 10, 30, 50 and 70 units of 104 ps and the noise shows different features for each of the timings. These studies revealed that we were suffering from two different problems, a low signal budget at the ADC sampling stage and a not uniform and somehow time dependant noise pattern.

In order to cope with the signal amplitude issue the gain in the DB board line amplifier was increased only for the IT detector DBs. In the first design of the DBs the gain of this amplifier was set the same for IT and TT, without taking into account the fact that the TT sensors generate a higher amplitude signal because of its thicker silicon sensors and that TT can deal with longer shaping times of the sensor signal, and hence higher signal amplitudes. Thicker sensors imply higher signals at the Beetle channel inputs but also higher noise since the noise figure is dominated by the sensor capacitance. At the same time, the digitization noise is the same for both detectors as we use similar DBs, so an increase on the line amplifier gain for the IT DBs makes sense in order to equalize the lower signals and noise coming from the sensors with the constant noise addition at the digitization stage.
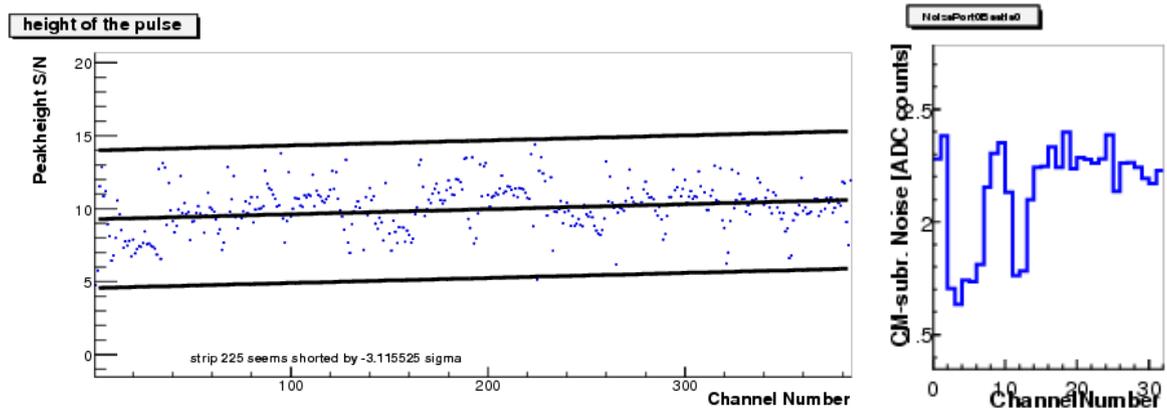
Figure 12.18: Peaking height of test-pulse data for one module (left) and pedestal pattern for one of the Beetle ports (right).

In order to understand the problem of the non flat noise distribution several tests were carried out. Our first candidate for this strange noise pattern was the long data cables going from the hybrid to the DB. These cables have different lengths depending on the detector location in the experimental area, ranging from 2.7 m to 5.4 m in the IT and 9 m in the TT. The plots in Figure 12.21 show the noise pattern for two different cable lengths, which is different for the two different Beetle output ports, but stays fairly equal independently of the cable length. We also tried cutting off the cable in the Beetle end and found that the noise pattern persisted as it can be observed from the Figure 12.22. The same result we got by setting the Beetle parameters such they behaved as if they were off (Figure 12.23). This was a clear indication that the noise pattern is independent of the cable length and the Beetle configuration parameters, so the problem was more likely originating in the DB.

In order to confirm the point that the problem was localized in DB we took off the line receiver input resistors (red framed in the Figure 12.24). The results evidenced patterns similar to the previously showed, so the noise was sneaking in somewhere in the board. Measurements with a low noise differential probe immediately showed in the oscilloscope that the noise was being added between the line amplifier and the ADC stage as can be seen in the Figure 12.26. The source of the noise was not identified but it was discovered that it was a kind a high frequency pickup noise that was slipping in the ADC stage. We found out that the ADC manufacturer spreadsheets during the dates of the design of the board stated that the input bandwidth of the ADC was 100 MHz and time later in a version review of the spreadsheet it changed to 1GHz without advice. The DB was designed such that the place for a capacitor implementing a 1st order low pass filter was foreseen. So the finally assembled DBs were produced with a LPF of 150 MHz built in thanks to adding this capacitor. Figure 12.25 shows the difference in the results with a 1st order and a 2nd order LPF. The order LPF provides a better performance but it would have implied a major modification in the DB so we considered that the solution with 1st order filter was effective enough. We got an improvement in the noise figure of ~25% since we moved from an average of ~1.2 to ~0.95 ADC counts.
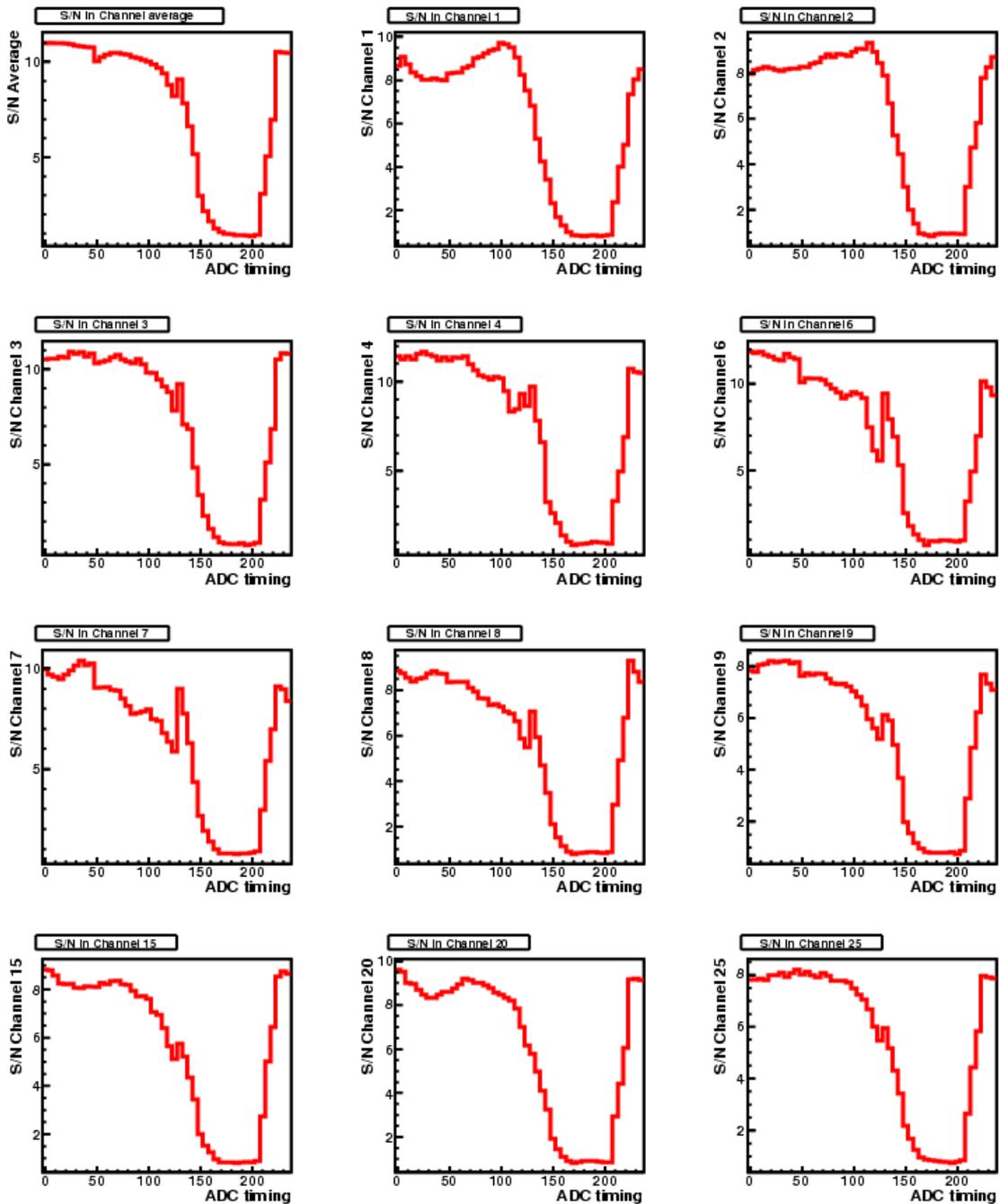
Figure 12.19: ADC timing scan plotting the S/N for different Beetle channels. The top left picture plots the average over all the Beetle channels. The channel 25 (bottom right) shows a shape similar to what should be expected in the absence of external coherent noise disturbances.
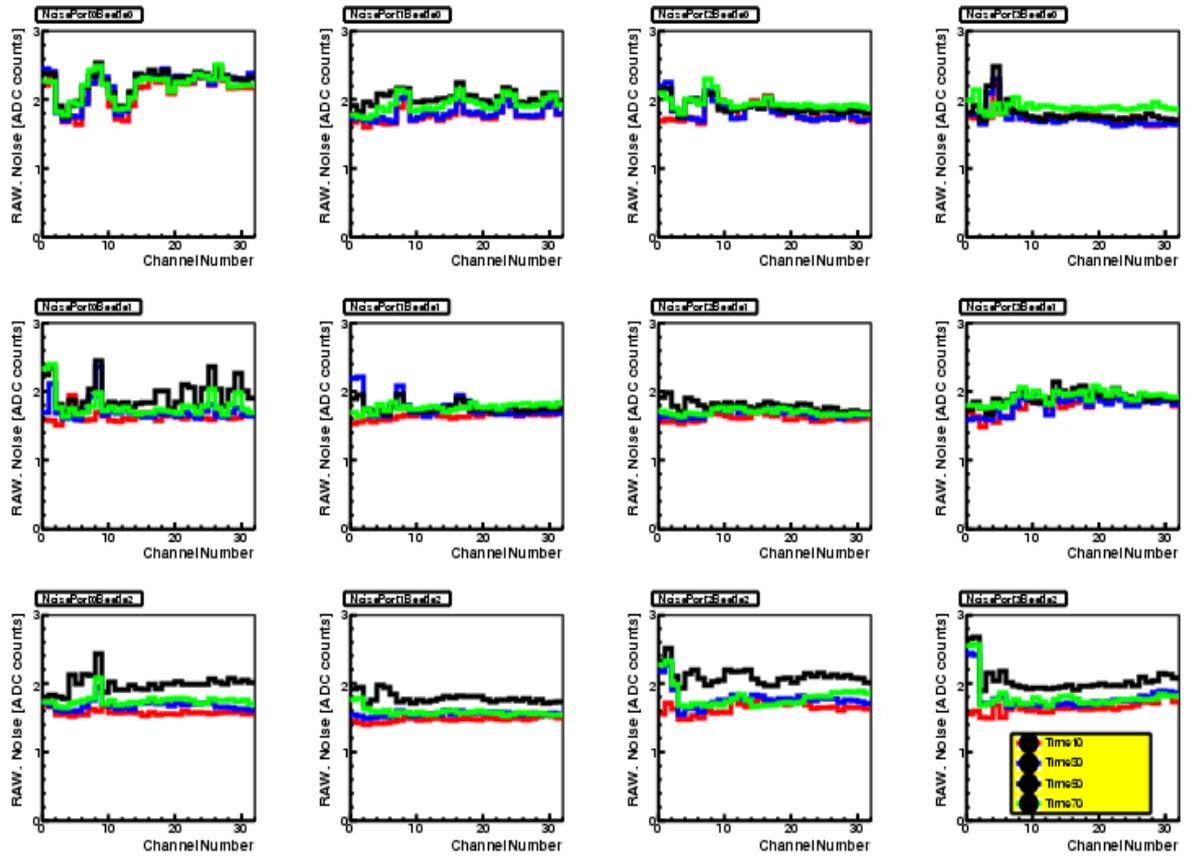
Figure 12.20: Pedestal noise distribution for different ADC sampling times. The different timings correspond to 10, 30, 50 and 70 units of 104 ps, respectively the red, blue, black and green lines.
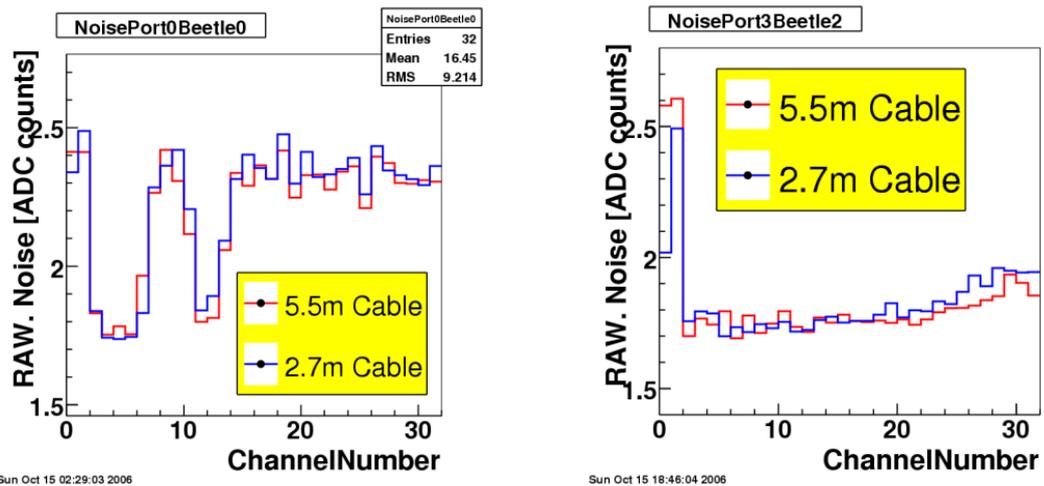
Figure 12.21: Pedestal patterns with two different cable lengths in two different Beetle ports.

The noise improvement together with the new amplifier gain resulted in getting a S/N ratio in agreement with what we would expect from the test beam results. The latter was confirmed by building a small setup for detecting cosmic rays with one IT module. The outcome from this setup was that we could achieve a S/N of 16 with the standard read-out configuration parameters, in perfect accordance with what we would expect.
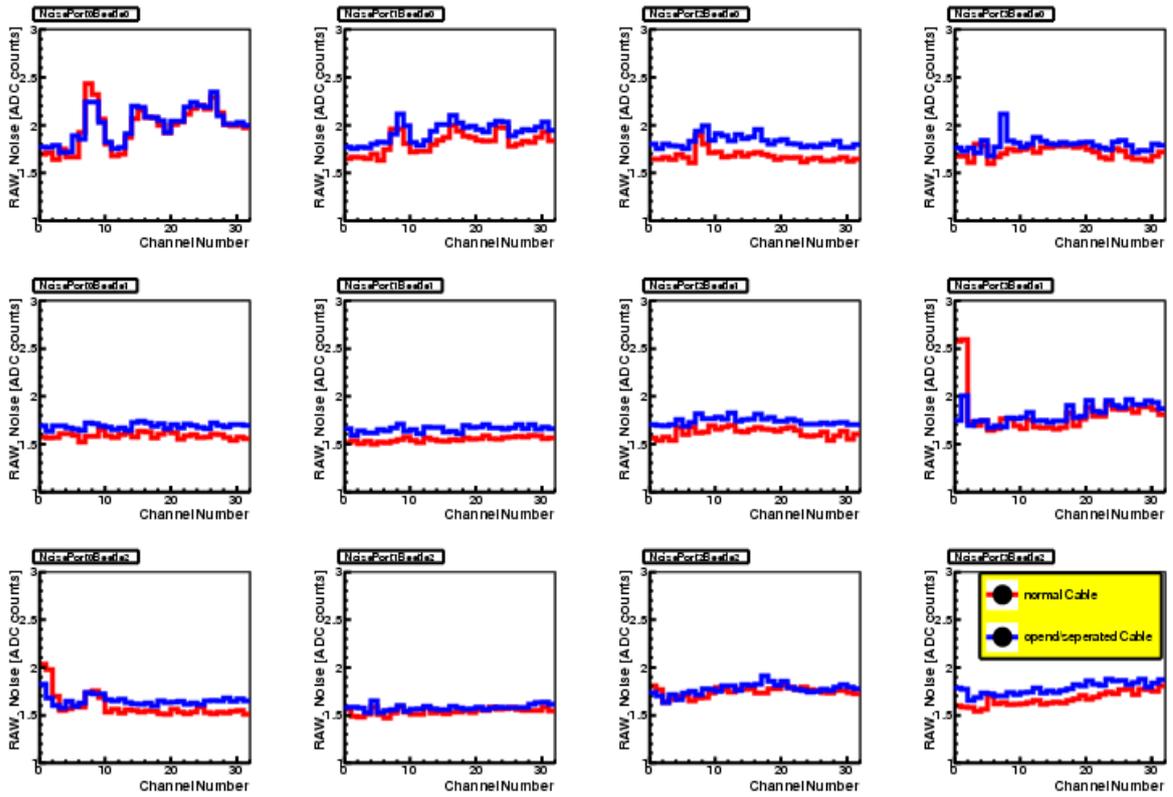
Figure 12.22: Noise pattern with a normal cable (red) and the cable cut off in the Beetle end (blue) for two different cables. Each row of plots corresponds to the four ports of a different Beetle in an IT hybrid.
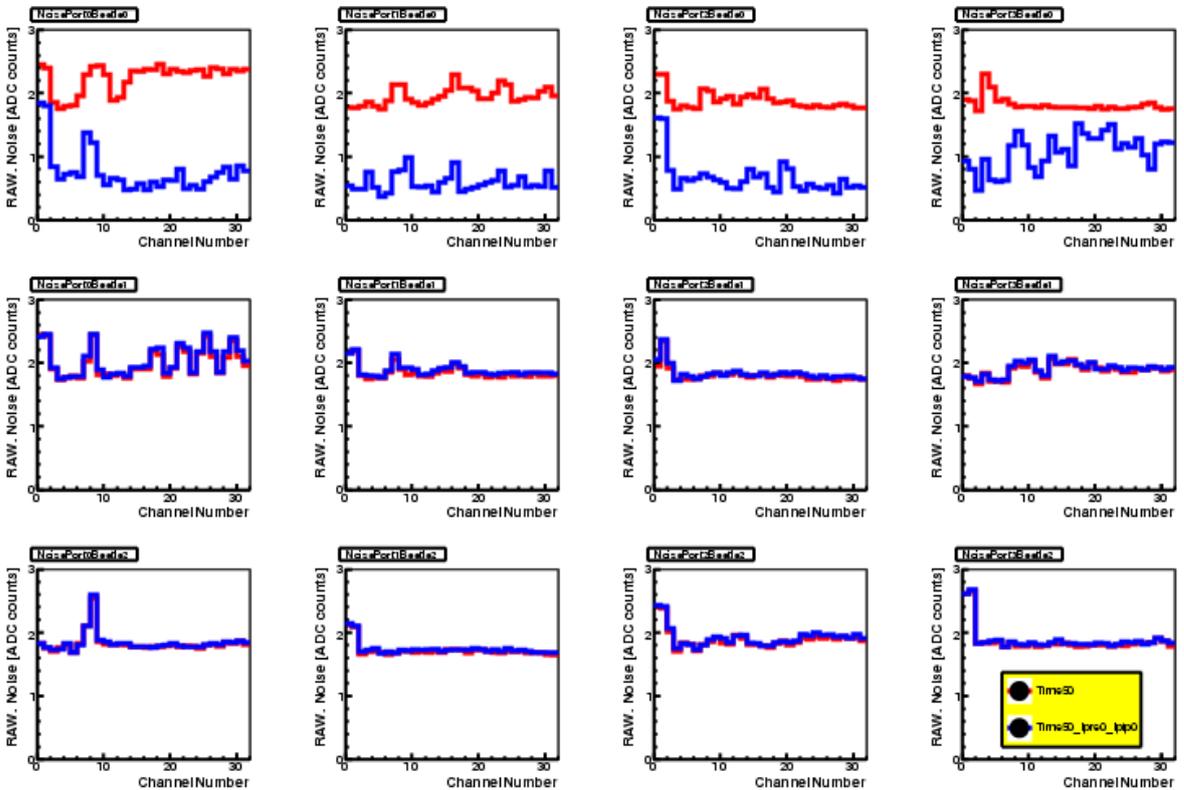
Figure 12.23: Noise pattern with a normal Beetle settings (red) and with different settings for the Beetle in the first row simulating a Beetle off status (red). Each row of plots corresponds to the four ports of a different Beetle of an IT hybrid.
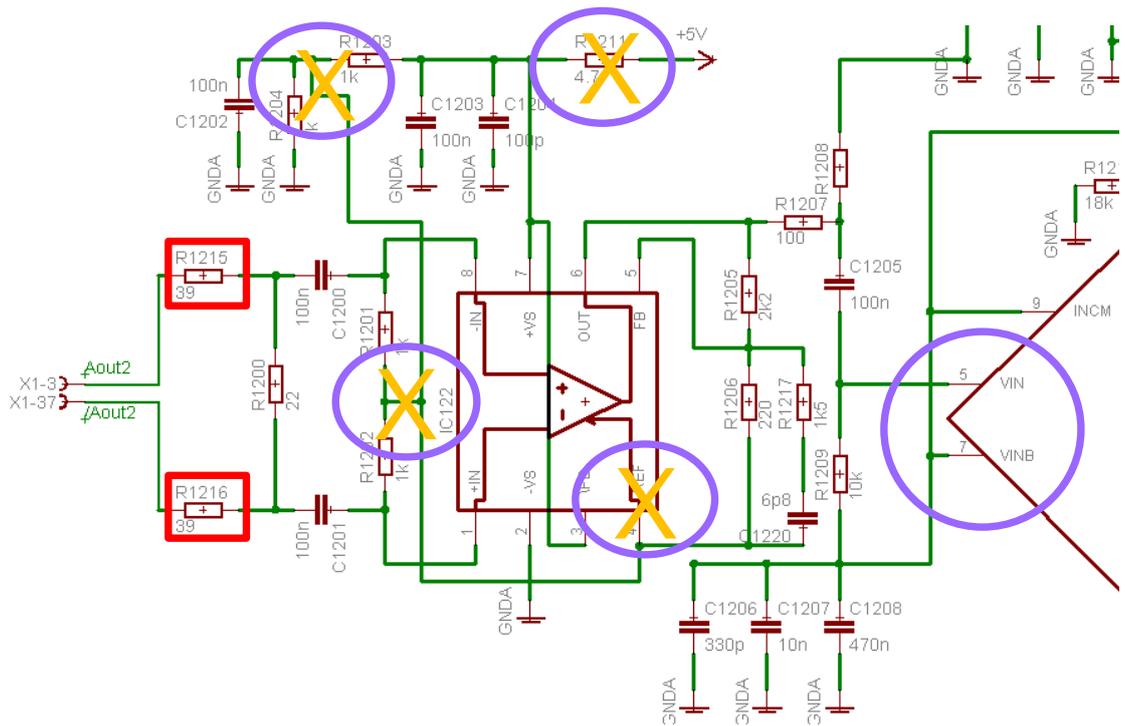
Figure 12.24: Schematic view of the DB amplifying and digitization stage. The blue framed areas with a cross inside indicate that no effect in the noise pattern was found when blocking and separating references of amplifier.
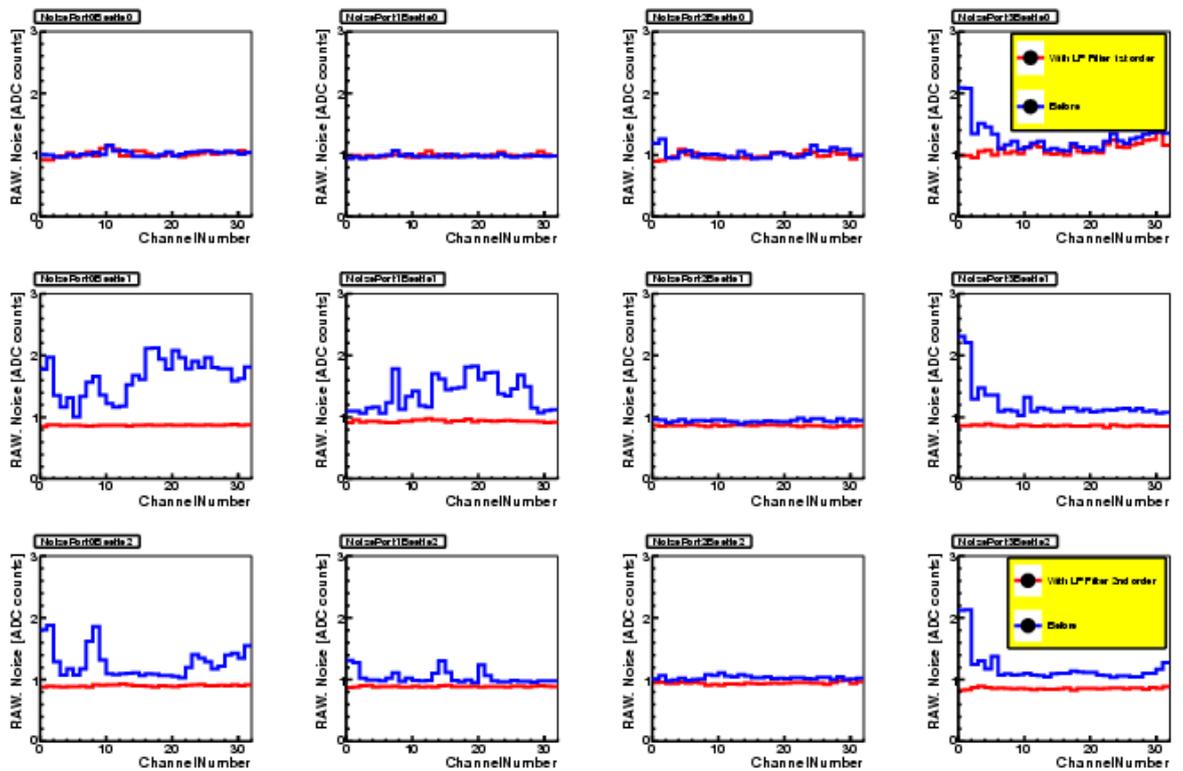
Figure 12.25: Noise figures for different Beetle ports before (blue) and after (red) implementing the LPF. The first row shows the response for a 1st order LPF and the 2nd and 3rd rows for a 2nd order LPF.
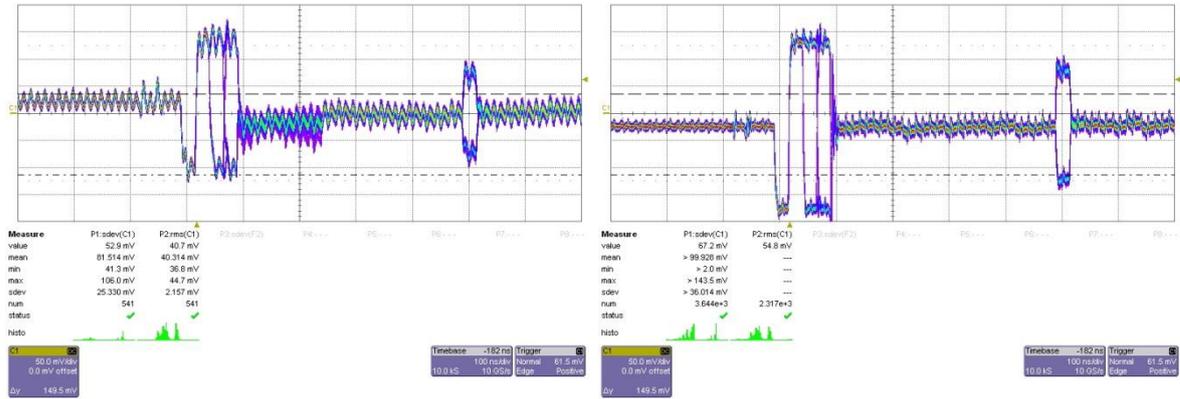
Figure 12.26: Oscilloscope screenshots of the data of one Beetle port at the DB. The first four pulses correspond to the Beetle header and just next comes the channels data. The left figure corresponds to the signal at the ADC input, where it is clearly visible the noisy aspect of the Beetle channels. The figure on the right corresponds to the signal at the input of the line receiver where the ugly noise after the header is not present.

12.1.5 THE CONTROL BOARD COMMISSIONING

The commissioning of the control electronics consisted in several stages. The first tests in the lab with different setups were already described. Hereafter the issues related to the integration in the service box and installation in the pit are explained.

The next stage was the installation of the CBs into the SBs. As the CBs were to be daisy-chained in the same SPECS bus each one needed to have assigned different SPECS addresses. Thus each SB had to be equipped with a CB whose settings cannot match with other one if they belong to the same bus. For that reason the location of each fully equipped service box had to be decided before installing them in the experimental area and the addressing configuration in the SPECS slaves micro-interrupters set before installing in the SB. The service boxes without the CB installed in were pretested in the Zurich production site. Once the SB was fully equipped a dummy hybrid was plugged to each DB and the following tests were performed:

1. Check that the CB communication is fine.
2. Individual powering of all the dummy hybrid and of each DBs partition.
3. Programming of all the GOLs in the DBs.
4. Readout of all the DCUs in the DBs and check that the readings of the channels are consistent.

Several faults were found and easily fixed since most of them were caused by some press-fit pins of the backplane connectors partially damaged while handling the SB. In two cases we had to exchange a DB which did not work properly.

Once all the previous tests were performed the SBs were considered to be ready to install in the experimental area. During this installation some other problems showed up. The most relevant in relation with the control electronics were:

- *Back-plane press-fit connector*. Both CB and DBs press-fit connectors got damaged in some service boxes, which was clearly spotted since some of the SB functionalities or the devices affected by the damaged pins stopped working.
- *Specs bus termination jumpers and cable*. It was found that during the installation of the SBs some CBs had the jumpers which permit cutting the SPECS daisy chain plugged in the wrong position. In addition we found that the LVDS receiver used to implement the daisy chain interconnection worked in an irregular way when leaving the LVDS input open, which is the case for the last board in the chain. Although in the device spreadsheet is stated that in case the pins are open they inputs turn to high impedance mode, this was not the case for some of them. We fixed the problem by building a termination cable which had to be plugged to last CB in the chain and set the LVDS line in a "quiet" state.
- *The CB regulators cooling odyssey*. Although some tests in the lab showed that the heat dissipation of the CB regulators could be performed by air convection installing heat-sinks on the regulators, this result showed to be true for lab conditions, i.e. the CB laying horizontally in an open environment and without any other electronics warming up around. Once the CB was installed in the

SB and set up running at full load new temperature measurements showed that the regulators were being working at a too high temperature. Several air heat-sinks and other solutions were tried out, but none of them showed to be efficient enough to dissipate the heating power other that a water cooling circuit. The design of this system consisted of a cooling block attached to the regulators with the inlet and outlet connected to the previous and next cooling block in the chain with a polyurethane hose. The water cooling fluid provided by the LHCb mixed water circuit present in the experimental cavern. As the LHCb mixed water works over-pressurized we got ~7 bar at the inlet and ~4.5 at the outlet of our circuit. This means that element has to withstand this high pressure, which was not the case of the polyurethane and cooling block joint and ended up in a water leak over the CBs. This joint was improved and tested for each cooling block up to 10 bar. After some months of operation the color of the blue hoses turned from blue to green which made us fear that any chemical reaction could be taking place between the hoses and the anticorrosion additives of the mixed water. Some destructive tests in some pieces of the affected tube dropped that possibility. Nevertheless in the TT a redesign of this CBs water cooling circuit was carried out and the polyurethane hoses were exchanged by copper pipes.

- *Failures of some TTCrq when running at high rate*. The LHCb FE is designed to run at 1.1 MHz event readout rate. The trigger signal is delivered by the TFC system to all detectors via TTC network. The control boards receive the TFC signal and decode them in the TTCrx chip housed in the TTCrq. It was discovered that several of the installed CBs missed triggers only when running at high rates (~1 MHz) from time to time. This had the fatal consequence of a desynchronization of the whole DAQ path. Some of the problems due to a low optical power budget at the TTCrq optical receiver inputs due to bad connections in the TFC optical network. All the connectors were cleaned up and checked. After that, three CBs still persisted to trouble and the diagnosis was that the TTCrx did not only decode wrongly all the incoming triggers but also detected single and double errors when decoding the channel B commands (this channel is encoded using a Hamming linear error-correction algorithm). We decided to replace those three CBs and the problem disappeared, so the TTCrq installed in those boards must be malfunctioning.

- *Specs communication problems*. The IT and TT detectors suffered from some SPECS communication hiccups from time to time. A first problem was found in one of the IT patch panels (PP) that caused signal reflections. The Figure 12.27 shows an oscilloscope screen-shot with the mentioned reflections at with a 160 ns delay. This 160 ns is the round trip of the signal from the CB to the first patch-panel, place around 14 m away. We decided to skip using that cable and plug to a different PP position. The rest of the ST Ethernet CAT5 cables used for the SPECS links were evaluated to be even within the CAT5e specifications (see sample in Figure 12.28) and some short readout tests showed that the bit error rate of the link was well far from 10^{-7}, when putting in operation the whole detector we found that readout errors in the SPECS link were more frequent than expected, at least for some specific parts of the system. The problems with the SPECS communication were investigated from both the hardware and software side. Some fixes and solutions were implemented:

  a. Supply voltage to the SPECS mezzanine. It was reported by the SPECS slave designer that for JTAG accesses the slave is very sensitive to the input voltage and it must be kept within 3.30 ± 0.01 V. The ST does not use the JTAG functionality and the I2C works fine down to 3 V as stated by the designer. In the TT the supply voltage was measured on the frontal connector of the control board, with the full load on the service boxes. The supply voltages are between $3.24-3.29$ V. IT measured similar voltages on a few selected control cards. These values are far from 3 V the limit.

  b. Signal shape measurements. The SPECS designers measured the signal shape in the ST and OT (see Figure 12.29). In the ST there is no pole zero compensation; in the OT the compensation is present. It was clear that the quality was affected due to the missing pole-zero compensation. How much this affect the actual communication performance is difficult to evaluate. What is clear is that lowering the speed would certainly help to improve the signal shape.

  c. SPECS libraries. Following tests with the TT and OT systems, several changes in the SPECS libraries were applied. One of the most important problems was that the functionality to prevent two mezzanines from accessing the same master did not work properly.

  d. 'SpecsServer'. One important improvement was applied to the SpecsServer:

     i. *Automatic recovery*: whenever there is an SPECS communication error, it (soft)resets the master port and tries again. Therefore the retry mechanism is transparent for the (PVSS) and is logged in a file.
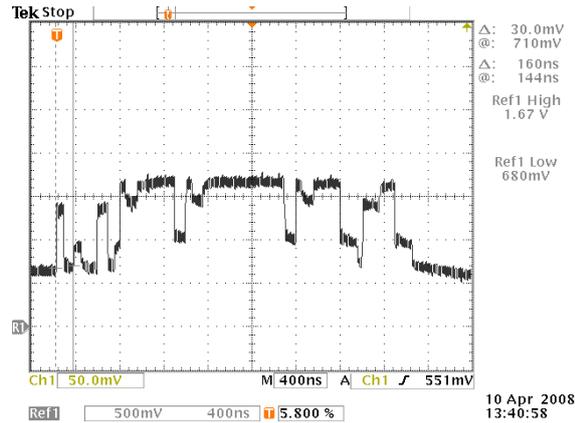
Figure 12.27: One of the polarities of the SPECS SDA differential signal. There are clear reflections around 160 ns after each pulse, the round trip from the CB to the patch-panel placed 14 m away.

Figure 12.28: Sample results of the SPECs Ethernet cables. All of them were even within the CAT5e specifications. The figure in the right is a zoom of that in the left.
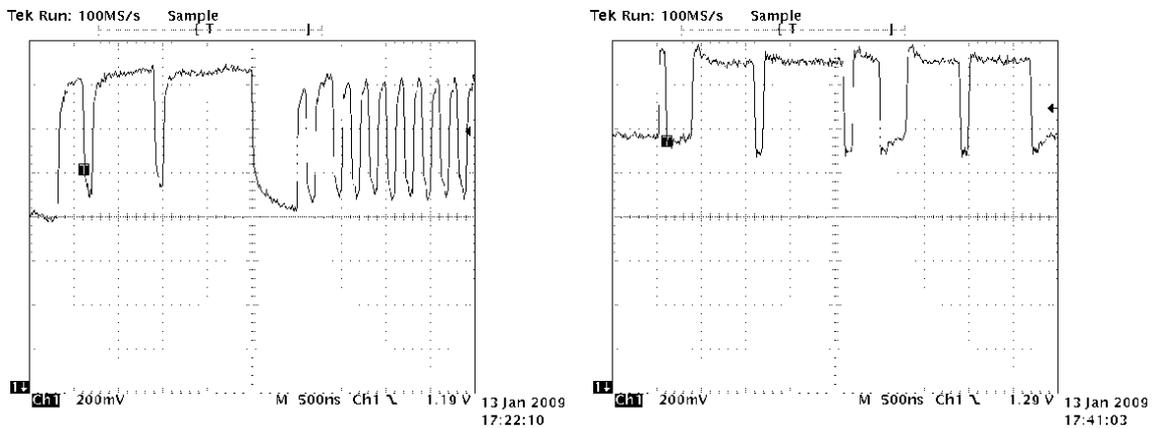
Figure 12.29: Shape of the SDA signal for the IT (left) and OT (right) in the far end for one of the LVDS signal polarities. In the case of the IT the signals rise and fall times are bigger due to the absence of the pole-zero compensation in the CB.

- *LV power oscillations*. During first trials with the maximum trigger rate of 1.1 MHz, several Service Boxes on the Cryo side of the TT subdetector reduced their current consumption by about 10 % and the output data from those SBs was completely desynchronized. The only way to recover the misbehaving SBs was a complete shutdown of the low-voltage supply followed by a restart. This effect was not seen on either the Access side of the TT subdetector nor in the complete IT subdetector. The source of this problem was determined to be large (> 2 Vpp) voltage oscillations on the low voltage supply line between the MARATON power supplies and the SBs. This behaviour was much less pronounced for the Access side of the detector. The difference between both sides and also with the IT is the power cables length, about 30 m of length in the Cryo side and less than half in the Access side and IT detector. Only ceramic capacitors were included on the input line of the LV regulators adding up to about 40 µF per SB backplane. The most likely reason for behavior was a positive feedback on the MARATON regulation stage caused by the equivalent impedance of the SB load. To suppress the oscillations we added 1000 µF electrolytic capacitors externally to each Service Box in order to shift the accumulated phase of the regulation loop away. Various load tests confirmed the stability of the regulation loop in this configuration, which was eventually extended to the complete IT and TT subdetectors. In addition and before installing the capacitors, some quasi-periodic oscillations of 50 s. have also been sporadically observed in the IT despite of not running at 1.1 MHz (Figure 12.30). This periodicity matches with the rate at which all the temperature measurements of the box are performed. Therefore the oscillations may be caused by a sudden power consumption in the monitoring electronics and thus originating from the regulation stage in the MARATON.

- *Beetle derandomizer at high rate*. During the commissioning of the LHCb electronics at high rate a serious problem in the Beetle derandomizer buffer was spotted, which affects the ST and VELO detectors. The buffer has a depth of 16 cells and the Readout Supervisor emulates the buffer occupancy to stop delivering triggers when it is full. It was found out after several tests that the ST detector was missing triggers from time to time when running with 1.1 MHz random triggers and it did not lose anyone when limiting the derandomizer occupancy to lower values. The behavior was exactly the same in the ST and VELO so the problem was clearly identified to be related to the Beetle chip. Some measurements with an oscilloscope carried out in a VELO Beetle brought to light what the problem was. The issue was that the derandomizer was not implemented as a pure FIFO queue and did not release the occupied cells immediately after readout. Figure 12.31 shows a screenshot of the Beetle signals describing the observed behavior. Contacting the Beetle designers confirmed the wrong implementation of the derandomizer: "As you already mentioned under certain circumstances it could take up to 186 clock cycles (4 us) that a locked pipeline cell is set free again for triggering after the readout has started. This implies that in case of a full derandomizing buffer the release for a further trigger acceptance is up to 150 clock cycles or 3.8us (worst case, latency 160, read 4 channels 36 clock cycles !!!). This behaviour was discussed in several meetings since the beginning of the Beetle project in 1999. The conclusion that came out of these meetings was, that as long as "someone" knows when the Beetle does not accept any triggers any more, the trigger master are sending no more triggers. The problem with this is that the calculation of the filling level is quite complex. Therefore the first idea was to implement the Verilog code of the trigger and pipeline management somewhere".
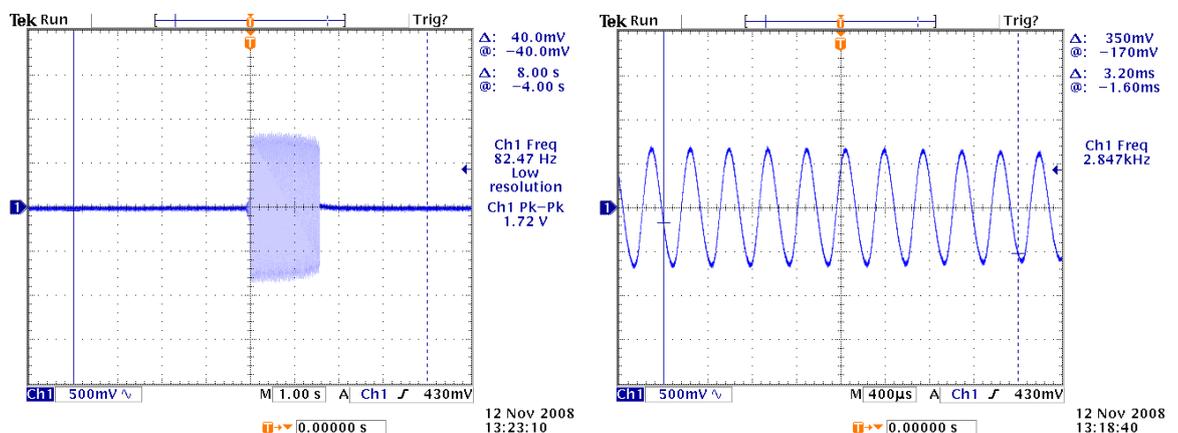
Figure 12.30: Oscillations in the IT power line.

Figure 12.31: The red line is the 'L0Accept', note that there is a bunch of 15 and then twice one trigger. The yellow curve is the 'DataValid' sent by the Beetle, this is the analog data port readout sequence (35 cycles high, one low, maximal speed readout). The green is the 'Derandomizer Fifo Full Flag': Is assigned correctly after 16 triggers but is not removed after the first readout is completed, goes low only after several times 36 clock cycles. The L0Accept during Fifo Full is asserted is lost.

12.1.6 RESULTS FROM DATA TAKING

Next we show some examples of results with real particles obtained with the ST detector after finalizing the installation of IT and TT and overcoming the numerous problems found in the way.

The commissioning of the detector with particles started during the closing stage of installation, by summer 2008. Still some minor problems were remaining in the detectors but were fixed by then. We have used cosmic events, albeit at a low rate due to the horizontal geometry of LHCb, to perform a coarse time alignment of the detector. From 2.55 million acquired cosmic triggers there are several hundred particles that pass through at least one of the IT boxes. 86 of these events contain track segments that cross neighboring stations and two events have a track that could be reconstructed in all three Inner Tracker stations. In Figure 12.32 one of the latter events is shown, in which ten out of the twelve possible hit layers are included in the reconstructed track. One hit is outside of the search window while the last one is missing.

In August 2008, June and October 2009, during the start-up of the LHC, the machine carried out several synchronization tests. In these tests a proton beam of 450 GeV extracted from the SPS (Super Proton Synchrotron) was dumped onto a beam stopper (the 'TED') in the SPS-LHC transfer line, 350 m behind LHCb. Monte Carlo simulations indicate that most of the particles produced in the TED, which reach the ST, are 10 GeV/c muons. During these tests a valuable amount of data was recorded and later used to study the detector S/N performance, time alignment and spatial alignment.

In Figure 12.33 we show the Landau charge distribution for one IT box and in Figure 12.34 a histogram with the S/N ratios of all the IT modules.
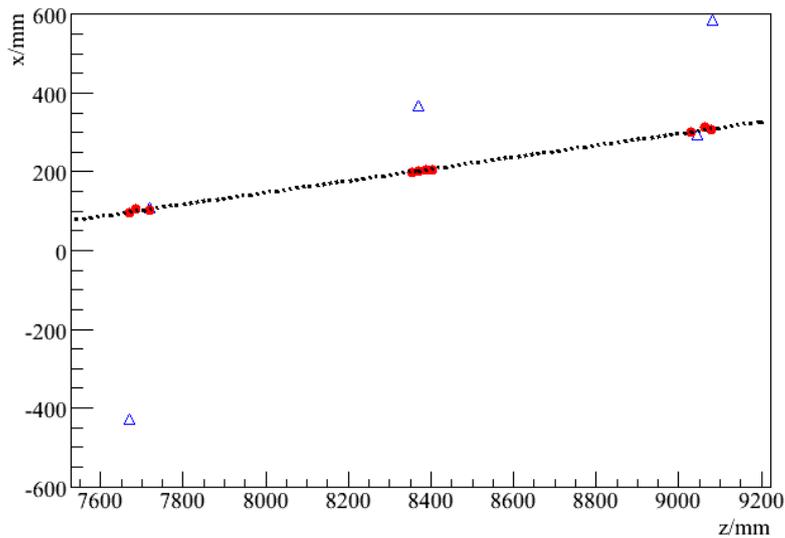
Figure 12.32: Golden cosmic ray through the 3 IT stations.

The sampling of the sensor signals must be done with the right timing, i.e. maximizing the signal amplitude. The timing of the 40 MHz sampling clock can be adjusted per service box with a resolution of 104 ps. The electronics has been designed such the maximum clock skew between service box and Beetles is ±1 ns as this is the tolerance of the clock drivers. In addition the delay of the FE trigger signal can be tuned in 25 ns steps (the 40 MHz clock period) up to 15 cycles.

In order to perform the internal time alignment of the detector runs were taken for both the IT and TT. First of all, the detector had to be coarse time aligned to match the trigger signal with the detector sampling to the right bunch crossing. This was done by reading out 15 consecutive clock cycles and searching for the one with the highest occupancy. After this a fine time alignment of the sampling clock was performed. The procedure consisted in getting data with different delays focusing around the maximum of the signal. The acquired data were averaged for each set of ladders grouped in one front-end service box and a function with the form of a 2nd order CR-RC shaper was used for the fit. The most probable value of the signal determined the optimum sampling point. The delay values for the subsequent data takings were then chosen as the one that maximized the most probable signal amplitude (Figure 12.35). After this procedure the detector is internally time aligned with a precision of ~1 ns.

The Figure 12.36 shows a snapshot of the event display with hits in the TT, IT and OT tracking stations during the collisions.
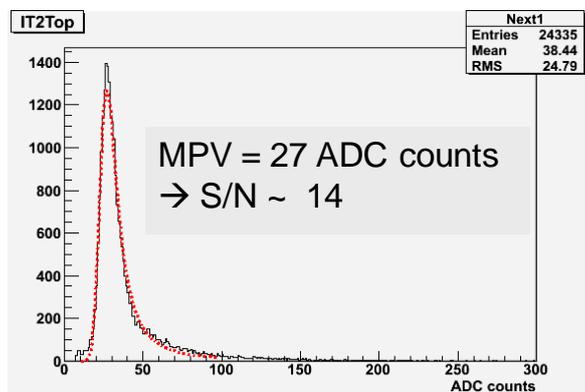
Figure 12.33: Landau charge distribution.
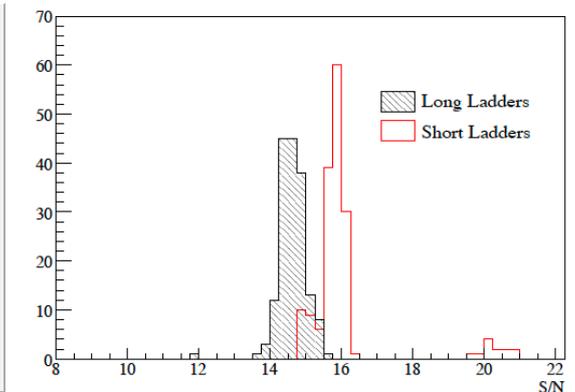
Figure 12.34: Signal to noise ratio (S/N) of the Inner Tracker ladders.

- 263 -
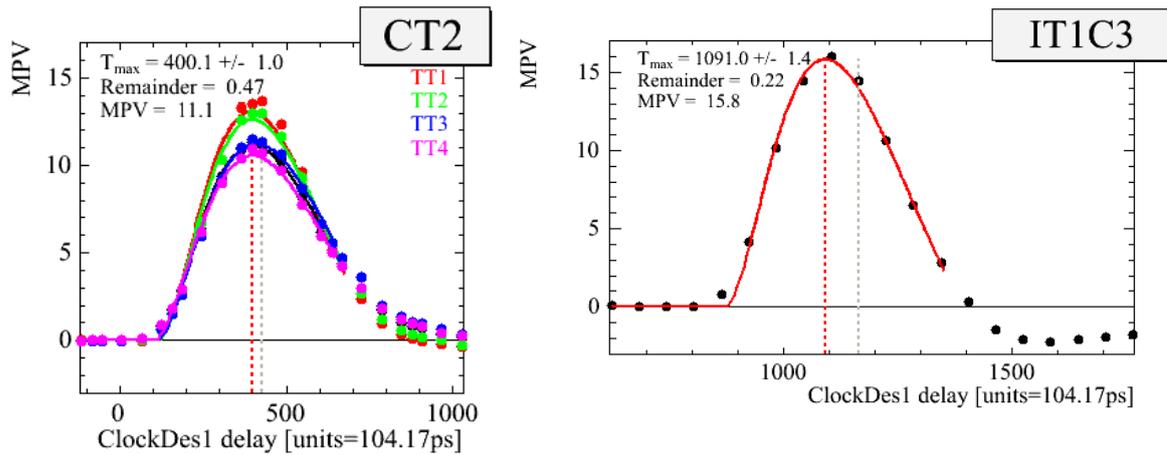
Figure 12.35: The most probable (MPV) of the sampled charge deposition in the silicon for different timing settings obtained in the TED run. The fit is to the expected form of a CR-RC shaper.IT1C3 and CT2 refer to one IT and one TT service boxes respectively. The labels of different curves in CT2 indicate the number of silicon sensors that conform the module, there the different signal shapes.
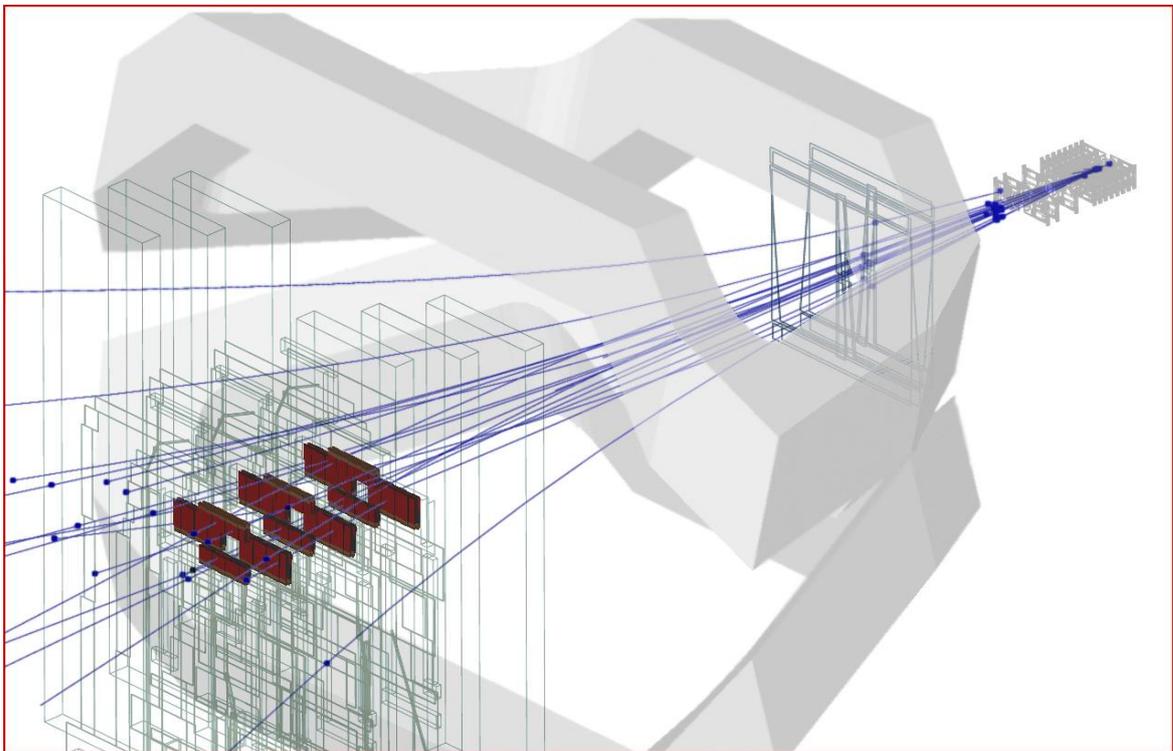
Figure 12.36: Event display showing hits in the tracking stations.

12.2 LESSONS LEARNED

Although several iterations and reviews of all detector elements were carried out, various errors were introduced in the design of the LHCb Silicon Tracker. Some problems were identified in early stages but other only later at the system level, after full assembly of the TT/IT components and integration into LHCb. Some of these were simple design mistakes while others were the result of underestimating the technical complexity of the system. Due to time constraints and lack of manpower no full system test was made in a test beam, which would have covered some of the issues found during system integration in the experimental area. Some other certainly would not have been found with such a test. I believe that there is not any better way to learn than spotting, facing, solving, describing and writing down the problems found and solutions adopted so we do not commit twice the same errors, not only the people who suffered them but also others

who pay attention to this reading. All that has been documented in is this chapter is to my opinion material enough to describe all the lessons which were learned during the construction of the ST detector.

CONCLUSIONS

The work described in this thesis covers many aspects of the development and construction of the Silicon Tracker detector of the LHCb experiment. Although the work of this thesis is focused in the development of the hardware and software for the Silicon Tracker experiment control system, some other tasks in which I was involved in are also reported.

The development of a closely similar control and monitoring system for the IT and TT sub-detectors was a challenging task. The development of a common Control Board is one of the corner-stones of this thesis. In turn, a highly similar hierarchical control system was built for the IT and TT. A large amount of work was dedicated to the design, production, testing and commissioning of the Control Boards. The expected performance was verified with prototype circuits and boards during the development phase in the lab. As the Control Board is exposed to ionizing radiation, radiation qualification was needed for the different electronic devices and hence some irradiation campaigns were carried out either by us or other groups to assess the compliance of the components.

Several test setups were developed to test the Control Board functionalities and the final ST readout electronics. In the very first setup, used to test the CB and perform the burn-in of the IT readout modules, a lot of work was carried out. On one hand, a controller box for the temperature cycling burn-in stand was developed. On the other hand, the assembly and commissioning of the first full readout chain with the almost final detector electronics posed a great effort. Many problems and bugs were encountered in the different hardware elements, which made the task hard to accomplish. In addition, a lot of software had to be coded for the test setups. Two more setups were installed; one to test the finally assembled IT detector boxes and other one to perform the PVSS control system development. In all those tasks I carried a central role, not only in the development of the system, but to some extent in the testing of the final readout elements as well.

It was also part of the work of this thesis the design and development of the LV and HV systems, which are described in the chapters 4 and 5. Different solutions were evaluated and presented to the collaboration, but here we have only described those finally chosen.

The design and implementation of the final control software for the ST is the other task to which I have devoted a lot of the working time. Although the design of the system has essentially been carried out by me, I have received the help of several colleagues during the first design stages and the final system implementation and production. It has to be noted down the challenging achievement that represents having built two different Control Systems, one for IT and one for TT, which present a very similar behaviour. The overall detector is controlled by the Experiment Control System. The ECS is a highly hierarchical, distributed and heterogeneous expert system that controls the detector hardware. Although the ST is not very big in size, it features a significant fraction of the readout channels in LHCb, all of which are under the scope of the ECS. The ST amounts more than ten thousand elements in terms of number of devices to be configured and monitored parameters. The complexity of the system is rather high, based on the Hierarchical Finite State Machines concept which hides to the user the complexity of the underlying hardware elements as well as the different hardware partitioning. It has been a great success being able to built two analogous control systems for IT and TT, which substantially eases the tasks of the operators.

The commissioning of the detector in the experimental area has also been one of the most time consuming tasks. The installation of the detector ended up by summer 2008 and it has extensively been tested and operated since then. The detector has successfully been used to take data during the first injections tests in the LHC machine and also in the first collisions by the end of 2009. The detector is currently running stable and taking data with the ~7 TeV collisions in 2010. Some results that show that the detector is working as expected have also been reported in the last chapter of this thesis.

A full system was implemented: from the basic hardware components to the high-level software. The low voltage and high voltage systems, the Control Board, the Digitizer Boards, the readout hybrids, TELL1s... were developed, assembled all together and linked to the Experiment Control System to build the ST detector of LHCb. The work of this thesis has been mainly focused in the construction of the three first items in the

list and the Experiment Control System, but an important role has also been taken in the testing and development of other parts of the system and their overall integration giving as a final result the construction of a fully functional Silicon Tracker detector.

CONCLUSIÓNS

O traballo descrito nesta tese abrangue diferentes aspectos do desenrolo e construción do detector *Silicon Tracker* do experimento LHCb. Aínda que o traballo aquí exposto se centra no desenrolo do hardware e software para o sistema de control do *Silicon Tracker*, tamén se expoñen outros temas nos que tomei parte.

No comezo, a primeira tarefa desenrolada durante o período de tese foi colaborar e realizar un test de irradiación de compoñentes electrónicos. Deste test concluíuse que os compoñentes que foran seleccionados para a electrónica de DAQ eran resistentes á radiación. Tamén se comprobou que o convertedor analóxico-dixital seleccionado para realizala supervisión de temperaturas e outros parámetros non superou o test para os niveis de radiación esperados no lugar onde se instalaría finalmente a electrónica das *Service Boxes*. Como o problema de encontrar un convertedor analóxico-dixital era un problema común a tódolos experimentos do LHC, o grupo de micro-electrónica do CERN desenrolou un dispositivo 'ad-hoc'. Este test de irradiación permitiu en certo modo empezar a marcalas liñas do que ía ser a electrónica de control lento do detector.

Posto que unha das tarefas principais do grupo era desenrolar o sistema de control do *Silicon Tracker*, decidiuse que sería o grupo de Santiago o encargado de deseñar e producir a electrónica e o software de control. A responsabilidade de levar a cabo estas tarefas recaeu na miña persoa. Como labor previo a desenrolar a electrónica de control do ST, realicei o deseño e a produción da electrónica de control do sistema de ciclos térmicos para os sensores do ST. Este traballo foi de gran proveito de cara a adquirir coñecementos e experiencia para a construción da electrónica de control do ST.

Desenrolar un sistema de control similar para os sub-detectores IT e TT foi unha tarefa esixente. Gran cantidade de horas de traballo foron empregadas en aprender a deseñar e construír tarxetas electrónicas que funcionasen en réxime de alta frecuencia. Isto requiriu buscar e estudar unha gran cantidade de bibliografía así como facer probas con prototipos no laboratorio. A construción dunha tarxeta de control (*Control Board*) común é unha das pezas claves do sistema de control e monitoraxe. É o que permite realizar un sistema de control común tanto para a IT coma a TT. Ademais, dende o punto de vista do software, un sistema de control altamente xerárquico foi construído para o IT e o TT. Moito esforzo foi dedicado ao deseño, produción, probas e posta en funcionamento das *Control Boards*. Comprobouse que cumprían as especificacións de funcionamento con circuítos e tarxetas prototipos durante a fase de desenrolo no laboratorio. Durante o proceso de deseño e probas encontráronse distintos problemas tanto de deseño como de hardware nas distintas partes da electrónica, os cales foron corrixidos antes de seren instalados na area experimental. As *Control Board* están expostas a radiación ionizante, motivo polo cal os distintos dispositivos montados nas *Control Boards* teñen que estar cualificados para os niveis de radiación presentes na área experimental. A busca de dispositivos adecuados tamén requiriu un longo traballo. Por un lado, leváronse a cabo diferentes campañas de irradiación en dispositivos, tanto pola nosa parte como por parte doutros grupos de LHCb. Por outro, buscáronse dispositivos que foran avaliados por outra xente tanto dentro do CERN coma noutros ámbitos coma os da industria aeroespacial.

Construíronse varios sistemas para probar as funcionalidades da *Control Board* así coma a electrónica final do ST. No primeiro dos sistemas que se instalou para testar as *Control Board* e os módulos do IT adicouse moito tempo de traballo. Como xa foi mencionado anteriormente, por unha parte, houbo que facer unha caixa controladora para o sistema de ciclos térmicos. Por outro lado, instalar o primeiro sistema completo de lectura coa electrónica final ou quasi-final supuxo un gran esforzo. Durante o proceso encontráronse diversos problemas e fallos nos distintos compoñentes, o que fixo que a tarefa de integración virase un pouco mais difícil. Adicionalmente, houbo que facer diversas aplicacións software para por os sistemas a funcionar. Dous sistemas mais foron instalados; un para validar as caixas de detección finais do IT e outro para desenrolar o sistema de control PVSS. En todas estas tarefas tiven un papel relevante, non só no desenrolo dos sistemas, se non que incluso nas probas dos dispositivos construídos e coordinando a súa posta en marcha.

Tamén foi parte do traballo desta tese o deseño e construción dos sistemas de LV e HV, que se describen nos capítulos 4 e 5. Estudáronse distintas opcións e presentáronselle á colaboración. Entre as posibles opcións para o LV estaba a de ter as fontes de alimentación fora da zona de radiación hostil do experimento e utilizar cables longos e de alta sección ou a de ter un sistema de alimentación distribuído cos módulos de baixa

potencia cerca do detector e resistente á radiación e a campos magnéticos intensos. Na tese só se describe a que foi finalmente adoptada, que é a segunda. Con respecto ao HV, fixéronse probas con módulos de alta tensión de distintos fabricantes e vendo que en canto as prestacións eran similares, decidimos decantarnos por aquel que ofrecía mellores prestacións en canto ao software de control.

O deseño e implementación do software de control para o ST é outra das labores que requiriu unha dedicación de gran parte do meu tempo. Aínda que fundamentalmente fun eu quen deseñou o sistema, tiven bastante axuda de varios compañeiros de traballo na implementación e produción. De feito, parte deste traballo deu lugar a un proxecto de fin de carreira dunha estudante de enxeñería industrial. Neste traballo avaliáronse as distintas ferramentas de software dispoñibles para implementar o sistema e fixéronse os primeiros prototipos. Hai que resaltar que construír dous sistemas de control, un para o IT e outro para o TT, que presentan comportamento similar foi un logro considerable. O detector completo contrólase a través do Sistema de Control do Experimento (ECS en inglés). O ECS é un sistema experto altamente xerárquico, distribuído e heteroxéneo que controla o detector. A pesar de que o ST non é un detector dun tamaño moi grande, si que constitúe unha fracción importante dos canais de lectura de LHCb (preto duns 300 mil), os cales se controlan todas a través do ECS. O ST posúe mais de dez mil elementos contando tanto o número de dispositivos a configurar coma parámetros a supervisar. O sistema fundaméntase no concepto de maquinas de estados finitas xerárquicas, o cal fai transparente para o usuario a complexidade tanto do hardware que hai debaixo cómo do xeito en que este está agrupado. Foi un éxito construír dous sistemas de control semellantes para o IT e o TT, o cal facilita o traballo dos operadores.

A posta en marcha do detector na zona experimental foi outra das tarefas que implicou unha carga de traballo considerable. A instalación do detector rematou no verán de 2008 e dende entón estívose validando e utilizando de forma intensiva. Empregouse para tomar datos de xeito satisfactorio nas primeiras inxeccións de partículas do LHC e tamén nas primeiras colisións a finais de 2009. O detector actualmente funciona estable e estanse tomando datos con colisións de ~7 TeV en 2010. Algúns dos resultados mostráronse no último capítulo da tese. Durante este tempo entendéronse mellor as feblezas e virtudes do sistema de control e foise facendo un traballo de mellora continuo no sistema. Dende a versión inicial do software ata a que finalmente se está a utilizar hoxe en día houbo unha evolución considerable, tanto en termos de fiabilidade coma de prestacións e facilidade de manexo. Tamén se aprenderon leccións importantes con respecto a temas de hardware. Diversos problemas e puntos débiles do sistema foron identificados unicamente unha vez que o detector foi instalado na area experimental. Houbo dous en concreto relacionados cos temas centrais desta tese. Por un lado está o problema da refrixeración dos reguladores da *Control Board* e por outro cruzamento das liñas de HV.

As *Control Board* foron deseñadas de tal forma que os reguladores para prover as distintas voltaxes para o seu funcionamento están na propia tarxeta en si. A refrixeración destes reguladores era inicialmente con disipadores de calor por convección natural, os cales se viron que baixo condicións de traballo reais eran ineficaces. Estes disipadores foron mudados por uns refrixerados por auga, os cales ofrecen unha capacidade de disipación mais eficiente. O problema destes disipadores foi que tiveron fugas na conexión do bloque de metal coa tubaria de poliuretano debido á alta presión do circuíto de auga. Tras este contratempo foron modificados e o circuíto e actualmente completamente metálico, minimizando así o risco de fugas.

Con respecto ao problema das liñas de HV cruzadas no Inner Tracker, este ten a súa orixe no feito de que cada unha das catro caixas detectoras de cada estación está instalada dun xeito distinto, rotada con respecto ao seu propio centro. Isto implicou que para facer ben os *patch-panels* de HV houbera que ter en conta as rotacións. Sen embargo, debido a unha confusión na maneira de numerar os módulos de cada unha dos planos de detección, na metade deles as liñas de alimentación de HV están cruzadas con outras. O problema foi resolto de forma temporal recableando os *jumpers* no *patch-panel*.

Acadouse construír un sistema completo: dende os elementos hardware fundamentais ata o software de alto nivel. Desenroláronse os sistemas de baixa e alta voltaxe, a *Control Board*, as *Digitizer Boards*, os híbridos, TELL1s… e ensambláronse todos xuntos e conectáronse ao ECS para construír o detector ST de LHCb. O traballo desta tese estivo centrado na construción dos tres primeiros elementos da lista e no sistema de control do experimento, pero tamén tiven un papel relevante nas probas e desenrolo doutras partes do sistema así coma na integración final de tódolos elementos, dando como resultado a construción do detector *Silicon Tracker*.

APPENDIX A   CONTROL BOARD SCHEMATICS
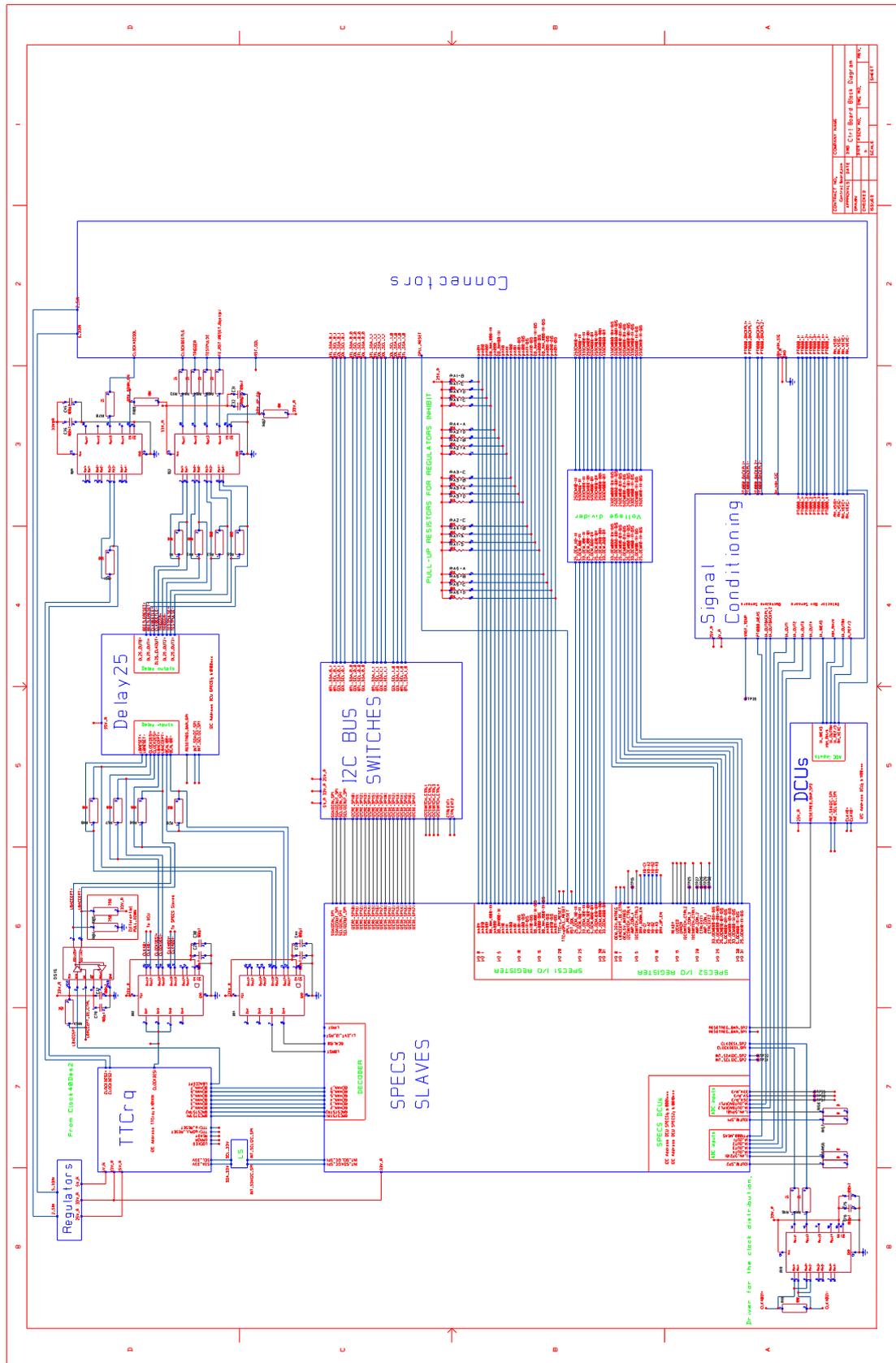
Figure A.1: Control Board block diagram
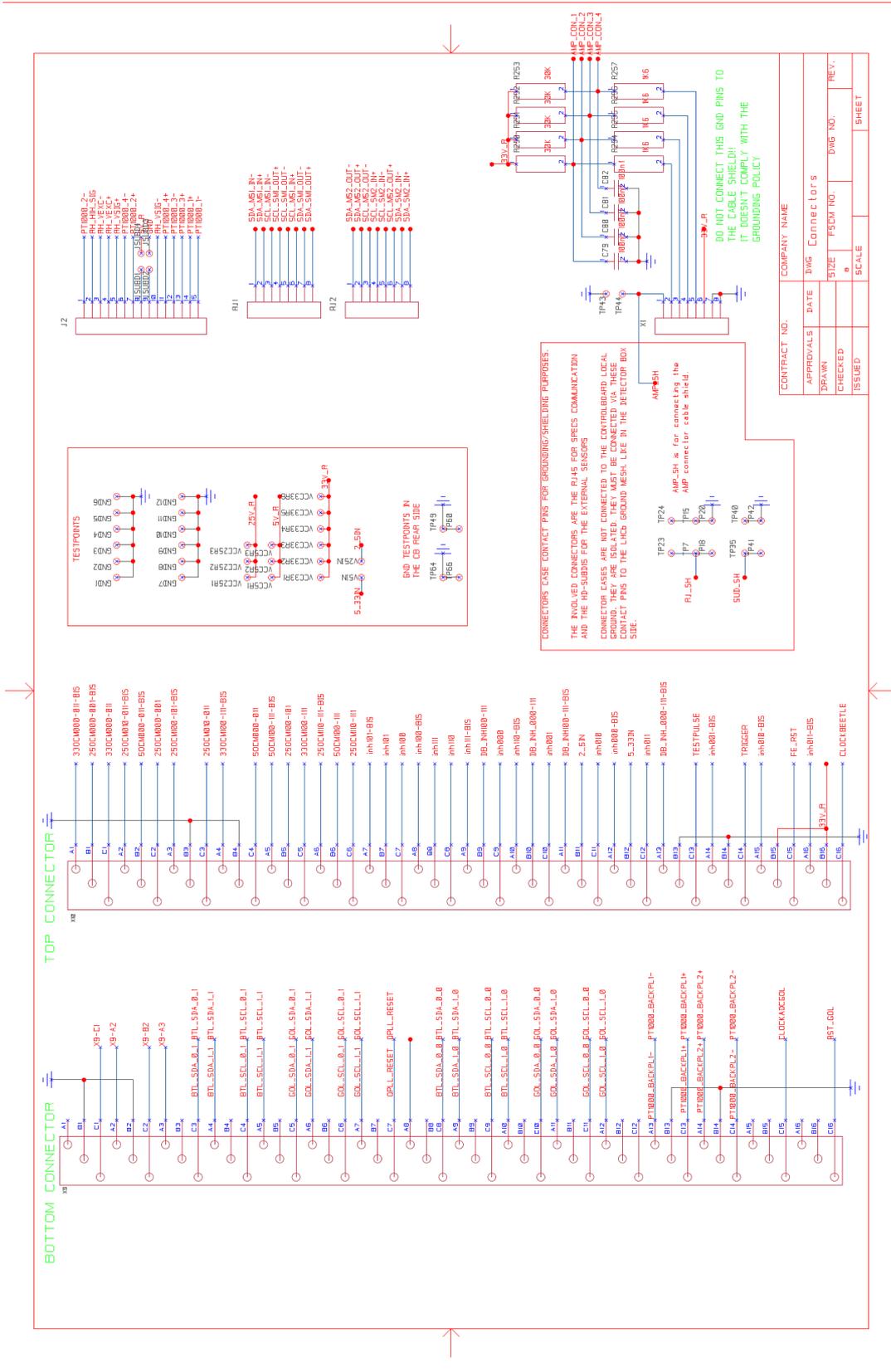
Figure A.2: Connectors
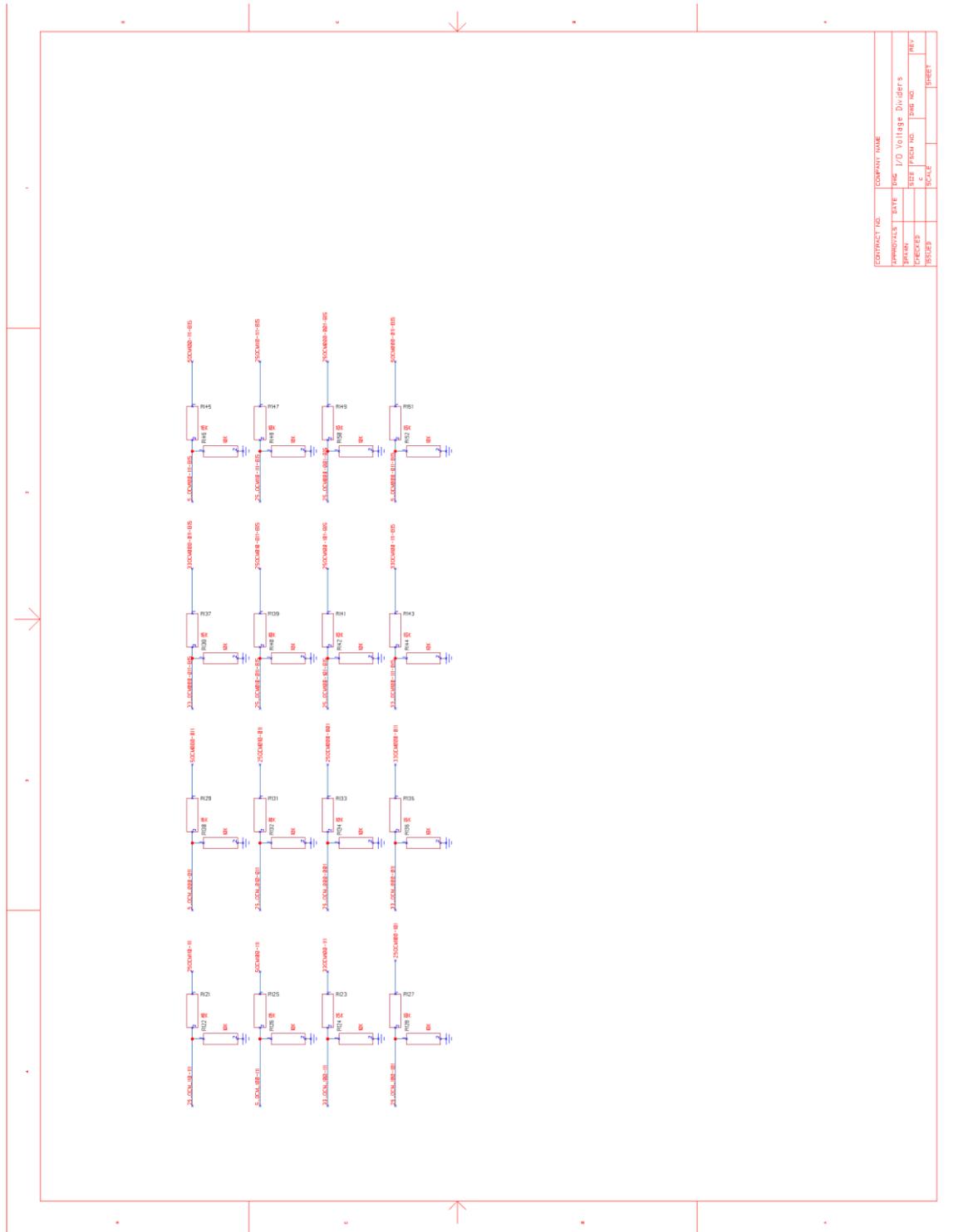
Figure A.3: I/O voltage dividers
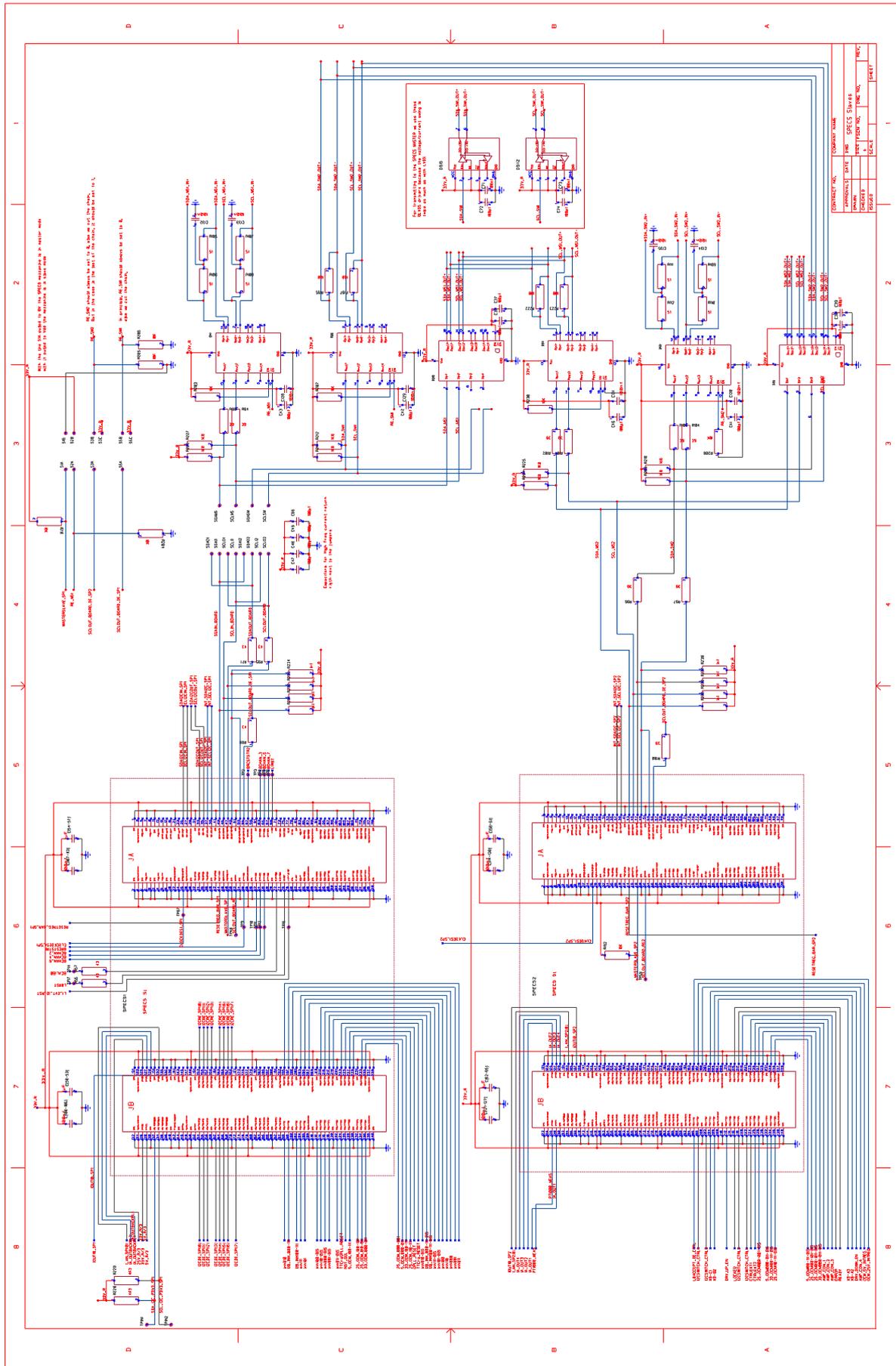
Figure A.4: Specs slaves + SPECS bus chaining

Figure A.5: Delay25 chip
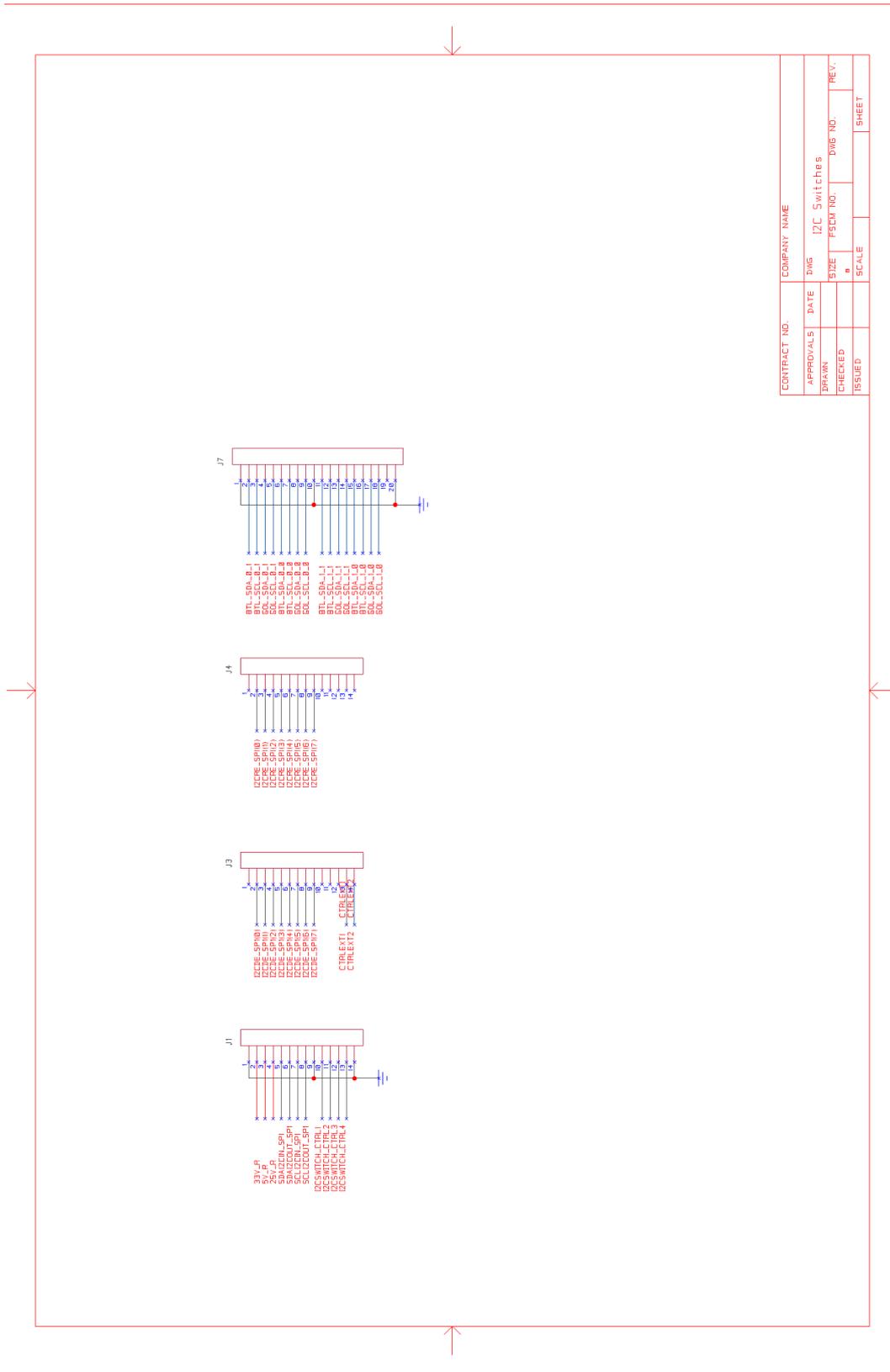
Figure A.6: I^2C mezzanine connectors interfacing

Figure A.7: Level shifter
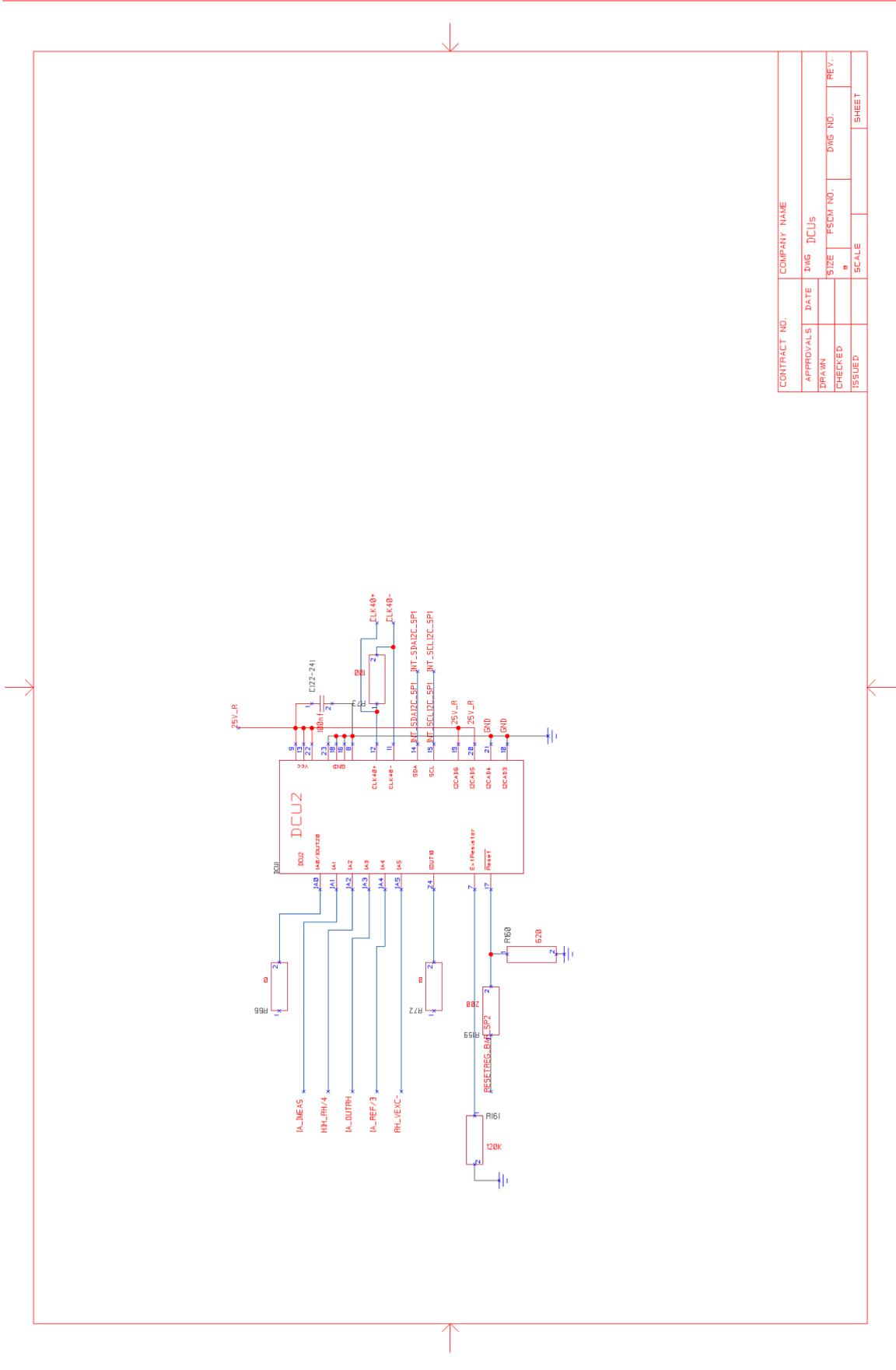
Figure A.8: DCU2

Figure A.9: Signal conditioning
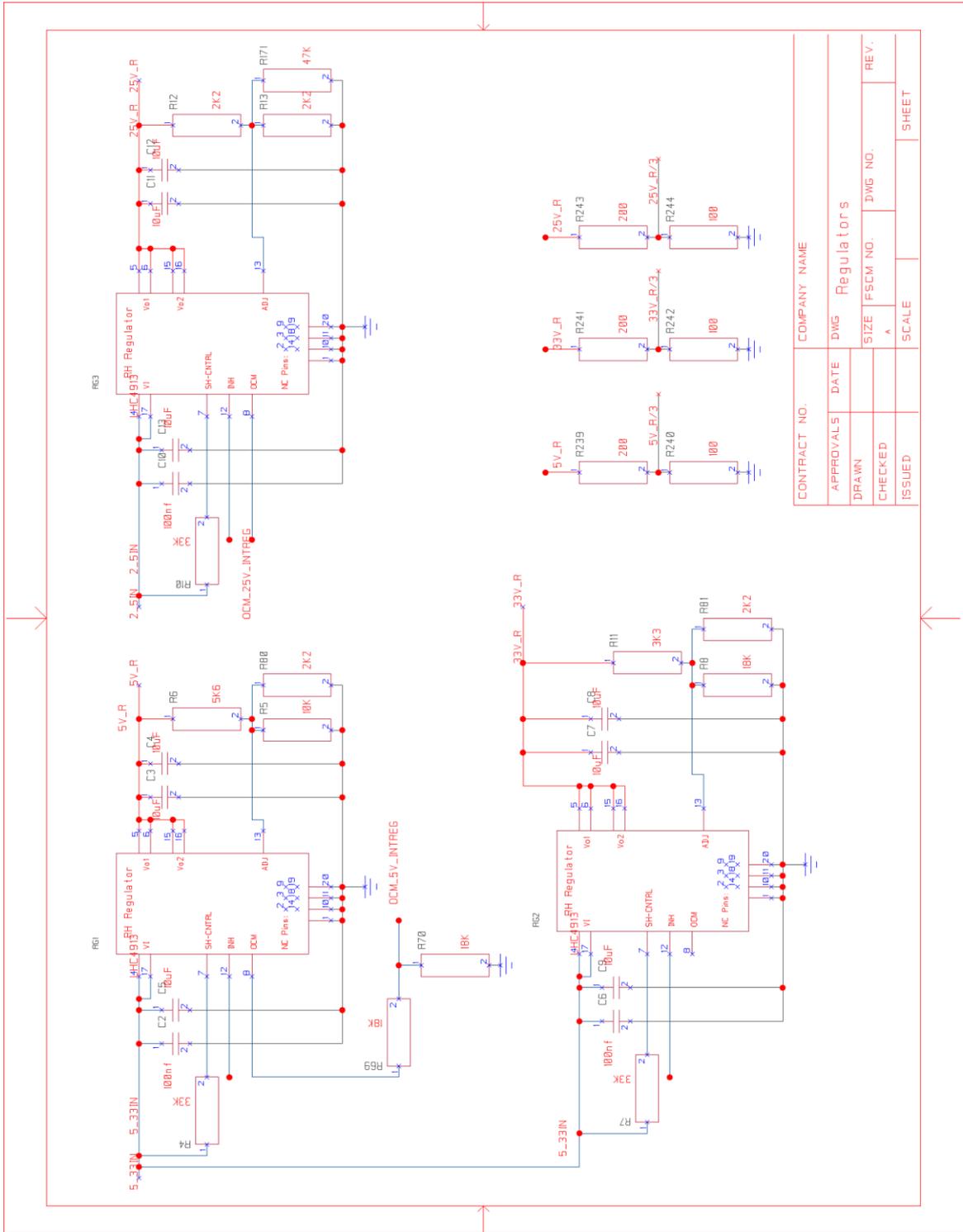
Figure A.10: Voltage regulators
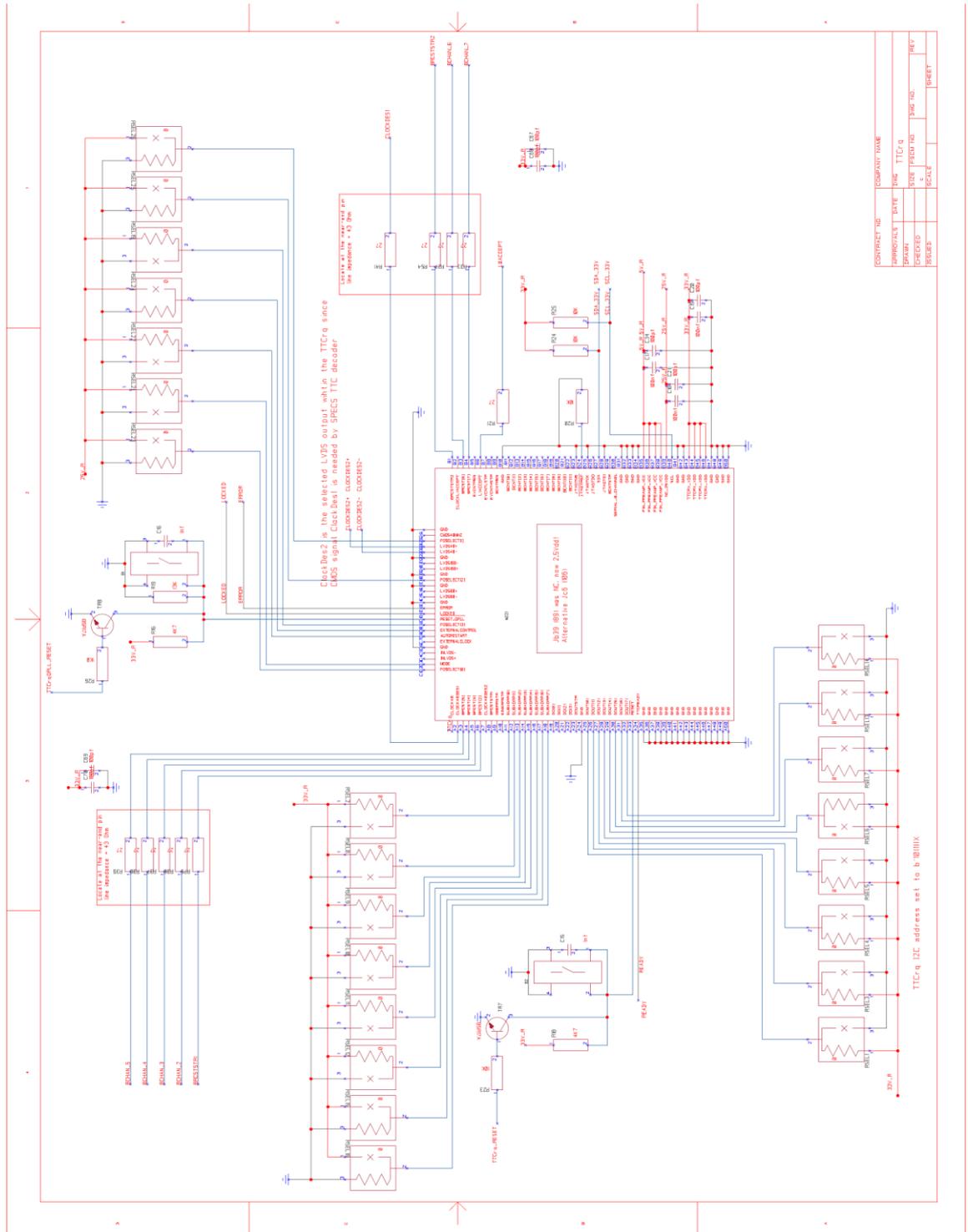
Figure A.11: TTCrq

APPENDIX B   CB CONTROLLED IMPEDANCE LINES
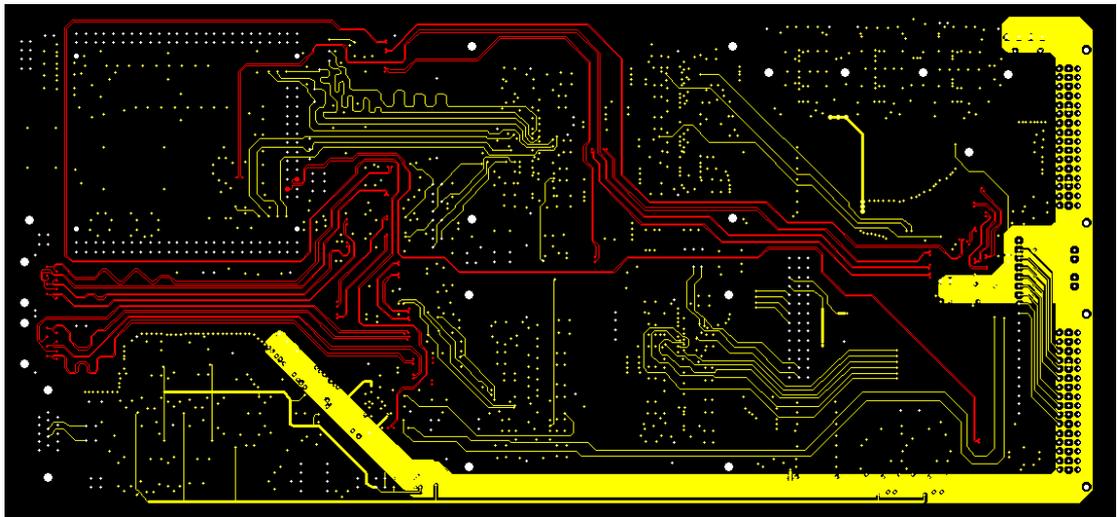
Figure B.1: 100Ω differential impedance tracks. "Signal2" layer
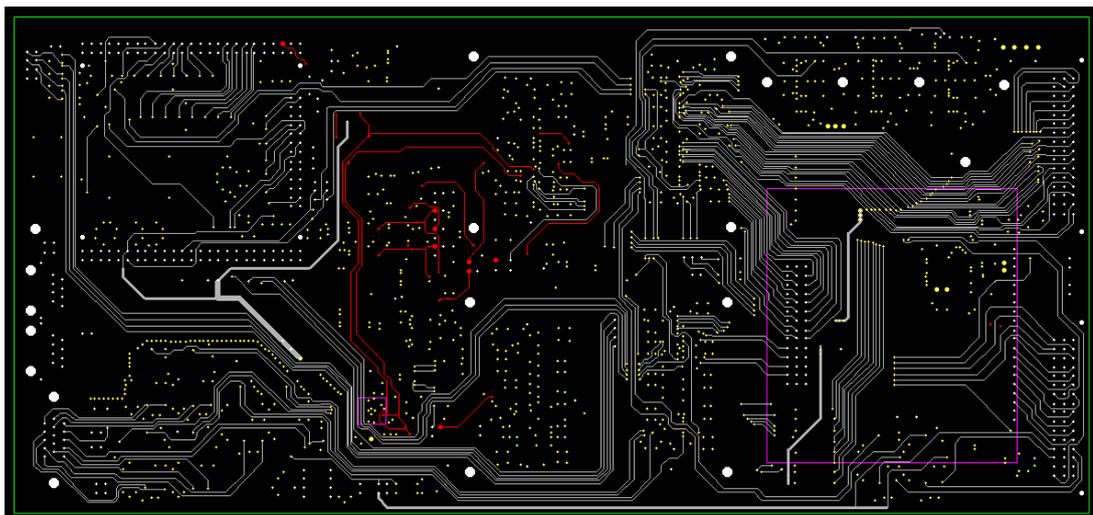
Figure B.2: 60Ω impedance tracks. "Signal1" layer
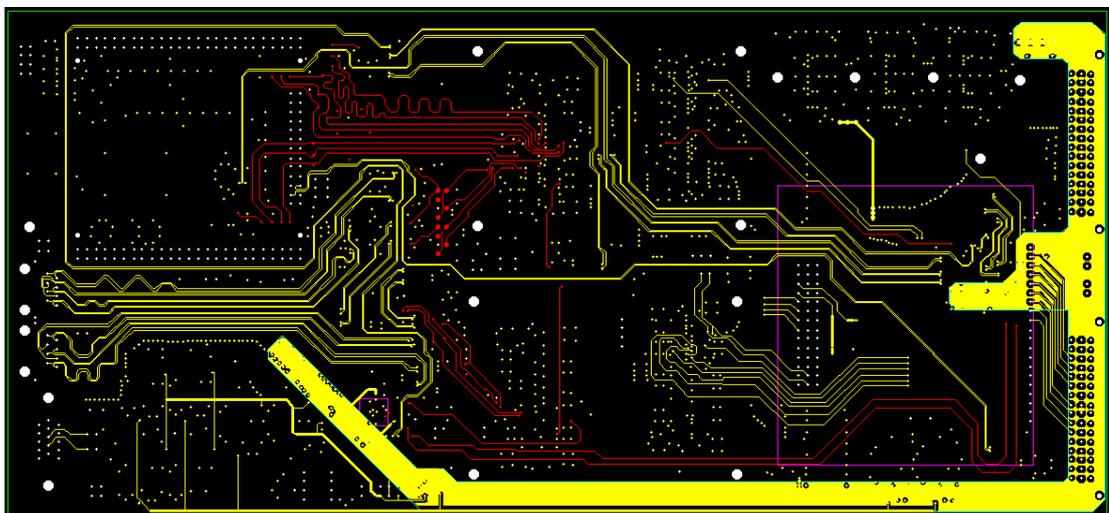
Figure B.3: 60Ω impedance tracks. "Signal2" layer
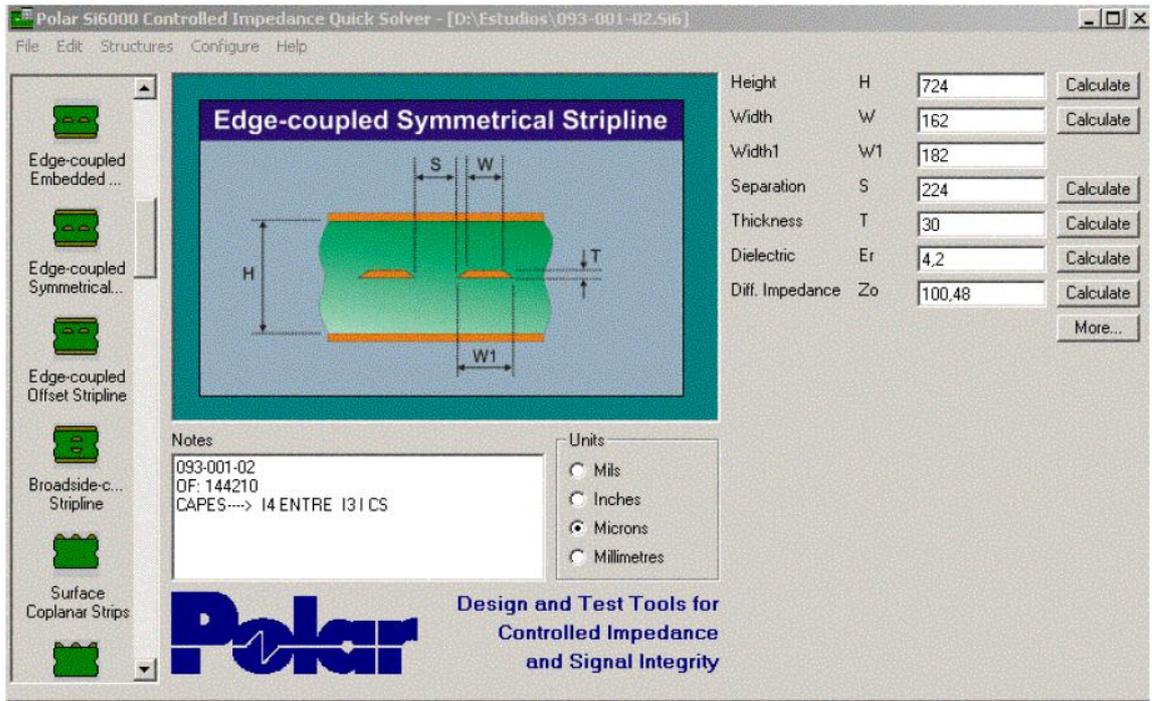
Figure B.4: Manufacturer controlled impedance solver result for 100Ω differential impedance
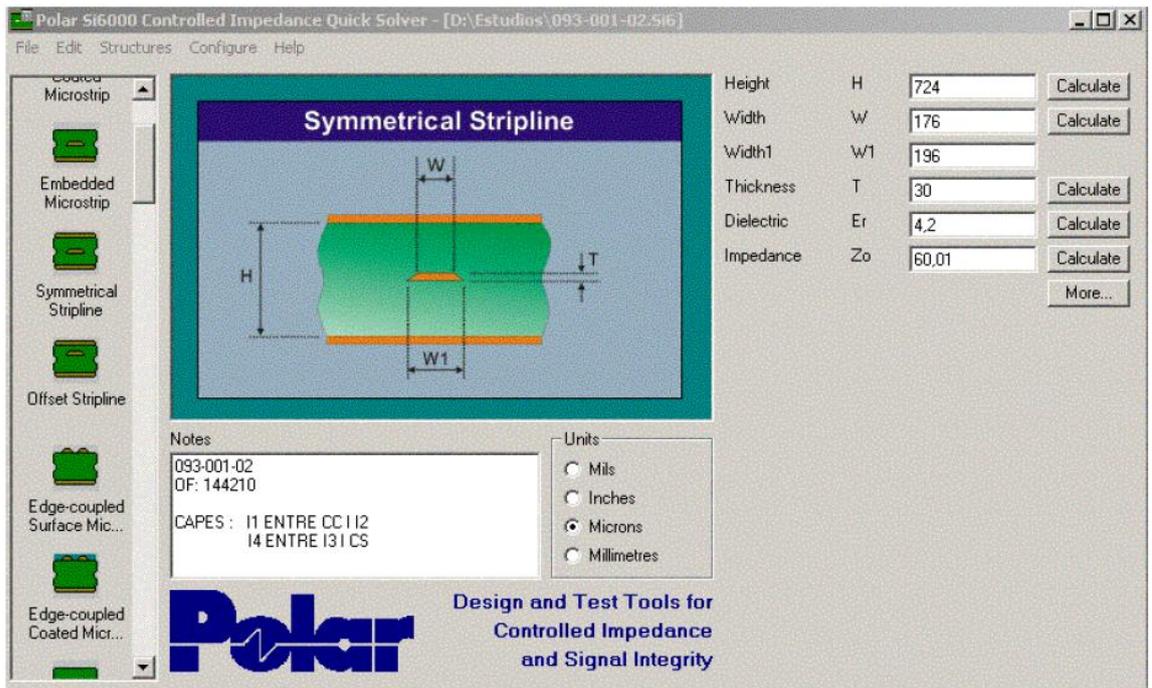
Figure B.5: Manufacturer controlled impedance solver result for 60Ω impedance

APPENDIX C I^{2}C MEZZANINE SCHEMATIC
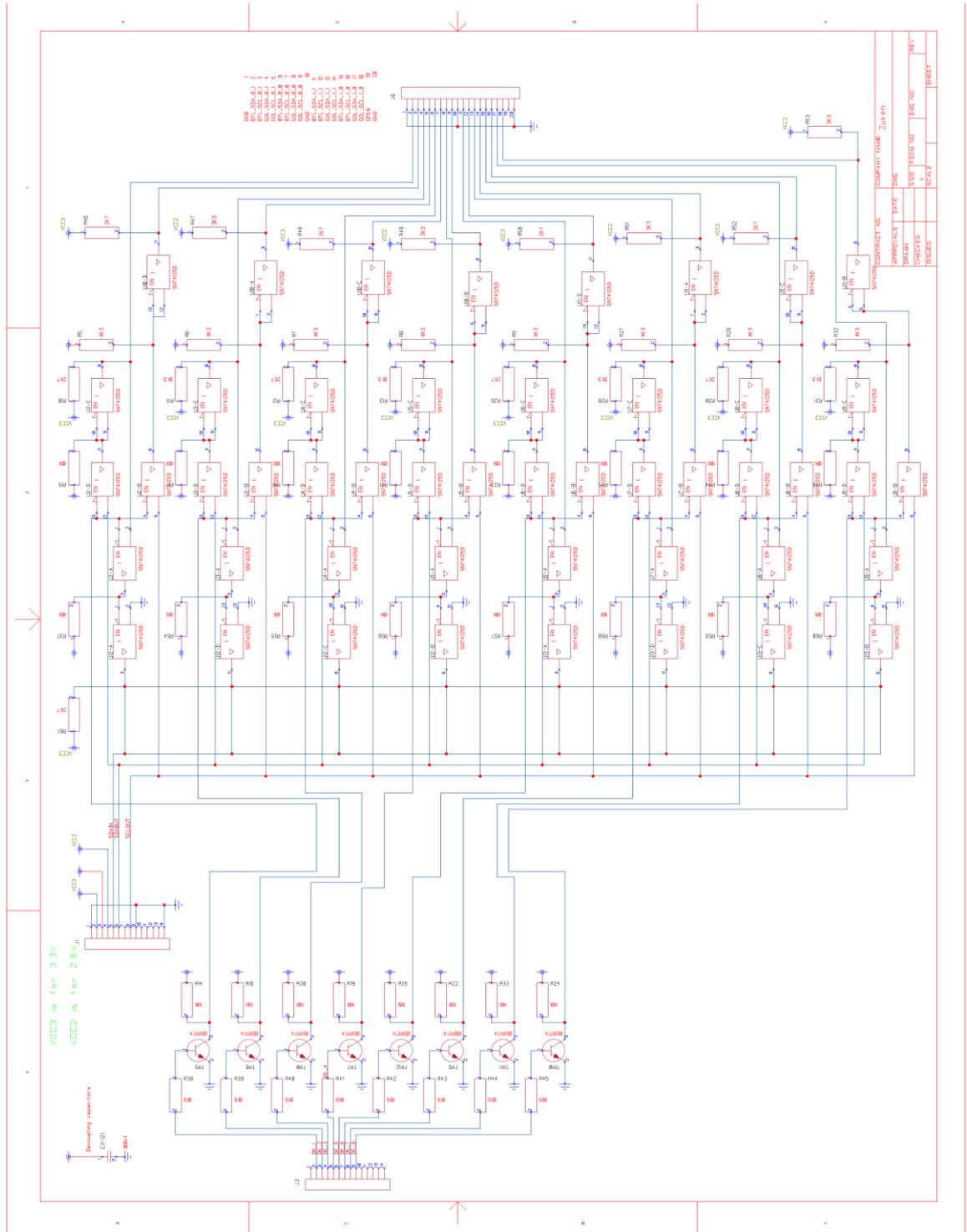
Figure C.1: I^{2}C 8 buses mezzanine based in F125 tri-state gates

APPENDIX D   SERVICE BOX I^{2}C ADDRESSING SCHEME

Long Multiplexed I2C bus addressing

A6	A5	A4	A3	A2	A1	A0
BOARD-ID		1	0	CHIP-ID		0

Beetle Chip I2C address definition.

A6	A5	A4	A3	A2	A1	A0
BOARD-ID		1	0	CHIP-ID		X

GOL I2C address definition. 'X' bits are used internally

A6	A5	A4	A3	A2	A1	A0
BOARD-ID		0	1	X	X	X

DCU I2C address definition. 'X' bits are used internally

Control Board I2C bus addressing

A6	A5	A4	A3	A2	A1	A0
0	1	0	0	X	X	X

Delay25 Chip I2C address definition. 'X' bits are used internally

A6	A5	A4	A3	A2	A1	A0
1	0	1	1	1	1	X

TTCrq I2C address definition. 'X' bits are used internally

A6	A5	A4	A3	A2	A1	A0
1	1	0	0	X	X	X

DCU I2C address definition. 'X' bits are used internally

Figure D.1: ST I^{2}C addressing scheme

As each readout hybrid is associated with a Digitizer Board, a Board-ID is defined, ranging from 0 to 3 and set by the I^{2}C address bits A6 and A5. All devices (Beetle, GOL, DCUF) linked to a specific board will be programmed with the same Board-ID. This is done with the common lines I2C ADR5 and I2C ADR6, which are connected to all I2C devices.

The logic state of these lines is determined for each Digitizer Board and its associated readout hybrid by the backplane slot, in which the Digitizer Board has been placed.

For the GOL/DCUF I^{2}C bus, address Bit A4 determines the device type, where '1' is associated with a GOL chip and '0' with a DCUF.

The GOL bits A2 and A1 are used for the Chip-ID, ranging from 0 to 3, which corresponds to the consecutive numbering of the Beetle chips on a given hybrid. In addition, associating bits A2 and A1 on the readout hybrid to the four Beetle readout chips creates identical addresses for each Beetle/GOL pair. No address conflict occurs since the I^{2}C buses are held separate. As the bit A0 is used for programming internal GOL registers, the matching of addressed cannot be extended to the full I^{2}C address. The remaining bit A0 of every Beetle is connected to '0' as no further addresses are needed for the Beetle I^{2}C bus.

SPECS internal DCUs have its own I^{2}C bus each. The I^{2}C address is 0x00 for both.

APPENDIX E   THE ST LV SYSTEM CONFIGURATION AND CONTROL

This section is intended to give a detailed description on how to set up the low voltage system of the ST. The description will dig into software details as long as it is necessary, and that's the main reason why we will explain some features about the OPC server configuration of the MARATON power supplies, but we won't deal at all with PVSS. It is intended as a guideline not as a specific implementation description.

The system consists of two distinguishable parts: the MARATON PS and the backplane regulators. The PS control is performed by the so-called RCM MARATON modules, which are controlled via an OPC server. The regulators are controlled by the Control Board, based on the SPECS system and that's the reason why the control software is based on a DIM server named "Specsserver".

E.1   THE MARATON PS CONFIGURATION AND CONTROL

The channel output voltage, current limit and dedicated over voltage protection level can be adjusted via hardware and software, although hardware settings always prevail over software ones. The hardware settings are implemented via potentiometers (Figure E.1) and we can adjust the:

- Output Voltage at the load (sense point)
- Maximum Output Voltage at the Terminals of the Power Box (OVP)
- Maximum current
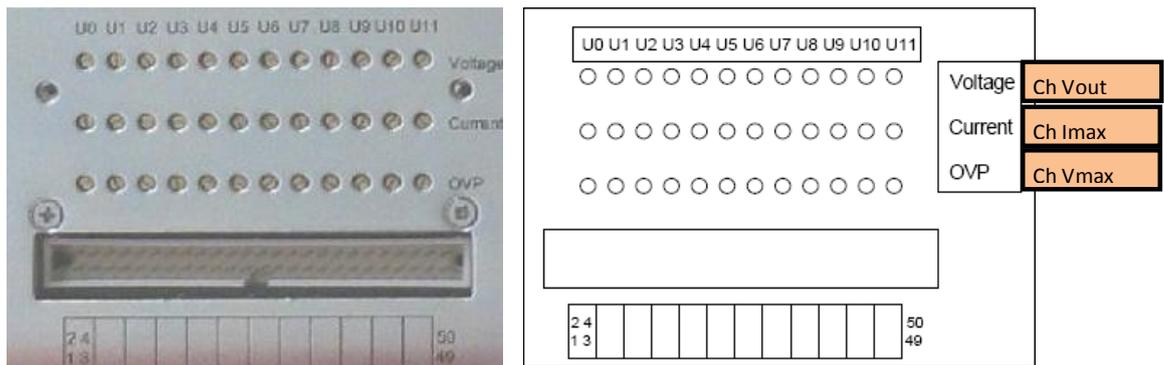
Figure E.1: MARATON Power box potentiometers

One can turn the potentiometers and measure the change of the trimmed item, e.g. output voltage, but for the current limit and OVP this is difficult. So a 50-pin connector is provided, which gives access to the relative voltage values of the settings. For each channel there is one GND pin, one OVP, one V-Set and one I-limit (Table E.1).

Connecting a voltmeter between the GND pin and the OVP, I-limit or V-Set pin of the channel of interest we obtain the real value by multiplying a scale factor with the measured relative value. The scale factor is different for the different MDC module types (Table E.2).

The power box is connected to the RCM module by two 37 pin D-Sub connectors. For each of the power channels 3 signals are available: voltage and current monitoring, and a combined inhibit/status signal. To simplify the design of the remote control, the monitoring signals are scaled depending on the maximum MDC module output voltage and the maximum output current (see the scaling factor in Table E.3).The PS control is carried out by the RCM module. This module can monitor and control up to 12 channels (6 per 37 pin connector). It has an USB and a RJ-45 port: the RJ-45 is used to communicate with the RCM by TCP/IP using an OPC client over SNMP protocol; the USB connector can locally connect to the RCM using the manufacturer program MUHSE.

Channel	1		2		3		4		5		6		7		8		9		10		11		12		
	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40	42	44	46	48	50
	1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31	33	35	37	39	41	43	45	47	49

- GND
- OVP
- I limit
- V output

Table E.1: Frontal pins assignment

Module	Voltage	OVP	Current
MDC 8 V @ 50 A	0.940 V/V	1.881 V/V	6.0 A/V
MDC 15V @ 20 A	1.989 V/V	3.978 V/V	3.0 A/V

Table E.2: Frontal pins scaling factors

Module	Voltage Scaling	Current Scaling
8V@50A	1.2 V/V	12 A/V
15V@20A	2 V/V	6.0 A/V *

Table E.3: Channels signals scaling factors
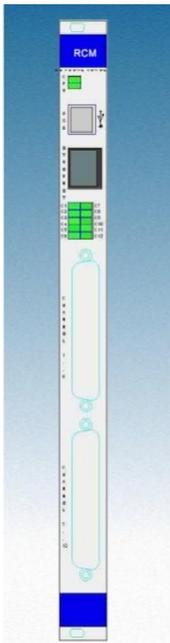
The following direct control functions are possible from MUHSE and from OPC:

- Measurement of each MARATON output voltage.
- Measurement of each MARATON output current.
- Read the status of each MARATON.
- Switch a MARATON channel on or off.

The on-board microcontroller extends this functionality by comparing these values with additional limits, which can be modified via software. The reaction to any failure can be selected independently, and vary from ignore the failure, to switch off all the channels.

OPC and MUHSE software are not equivalent, and each one has its own functionalities. The MUHSE is an application developed by Wiener just for local control and configuring the MARATON: current limits, sense voltage maximum allowed value, set scaling factors, IP address… The OPC server is an industry standardized way for controlling the PS by any OPC client, in our case the PVSS SCADA.

The Figure E.3 shows a screenshot of the MUHSE panel for configuring a MARATON channel. As we can see, apart from the scaling factors, it is necessary to set the maximum allowed values for the current and the over-voltage in the RCM and in the OPC server. Also the action to be executed when the limit is reached must be set (ignore the failure, switch channel off...).

Figure E.2: RCM module
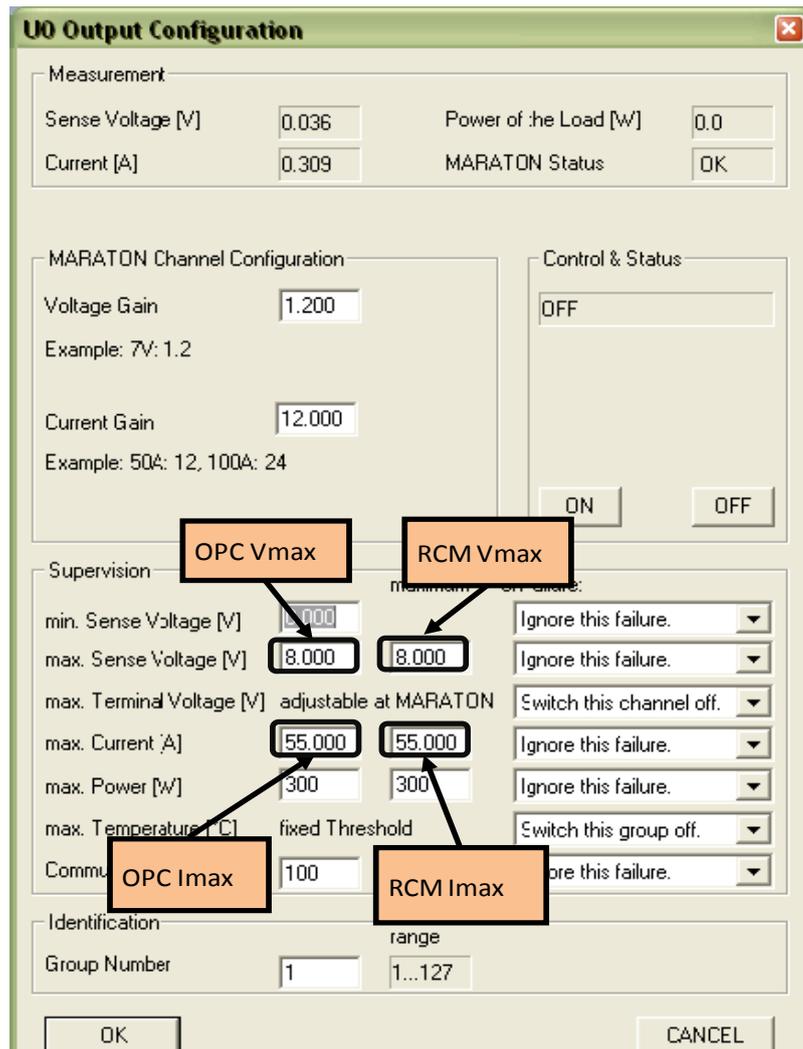
Figure E.3: MUHSE screen-shot for an single channel

E.2   THE CURRENT AND VOLTAGE LIMITS

The V-Max and I-Max values to be set are defined in the tables below. The setting will depend on the voltage drop in the cable, so also in the cables length and whether on the SB is completely filled with digitizer boards (half of the IT Service boxes will host 12 boards and the other half 16). The most restrictive setting will be the OPC value (10%), then the RCM setting (20%) and finally the potentiometer (25%).

IT 1-8V@50A															
		SvceBox		cable				contact losses		MARATON (potentiometers)			RCM		OPC	
		V [V]	I [A]	lengt h [m]	xsec [mm²]	Vdro p [V]	R [Ω]	Vdro p [V]	Vou t [V]	Vma x [V]	Imax [A]	Vma x [V]	Imax [A]	Vma x [V]	Imax [A]
										125 %	125 %	120 %	120%	110 %	110%
IT16	IT1 Out	4,50	24	18	25	0,69		0,00	5,19	6,49	30,00	6,23	28,80	5,71	26,40
	IT3 In	4,50	24	22	25	0,84		0,00	5,34	6,68	30,00	6,41	28,80	5,88	26,40
IT12	IT1 Out	4,50	18	20	25	0,58		0,00	5,08	6,35	22,50	6,09	21,60	5,58	19,80
	IT3 In	4,50	18	22	25	0,63		0,00	5,13	6,42	22,50	6,16	21,60	5,65	19,80
MAX									5,34	6,68	30,00	6,41	28,80	5,88	26,40

Table E.4: V-Max and I-Max values of the IT 1-8V module for the different Svce Boxes

IT 2-15@20A															
		SvceBox		cable			contact losses		Maraton (hardware)			RCM		OPC	
		V [V]	I [A]	length [m]	xsec [mm2]	Vdrop [V]	R [Ω]	Vdrop [V]	Vout [V]	Vmax [V]	Imax [A]	Vmax [V]	Imax [A]	Vmax [V]	Imax [A]
										125%	125%	120%	120%	110%	110%
IT16	IT1 Out	7	8	18	16	0,36		0,00	7,36	9,20	10,00	8,83	9,60	8,10	8,80
	IT3 In	7	8	22	16	0,44		0,00	7,44	9,30	10,00	8,93	9,60	8,18	8,80
IT12	IT1 Out	7	6,5	20	16	0,33		0,00	7,33	9,16	8,13	8,79	7,80	8,06	7,15
	IT3 In	7	6,5	22	16	0,36		0,00	7,36	9,20	8,13	8,83	7,80	8,09	7,15
MAX									7,44	9,30	10,00	8,93	9,60	8,18	8,80

Table E.5: V-Max and I-Max values of the IT 2-15V module for the different Svce Boxes

TT 1-8V@50A															
		SvceBox		cable			contact losses		MARATON (potentiometers)			RCM		OPC	
		V [V]	I [A]	length [m]	xsec [mm²]	Vdrop [V]	R [Ω]	Vdrop [V]	Vout [V]	Vmax [V]	Imax [A]	Vmax [V]	Imax [A]	Vmax [V]	Imax [A]
										125%	125%	120%	120%	110%	110%
TT	Access Bottom	4,50	24	17	25	0,65		0,00	5,15	6,44	30,00	6,18	28,80	5,67	26,40
	Access Top	4,50	24	19	25	0,73		0,00	5,23	6,54	30,00	6,28	28,80	5,75	26,40
TT	Cryo Bottom	4,50	24	32	25	1,23		0,00	5,73	7,16	30,00	6,87	28,80	6,30	26,40
	Cryo Top	4,50	24	34	25	1,31		0,00	5,81	7,26	30,00	6,97	28,80	6,39	26,40
MAX									5,81	7,26	30,00	6,97	28,80	6,39	26,40

Table E.6: V-Max and I-Max values of the TT 1-8V module for the different Svce Boxes

TT 2-15@20A															
		SvceBox		cable			contact losses		Maraton (hardware)			RCM		OPC	
		V [V]	I [A]	length [m]	xsec [mm²]	Vdrop [V]	R [Ω]	Vdrop [V]	Vout [V]	Vmax [V]	Imax [A]	Vmax [V]	Imax [A]	Vmax [V]	Imax [A]
										125%	125%	120%	120%	110%	110%
TT	Access Bottom	7	8	17	16	0,34		0,00	7,34	9,18	10,00	8,81	9,60	8,07	8,80
	Access Top	7	8	19	16	0,38		0,00	7,38	9,23	10,00	8,86	9,60	8,12	8,80
TT	Cryo Bottom	7	8	32	16	0,64		0,00	7,64	9,55	10,00	9,17	9,60	8,40	8,80
	Cryo Top	7	8	34	16	0,68		0,00	7,68	9,60	10,00	9,22	9,60	8,45	8,80
MAX									7,68	9,6	10	9,216	9,6	8,44	8,8

Table E.7: V-Max and I-Max values of the IT 2-15V module for the different Svce Boxes

E.3   THE RCM IP SETTINGS

As the RCM modules connection to a control computer is based upon a TCP/IP connection (SNMP) they need an IP address as for any device in the LHCB network. The address are also set using the MUHSE program. The official IP addresses and device names assigned to the ST RCMs are the following:

MARATON RCM	IP address	Powered svce boxes
itmatna01.lbdaq.cern.ch	10.130.20.162	IT1A, IT3A
itmatna02.lbdaq.cern.ch	10.130.20.163	IT2A, IT3A
itmatnc01.lbdaq.cern.ch	10.130.20.164	IT1C, IT3C
itmatnc02.lbdaq.cern.ch	10.130.20.165	IT2C, IT3C

Table E.8: IT RCMs network configuration

MARATON RCM	IP address	Powered svce boxes
ttmatna01.lbdaq.cern.ch	10.130.20.166	TTAT
ttmatna02.lbdaq.cern.ch	10.130.20.167	TTAB
ttmatnc01.lbdaq.cern.ch	10.130.20.168	TTCT
ttmatnc02.lbdaq.cern.ch	10.130.20.169	TTCB

Table E.9: TT RCMs network configuration

E.4   THE OPC SERVER CONFIGURATION

After installing in the control computer the latest version on the MARATON PS some few things still need to be configured before the power supplies can be operated. First of all, the configuration file must be as follows, which is the sample configuration file for the IT MARATON OPC server:

```
<?xml version="1.0"?>
<!--
This is the WienerOPCS server configuration file.
-->
<WienerOPCS version="1.0" date="20060307" time="111249">

<!--
Element Debug: All Debug specific parameters
Attribute          Default            Description
_____
TraceErrorLevel    0xffffffff         All error messages enabled
TraceWarningLevel  0xffffffff         All warning messages enabled
TraceInfoLevel     0x00000000         Some informational messages enabled
TraceDebugLevel    0x00000000         All debug messages disabled. Enabling
                                      debug will produce massive output
                                      to the log files ans slow down the
                                      system!
TraceFileMaxSize   1000000            Maximum trace file size
TraceFile          OPCSlog1.txt       Name of first trace file
TraceFile2nd       OPCSlog2.txt       Name of second trace file
TraceFileEnable    1                  Trace to file disabled (0) or enabled (1)
-->
<Debug
  TraceErrorLevel="0xf0000000"
  TraceWarningLevel="0xf0000000"
  TraceInfoLevel="0x00000000"
  TraceDebugLevel="0x00000000"

  TraceFileMaxSize="1000000"
  TraceFile="OPCSlog1.txt"
  TraceFile2nd="OPCSlog2.txt"
  TraceFileEnable="1"
/>
<!--  TraceInfoLevel="0xF0000000"
  TraceDebugLevel="0xF0000000"-->

<!--
Element SNMP: All SNMP specific parameters
Attribute          Default            Description
_____
ReadCommunity      public             Community name for SNMP get operations.
WriteCommunity     private            Community name for SNMP set operations.
-->
<SNMP
  ReadCommunity="public"
  WriteCommunity="guru"
/>
<!--
Element Crates: Collection of all crates controlled by this server
-->
<Crates
```

- 293 -

```
                ShowMeasurementTemperature="NO"
>
  <Crate Alias="itmatna01" Type="RCM" Transport="SNMP" Address="10.130.20.162">
    <Channels>
      <Channel0/>
      <Channel1/>
      <Channel2/>
      <Channel3/>
      <Channel4/>
      <Channel5/>
      <Channel6/>
      <Channel7/>
      <Channel8/>
      <Channel9/>
      <Channel10/>
      <Channel11/>
    </Channels>
    <Groups>
    </Groups>
  </Crate>
  <Crate Alias="itmatna02" Type="RCM" Transport="SNMP" Address="10.130.20.163">
    <Channels>
      <Channel0/>
      <Channel1/>
      <Channel2/>
      <Channel3/>
      <Channel4/>
      <Channel5/>
      <Channel6/>
      <Channel7/>
      <Channel8/>
      <Channel9/>
      <Channel10/>
      <Channel11/>
    </Channels>
    <Groups>
    </Groups>
  </Crate>
  <Crate Alias="itmatnc01" Type="RCM" Transport="SNMP" Address="10.130.20.164">
    <Channels>
      <Channel0/>
      <Channel1/>
      <Channel2/>
      <Channel3/>
      <Channel4/>
      <Channel5/>
      <Channel6/>
      <Channel7/>
      <Channel8/>
      <Channel9/>
      <Channel10/>
      <Channel11/>
    </Channels>
    <Groups>
    </Groups>
  </Crate>
  <Crate Alias="itmatnc02" Type="RCM" Transport="SNMP" Address="10.130.20.165">
    <Channels>
      <Channel0/>
      <Channel1/>
      <Channel2/>
      <Channel3/>
      <Channel4/>
      <Channel5/>
      <Channel6/>
      <Channel7/>
      <Channel8/>
      <Channel9/>
      <Channel10/>
      <Channel11/>
    </Channels>
    <Groups>
    </Groups>
  </Crate>
</Crates>
</WienerOPCS>
```

The last step for setting up the OPC server properly is registering it. This must be done in a command shell:

- Go to the installation folder, ie, *C:\Program Files\W-IE-NER\WienerOPCS*
- Unregister the server typing: *WienerOPCS –u*
- Register the server typing: *WienerOPCS –r*

After this, any OPC client should be able to connect to the PS and take control of it.

E.5 THE POWER REGULATOR'S CONTROL

The only actions that can be performed in the power regulators are switching them on and off via the INH signal and monitor the OCM (over-current status) signal. These tasks are performed by the CB, and more precisely, by the SPECS slaves housed in it.

The CB must always be powered and it is mandatory that all the CBs belonging to the same SPECS control link are powered as well. In case any of the SB is off, the downstream link service boxes are not reachable anymore since the link repeaters implemented in the CB are not powered.
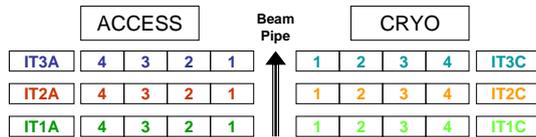
Figure E.4: IT SPECS links in different colors
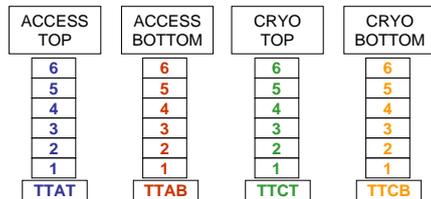
Figure E.5: TT SPECS links in different colors

Every CB hosts 2 SPECS slaves and each slave in the link has an unique address, so when trying to operate the regulators of a given SB it is necessary to address the right CB (Table E.10).

The communication between the control computer and the CB is commanded by the SPECS master in the computer. The "Specsserver" is the software interface for this SPECS master hardware. It is based on the DIM client/server protocol architecture and it can be remotely accessed by the PVSS DIM client. The "Specsserver" must run in the same computer where the SPECS master is installed. As it is based on the DIM protocol it must publish the service against a DIM Name Server also known by the DIM client.

The regulators inhibit control is performed through the 32 bit I/O register. For operating it, the corresponding regulator I/O line mode must be set to output (implies setting a "1") and the value must be set to "0" for switching it on and to "1" for setting it to off. The SPECS slave registers for setting the mode of these I/O lines are CONFREGOUT_LSB (bits 0-15) and CONFREGOUT_MSB (16-31), and the registers for writing/reading the output value are REGOUT_LSB (0-15) and REGOUT_MSB (16-31).

For the OCM monitoring the corresponding bit must be periodically read. The mode of the SPECS I/O line must be set to input (default starting mode) by setting it to "0". A low level present in the line (a "0" coming from the regulator OCM signal) implies that the regulator is in over-current status.

For instance, in case we just wanted to power on the DB of the service box partition 2 for the TT, the steps would be the following:

- Set the mode of the line #15 of the slave1 to output ("1"): write a "b'x1xxxxxxxxxxxxxx" to register CONFREGOUT_LSB[15:0].
- Set the value of the line #15 of the slave1 to "0" (regulator enabled): write a "b'x0xxxxxxxxxxxxxx" to register REGOUT_LSB[15:0].

The previous sequence of commands should in principle be correct, but in case that what we wanted to do is setting the regulator to off state then we would suffer a power glitch. The initial state of the I/O line is input mode, so there's a "0" level in the line, which will be the default value when moving it to output mode. Like this the regulator would go to on state for a while equivalent to the time that the Specs system needs to process the next command which sets it to off, and that's why we get a power glitch. To get rid of it we have to set the value that we want the line to have prior to set the mode. This works because the written value is "memorized" by the SPECS slave. The complete right sequence should then be as follows:

- Set the value of the line #15 of the slave1 to "1" (regulator disabled): write a "b'x1xxxxxxxxxxxxxx" to register REGOUT_LSB[15:0].
- Set the mode of the line #15 of the slave1 to output ("1"): write a "b'x1xxxxxxxxxxxxxx" to register CONFREGOUT_LSB[15:0].
- Set the value of the line #15 of the slave1 to "1" (regulator enabled): write a "b'x0xxxxxxxxxxxxxx" to register REGOUT_LSB[15:0].

The whole procedure for starting up and monitoring a partition of hybrids/dig Boards should be implemented is just shown below. We will describe it for example for the partition 3 of an IT service box:

- Switch ON the dig boards:

      a. Set the line #17 of slave 1 to "1" (OFF) → write "b'xxxxxxxxxxxxxxx1" to REGOUT_MSB[15:0].

      b. Configure it to OUTPUT mode ("1") → write "b'xxxxxxxxxxxxxxx1" to CONFREGOUT_MSB[15:0] of slave 1.

      c. Set it to ON ("0") → write "b'xxxxxxxxxxxxxxx0" to REGOUT_MSB[15:0] of slave 1.

- Switch ON the hybrids:

      a. Set lines #16,18,19,20 of slave 1 to "1" → write "b'1xxxxxxxxxxxxxxx" to REGOUT_LSB[15:0] of slave 1 and "b'xxxxxxxxxxxx111x" to REGOUT_MSB[15:0].

      b. Configure them to OUTPUT mode ("1") → write "b'1xxxxxxxxxxxxxxx" to CONFREGOUT_LSB[15:0] of slave 1 and "b'xxxxxxxxxxxx111x" to CONFREGOUT_MSB[15:0] of slave 1

      c. Set them to ON ("0") → write "b'0xxxxxxxxxxxxxxx" to REGOUT_LSB[15:0] of slave 1 and "b'xxxxxxxxxxxx000x" to REGOUT_MSB[15:0] of slave 1.

- Monitor the dig board regulators OCM signals:

      a. Set the lines #25, 26, 27, 28 of slave 2 to INPUT mode: write "b'xxxx0000xxxxxxxx" CONFREGOUT_MSB[15:0].

      b. Periodically monitor (read) the lines #25, 26, 27, 28.

- Monitor the hybrid regulators OCM and the Beetle Vcc signals:

      a. Reset and initialize the DCUs of Dig boards plugged into slots 12,13,14,15 if it wasn't done yet.

      b. Periodically monitor (read) the DCU channels 3 and 4 for reading the OCM level of both regulators.

Periodically monitor (read) the DCU channel 5 for the Vcc power voltage value of the Beetle.

CB number	Slave1 address	Slave2 address
1	1	2
2	3	4
3	5	6
4	7	8
5	9	10
6	11	12

Table E.10: Slaves addresses

APPENDIX F   THE ST HV SYSTEM CONFIGURATION AND CONTROL

This section is intended to give a detailed description on how to set up the high voltage system of the ST. The description will dig into software details as long as it is necessary, and that's the main reason why we will explain some features about the OPC server configuration of the CAEN power supplies, but we won't deal at all with PVSS. It is intended as a guideline not as a specific implementation description.

F.1   THE CAEN PS CONFIGURATION AND CONTROL

CAEN, in collaboration with CERN, has developed an OPC server which allows powerful, flexible and yet simple control of its power supply systems, indifferently through TCP/IP, RS232 or H.S. CAENET communication path, by any OPC compliant client application.

Apart from the server, CAEN provides the OPC Server Configurator, which permits to define the communication network and the address of the crate to be controlled. Once the address of the crate is defined in the server, all the items of this crate (crate status, boards, channels parameters…) are accessible via an OPC Client in a tree view.

There is an OPC server and a control computer for each of the CAEN crates, and the communication protocol is TCP/IP. The figure shows the OPC control node names as well as the CAEN crates names and IP addresses. The IP address of the CAEN crate must be set for the first time by logging in locally, and from then on one can do it via telnet.
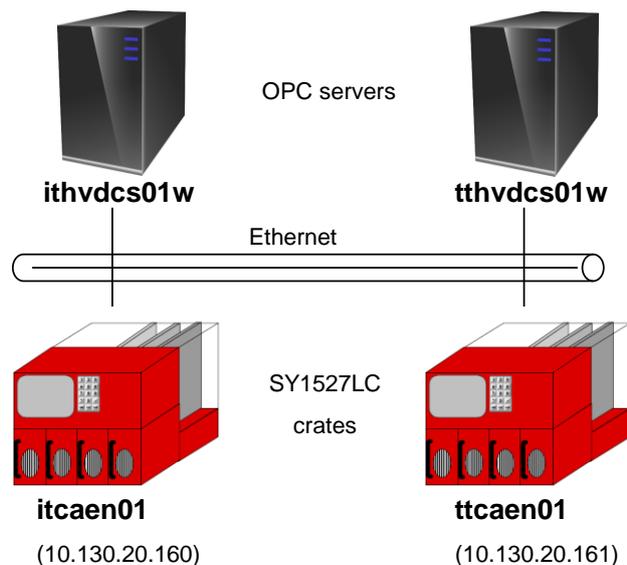
Figure F.1: HV networking scheme

There are some settings of the A1511B module that the user must manually set in the board itself putting some jumpers into predefined positions. The settings that must be set are the operational current range of the module and the mode of the enable signal. Apart from this, in order to operate the board, the enable lines in the board frontal panel and in the output connectors must be terminated properly, which is done by the DSS system and in the patch-panel respectively.

The PS is remotely controllable via OPC server and there at three different: channel, board and crate. For this purpose, the OPC items that represent the different device properties are fully identified in the server address space during data access by an *ItemID* which has the general syntax *PowerSupplyName.BoardXX.ChanYYY.ItemName*.

- Items of type *PowerSupplyName.ItemName* are associated with general system parameters.
- Items of type *PowerSupplyName.BoardXX.ItemName* are associated with boards' parameters.

- Items of type *PowerSupplyName.BoardXX.ChanYYY.ItemName* are associated with channels' parameters.

The total amount of properties that the user can handle in these PS is quite huge and a detailed description can be found in the CAEN OPC server manual[1]. Among the general control and status items of crate the most useful and important ones are:

- *'kill'*: allows to switch OFF at the maximum rate all of the channels.
- *'clear alarm'*: clears channel's alarm messages.
- *'FrontPanOut'*: 16 bit pattern indicating the system output status.
- *'HvPwSM'*: shows the PS module status

At the board level the interesting items are:

- '*BdStatus'*: returns the status of generic board parameters.
- '*HVMax'*: hw voltage limit set by trimmer on the board.
- '*Temp'*: returns the temperature of the board.

At the channel level, the important settings are:

- '*V0set'*: sets the channel output voltage.
- '*I0set'*: sets the channel maximum current.
- '*RUp'*: sets the ramping up rate.
- '*RDWn'*: sets the ramping down rate.
- '*Trip'*: allows to program the trip time.
- '*VMon'*: returns back the output voltage value.
- '*IMon'*: returns back the current value.
- '*Status'*: returns back a 16 bit pattern showing the channel status.
- '*Pw'*: allows to switch ON/OFF the channel.
- '*POn'*: allows to select the power ON option.
- '*PDwn'*: allows to select the power-down option.

F.2   THE CAEN IP SETTINGS

As the CAEN PS control computer is based upon a TCP/IP connection and they need an IP address as for any device in the LHCb network. The addresses have to be for the first time by locally logging in, and from then on one can connect and change them via telnet. The official IP addresses and device names assigned to the ST CAEN crates are the following:

CAEN machine	IP address	Powered system
itcaen01.lbdaq.cern.ch	10.130.20.160	IT
ttcaen01.lbdaq.cern.ch	10.130.20.161	TT

Table F.1: ST CAEN network configuration

APPENDIX G   CB CALIBRATION CONSTANTS

Ctrl Board ID	DCU1-Ch1 Mx (slope)	K (offset)	DCU1-Ch2 Mx	K
1	-0,170604	227,035	-0,169320	225,167
2	-0,168510	224,909	-0,166283	222,119
3	-0,167162	224,176	-0,166806	223,361
4	-0,170120	227,736	-0,168047	224,672
5	-0,166773	225,107	-0,164575	221,593
6	-0,167690	224,867	-0,166256	222,956
7	-0,168422	225,476	-0,167033	223,371
8	-0,167125	225,828	-0,165986	223,787
9	-0,166316	224,797	-0,164688	222,332
10	-0,168860	226,873	-0,167701	224,660
11	-0,167087	225,577	-0,165210	222,842
12	-0,168853	225,933	-0,165999	222,328
13	-0,169958	227,866	-0,167584	224,809
14	-0,170951	230,262	-0,168426	226,530
15	-0,167410	224,963	-0,165305	221,551
16	-0,166853	225,146	-0,164641	221,541
17	-0,168191	225,761	-0,165775	222,390
18	-0,165931	225,031	-0,164758	223,600
19	-0,170294	226,303	-0,168988	224,443
20	-0,166552	224,820	-0,163531	220,264
21	-0,166611	224,190	-0,164340	220,884
22	-0,168309	227,549	-0,165786	223,710
23	-0,165118	224,438	-0,162279	220,570
24	-0,170003	228,621	-0,168476	226,125
25	-0,171830	229,022	-0,170181	225,989
26	-0,167079	225,085	-0,166016	223,297
27	-0,167132	226,374	-0,165470	224,124
28	-0,168491	226,119	-0,166371	223,231
29	-0,170904	227,808	-0,170703	227,586
30	-0,167844	227,909	-0,165038	223,349
31	-0,169468	228,512	-0,166925	224,966
32	-0,169285	227,798	-0,167343	224,686
33	-0,165476	227,486	-0,163637	225,126
34	-0,166922	223,938	-0,166350	223,004
35	-0,168896	226,464	-0,167028	223,961
36	-0,167106	223,940	-0,165000	221,183
37	-0,170131	226,970	-0,167364	223,201
38	-0,169775	228,541	-0,167419	224,546
39	-0,168238	224,830	-0,166080	221,605
40	-0,164204	223,773	-0,162606	221,175
41	-0,168857	226,725	-0,166866	223,855
42	-0,167535	226,229	-0,166245	223,715
43	-0,167502	224,659	-0,165710	222,356
44	-0,168915	222,856	-0,166515	219,666
45	-0,170452	224,157	-0,168729	221,770
46				
47	-0,168787	225,464	-0,166144	221,377
48	-0,172843	229,119	-0,171092	226,543
49	-0,170812	227,405	-0,170743	226,714
50	-0,169873	226,967	-0,168480	224,400
51	-0,167165	224,977	-0,165471	222,197
52	-0,166171	222,882	-0,165178	221,421
53	-0,165340	224,909	-0,163616	222,137
54	-0,168673	227,913	-0,166181	224,396
55	-0,170001	228,390	-0,168216	225,653
56	-0,167981	226,798	-0,165689	223,390
57	-0,167910	226,150	-0,165903	223,232
58	-0,171020	227,958	-0,169135	225,403
59	-0,166340	227,365	-0,163654	223,157
60	-0,167839	226,066	-0,166525	224,323

Table G.1: Calibration constants for channels 1 and 2 of the DCU1 in the CB

Ctrl Board ID	DCU2-Ch1 Mx	K	DCU2-Ch2 Mx	K	DCU2-Ch3 Mx	K	DCU2-Ch4 Mx	K
1	-0,165789	224,422	-0,165590	224,415	-0,165266	223,612	-0,165644	224,312
2	-0,166631	222,969	-0,165534	221,380	-0,170062	226,557	-0,166041	221,213
3	-0,168062	224,820	-0,167178	223,351	-0,166987	223,374	-0,166855	222,684
4	-0,166497	223,569	-0,167413	225,003	-0,166822	224,429	-0,166624	223,481
5	-0,163556	222,946	-0,163080	222,130	-0,162742	222,077	-0,163710	222,949
6	-0,166918	221,878	-0,167131	222,124	-0,165917	220,807	-0,167182	222,793
7	-0,168130	224,853	-0,167460	223,367	-0,166888	222,895	-0,166650	222,506
8	-0,166220	223,829	-0,166326	223,652	-0,165407	222,489	-0,166057	223,428
9	-0,167568	223,589	-0,168263	224,633	-0,166899	222,602	-0,166101	221,618
10	-0,167137	224,530	-0,167093	223,960	-0,166428	223,679	-0,167425	224,756
11	-0,165879	223,322	-0,165366	222,357	-0,166590	224,146	-0,166241	223,438
12	-0,168281	222,637	-0,169090	223,730	-0,168284	222,885	-0,168898	223,282
13	-0,166321	223,381	-0,166895	224,014	-0,165700	222,311	-0,167485	224,907
14	-0,169251	225,623	-0,169259	225,665	-0,168976	225,394	-0,169075	225,343
15	-0,163337	222,496	-0,163632	222,873	-0,163175	222,587	-0,163666	222,576
16	-0,164902	221,477	-0,165137	221,575	-0,164371	220,684	-0,166444	223,381
17	-0,165771	222,090	-0,166149	222,708	-0,165783	222,082	-0,169890	226,989
18	-0,163636	222,090	-0,163851	222,574	-0,163016	221,417	-0,163593	221,806
19	-0,166803	224,264	-0,166823	224,225	-0,167090	224,487	-0,165878	222,746
20	-0,165033	220,388	-0,165300	220,634	-0,164555	219,722	-0,164675	219,654
21	-0,161702	221,505	-0,161885	221,458	-0,161043	220,164	-0,161863	221,298
22	-0,169208	224,494	-0,169199	224,509	-0,168485	223,517	-0,169233	224,793
23	-0,166244	221,919	-0,166416	221,909	-0,165714	220,910	-0,166120	221,017
24	-0,165789	225,466	-0,165738	225,160	-0,165272	224,480	-0,165721	225,075
25	-0,168268	225,173	-0,168502	225,343	-0,168047	225,150	-0,166007	222,093
26	-0,164453	222,366	-0,165447	223,801	-0,164421	222,111	-0,165082	222,741
27	-0,169615	225,090	-0,169185	224,908	-0,168165	223,294	-0,169298	224,918
28	-0,165381	224,018	-0,165245	224,108	-0,164798	223,334	-0,165581	223,699
29	-0,169596	227,021	-0,170048	227,449	-0,169269	226,344	-0,169910	226,702
30	-0,167593	225,448	-0,167300	224,834	-0,166753	224,227	-0,167636	225,164
31	-0,165687	225,664	-0,165511	225,820	-0,165303	225,366	-0,165269	225,166
32	-0,169152	224,313	-0,169144	224,863	-0,168627	224,089	-0,165059	218,984
33	-0,167614	224,082	-0,168618	225,493	-0,167250	223,404	-0,168019	223,785
34	-0,165176	223,423	-0,165242	223,317	-0,164751	222,571	-0,165906	224,379
35	-0,168058	224,992	-0,168234	225,054	-0,166950	223,293	-0,167577	223,918
36	-0,163721	220,254	-0,164189	220,836	-0,163184	219,629	-0,155991	209,754
37	-0,167885	222,328	-0,169376	224,691	-0,167437	221,852	-0,152541	202,338
38	-0,167484	224,632	-0,167403	224,479	-0,166963	223,933	-0,167370	224,512
39	-0,166979	221,393	-0,167502	222,164	-0,166716	221,355	-0,167221	221,447
40	-0,166367	222,314	-0,166778	222,896	-0,165759	221,553	-0,166917	223,402
41	-0,166574	223,556	-0,167036	224,435	-0,166844	224,463	-0,166812	224,077
42	-0,165938	222,602	-0,166791	223,831	-0,166296	223,014	-0,167235	224,488
43	-0,165690	222,486	-0,166539	223,760	-0,165418	222,142	-0,165764	222,017
44	-0,163812	218,994	-0,165670	221,569	-0,165196	221,070	-0,164730	219,743
45	-0,165812	223,158	-0,165708	222,709	-0,165445	222,688	-0,165654	222,882
46								
47	-0,166155	222,062	-0,166153	222,124	-0,166030	221,724	-0,165870	221,223
48	-0,168121	225,768	-0,168021	225,646	-0,167065	224,179	-0,168329	226,347
49	-0,168763	223,480	-0,169332	224,242	0,170580	225,691	-0,169092	223,287
50	-0,167702	222,748	-0,167445	222,368	-0,167032	221,751	-0,167956	222,677
51	-0,166491	223,535	-0,166602	223,364	-0,165463	221,738	-0,165952	222,019
52	-0,164353	221,122	-0,164709	221,508	-0,164410	221,559	-0,164746	221,403
53	-0,168687	223,409	-0,167851	222,166	-0,168113	222,535	-0,169128	223,613
54	-0,168756	225,018	-0,169520	226,156	-0,168488	224,630	-0,169002	225,173
55	-0,167510	224,171	-0,167693	224,793	-0,166538	222,396	-0,167113	223,397
56	-0,168851	224,087	-0,169438	224,405	-0,168320	223,123	-0,169150	224,070
57	-0,164987	223,504	-0,165493	223,986	-0,164595	223,475	-0,165544	224,294
58	-0,171722	225,103	-0,171699	225,282	-0,173592	227,096	-0,170794	223,681
59	-0,167981	224,028	-0,167571	223,991	-0,168290	225,035	-0,168046	224,088
60	-0,166682	224,032	-0,167261	224,766	-0,165779	222,713	-0,166201	223,317

Table G.2: Calibration constants for channels 1 to 4 of the DCU2 in the CB

Ctrl Board ID	DCU3-Ch3 Mx	K	gain DCU1	gain DCU2	gain DCU3
1	-0,0000274024	0,0493831	0,000573382	0,000558582	0,000463250
2	-0,0000275454	0,0490627	0,000564733	0,000563629	0,000464031
3	-0,0000273115	0,0491671	0,000563340	0,000564314	0,000461770
4	-0,0000272700	0,0494253	0,000570421	0,000562856	0,000456071
5	-0,0000273650	0,0493952	0,000558923	0,000550825	0,000463010
6	-0,0000272934	0,0494182	0,000563298	0,000562676	0,000461347
7	-0,0000273445	0,0495452	0,000565847	0,000564351	0,000459052
8	-0,0000274164	0,0492072	0,000561892	0,000560035	0,000462490
9	-0,0000276448	0,0491510	0,000558343	0,000564099	0,000465623
10	-0,0000272936	0,0495011	0,000567716	0,000563470	0,000461030
11	-0,0000271492	0,0492183	0,000560520	0,000560090	0,000457622
12	-0,0000272848	0,0495204	0,000564825	0,000568922	0,000459643
13	-0,0000272250	0,0495872	0,000569359	0,000562049	0,000457558
14	-0,0000275282	0,0491959	0,000572457	0,000570618	0,000464490
15	-0,0000276669	0,0494312	0,000561226	0,000551434	0,000467024
16	-0,0000277352	0,0495865	0,000559166	0,000557375	0,000465486
17	-0,0000273168	0,0492639	0,000563334	0,000563060	0,000462956
18	-0,0000275477	0,0495962	0,000557810	0,000551677	0,000465018
19	-0,0000275176	0,0494693	0,000572303	0,000562216	0,000463120
20	-0,0000274923	0,0492930	0,000556784	0,000556284	0,000465602
21	-0,0000271086	0,0493283	0,000558252	0,000545262	0,000455444
22	-0,0000271701	0,0491416	0,000563555	0,000570255	0,000459936
23	-0,0000274173	0,0493359	0,000552254	0,000560445	0,000464092
24	-0,0000274225	0,0498525	0,000570943	0,000558776	0,000460034
25	-0,0000274646	0,0496977	0,000576900	0,000565774	0,000460308
26	-0,0000273008	0,0492807	0,000561866	0,000556153	0,000458632
27	-0,0000274484	0,0492781	0,000561034	0,000570371	0,000460884
28	-0,0000278326	0,0493883	0,000564849	0,000557500	0,000470088
29	-0,0000271940	0,0493573	0,000576209	0,000572520	0,000458725
30	-0,0000271316	0,0493681	0,000561503	0,000564479	0,000455458
31	-0,0000274494	0,0496623	0,000567427	0,000558144	0,000461187
32	-0,0000272731	0,0499571	0,000567815	0,000566746	0,000456984
33	-0,0000274058	0,0496450	0,000555147	0,000566348	0,000459177
34	-0,0000275054	0,0496291	0,000562170	0,000557560	0,000462934
35	-0,0000275678	0,0497838	0,000566639	0,000565782	0,000462714
36	-0,0000271239	0,0492116	0,000560194	0,000545738	0,000456667
37	-0,0000273894	0,0495609	0,000569284	0,000554293	0,000460250
38	-0,0000272425	0,0496264	0,000568771	0,000564426	0,000453865
39	-0,0000275036	0,0491700	0,000563932	0,000563757	0,000464210
40	-0,0000273296	0,0493647	0,000551263	0,000561563	0,000461792
41	-0,0000274837	0,0494123	0,000566297	0,000562781	0,000463747
42	-0,0000271346	0,0494712	0,000563025	0,000561934	0,000457784
43	-0,0000272815	0,0492069	0,000562065	0,000559533	0,000461406
44	-0,0000270635	0,0491179	0,000565808	0,000556154	0,000456846
45	-0,0000274075	0,0496268	0,000572130	0,000558861	0,000461560
46					
47	-0,0000272420	0,0496527	0,000564965	0,000560203	0,000458901
48	-0,0000277836	0,0494635	0,000580145	0,000566378	0,000467368
49	-0,0000272397	0,0495154	0,000576125	0,000571630	0,000459160
50	-0,0000277016	0,0494492	0,000570729	0,000565197	0,000465489
51	-0,0000269551	0,0491569	0,000561090	0,000560454	0,000452567
52	-0,0000278834	0,0491992	0,000558925	0,000555153	0,000470942
53	-0,0000275125	0,0493990	0,000554882	0,000568276	0,000462882
54	-0,0000274918	0,0494599	0,000564827	0,000569943	0,000463593
55	-0,0000276014	0,0497875	0,000570494	0,000564116	0,000463308
56	-0,0000276120	0,0494725	0,000562831	0,000569943	0,000465357
57	-0,0000272863	0,0493692	0,000563081	0,000557175	0,000461756
58	-0,0000274751	0,0497310	0,000573767	0,000580093	0,000461746
59	-0,0000273322	0,0492961	0,000556634	0,000566672	0,000460589
60	-0,0000271009	0,0495274	0,000564007	0,000561651	0,000455918

Table G.3: Calibration constants for channels 3 and 2 of the DCU3 in the CB and gains of the three DCUs

APPENDIX H   MHX2000 HUMIDITY SENSOR
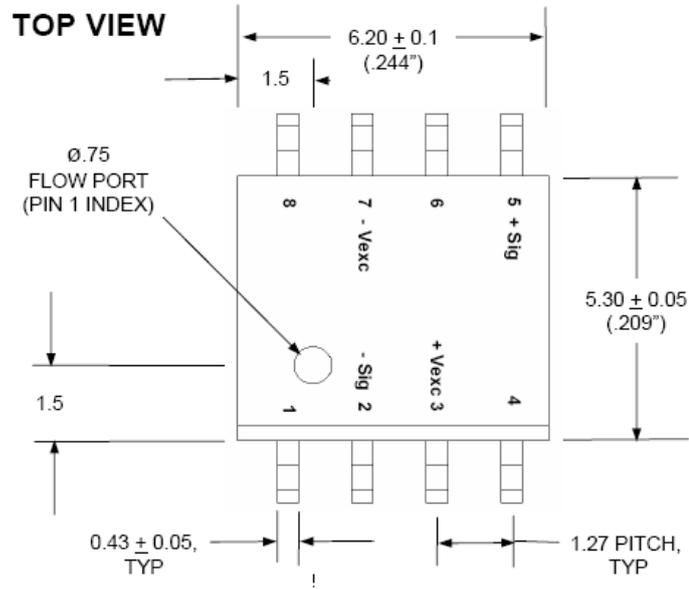
Figure H.1: HMX2000 drawing

APPENDIX I   DCU HARDWARE DESCRIPTION

Reg Nº	Reg Name	Default content (after reset)	Description
0	CREG	0x00	<7> START: "1" to start an A to D conversion <6> RESET: "1" to reset the r/w registers <5> HIRES: must be "0" <4> TSON: must be "0" <3> POLARITY: "1" in LIR Mode, "0" in HIR Mode <2:0> CHANNEL: Selects the ADC input channel
1	SHREG	0x00	<7> IDLE <6> SEU Error <5:4> UNUSED <3:0> DATA<11:8>
2	AREG	0x00	<7> AD_SET Writes into results register, must be written with 0 <6> AD_RESET Writes into results register, must be written with 0 <5> UNUSED <4> UNUSED <3:0> FSM_STATE Used to test control logic of AD, must be written with 0
3	LREG	0x00	<7:0> DATA<7:0 >
4	TREG	0x00	<7> CNT_TEST Used to test AD counter, must be written with 0 <6> ADCFSM_TEST Used to test AD control logic, must be written with 0 <5> UNUSED <4> BGON Selects the bandgap reference, must be written with 1 <3> FSMODE Activates single scale mode, must be written with 0 <2> DTPION Stops AD counter on external trigger, must be written with 0 <1> DTPO_SEL Selects digital test point, must be written with 0 <0> ATP_SEL Selects analog test point, must be written with 0
5	IDLREG		<7:0> CHIPIDL<7:0 >
6	IDMREG		<7:0> CHIPIDM<15:8 >
7	IDHREG		<7:0> CHIPIDH<23:16 >

Table I.1: DCU internal registers

APPENDIX J HMX2000 SENSORS CALIBRATION

$$Vout = offset + sT \times (T\text{-}T_0) + sT_2 \times (T\text{-}T_0)^2 + sH \times (H\text{-}H_0) + sH_2 \times (H\text{-}H_0)^2 + sTH \times (T\text{-}T_0) \times (H\text{-}H_0)$$

$T_0 = 0$ [ºC]

$H_0 = 20$ [%RH]

J.1   FIT TO A FIRST ORDER FUNCTION (A PLANE)

Sensor	offset	sT	sT2	sH	sH2	sTH
43-0652	-7,042	0,1563	0	0,3543	0	0
43-0604	-10,290	0,1603	0	0,3469	0	0
43-0591	-14,260	0,1779	0	0,3375	0	0
43-0595	-13,600	0,1756	0	0,3389	0	0
43-0661	-15,940	0,1608	0	0,3427	0	0
43-0584	-6,261	0,1205	0	0,2866	0	0
43-0747	-8,702	0,1730	0	0,3507	0	0
43-0730	-13,700	0,1708	0	0,3443	0	0
43-0767	-10,160	0,1729	0	0,3383	0	0
43-0687	-11,770	0,1674	0	0,3264	0	0
43-0766	-15,250	0,1442	0	0,3045	0	0
43-0763	-11,960	0,1677	0	0,3315	0	0
43-0589	-15,060	0,1640	0	0,3371	0	0
43-0669	-14,080	0,1779	0	0,3538	0	0
43-0764	-12,790	0,1675	0	0,3191	0	0
43-0704	-16,480	0,1851	0	0,3289	0	0
43-0750	-13,560	0,1983	0	0,3437	0	0
43-0751	-7,561	0,1622	0	0,3190	0	0
43-0651	-15,960	0,1756	0	0,3472	0	0
43-0630	-10,960	0,1871	0	0,3145	0	0
43-0640	-15,870	0,1716	0	0,3181	0	0
43-0700	-16,230	0,1763	0	0,3532	0	0
43-0761	-15,400	0,1666	0	0,3516	0	0
43-0601	-12,260	0,1684	0	0,3323	0	0
43-0587	-17,760	0,1736	0	0,3369	0	0
43-0697	-10,290	0,1451	0	0,3303	0	0
43-0611	-8,980	0,1853	0	0,3204	0	0
43-0754	-6,499	0,1751	0	0,3406	0	0
43-0698	-6,624	0,1497	0	0,3265	0	0
43-0642	-9,085	0,1499	0	0,2948	0	0

Table J.1: calibration curves for a 1st order function of the humidity sensors

J.2   FIT TO A SECOND ORDER FUNCTION

Sensor	offset	sT	sT$_2$	sH	sH$_2$	sTH
43-0652	-6,6440	0,1634	-0,005801	0,3542	-0,001877	0,00033970
43-0604	-9,9130	0,1673	-0,005676	0,3469	-0,001689	0,00012820
43-0591	-13,8900	0,1848	-0,005651	0,3375	-0,001617	0,00021000
43-0595	-13,2300	0,1826	-0,005767	0,3387	-0,001392	0,00040580
43-0661	-15,5700	0,1675	-0,005471	0,3426	-0,001695	0,00029860
43-0584	-5,8970	0,1276	-0,005783	0,2868	-0,001377	-0,00004279
43-0747	-8,3190	0,1798	-0,005636	0,3507	-0,001769	0,00004342
43-0730	-13,3500	0,1774	-0,005407	0,3441	-0,001521	0,00046670
43-0767	-9,7540	0,1807	-0,006291	0,3384	-0,001672	0,00005043
43-0687	-11,4200	0,1740	-0,005397	0,3264	-0,001547	0,00001246
43-0766	-14,9000	0,1504	-0,005035	0,3046	-0,001651	0,00004358
43-0763	-11,6100	0,1746	-0,005651	0,3313	-0,001401	0,00055930
43-0589	-14,6800	0,1711	-0,005798	0,3371	-0,001677	0,00014750
43-0669	-13,6900	0,1851	-0,005859	0,3539	-0,001767	-0,00006417
43-0764	-12,4400	0,1737	-0,005017	0,3192	-0,001693	-0,00000686
43-0704	-16,1200	0,1919	-0,005509	0,3286	-0,001430	0,00055960
43-0750	-13,2300	0,2040	-0,004666	0,3437	-0,001637	0,00007034
43-0751	-7,1930	0,1689	-0,005488	0,3191	-0,001649	-0,00001017
43-0651	-15,5900	0,1822	-0,005382	0,3473	-0,001829	-0,00007879
43-0630	-10,6400	0,1935	-0,005230	0,3141	-0,001173	0,00077790
43-0640	-15,5200	0,1783	-0,005410	0,3180	-0,001488	0,00024510
43-0700	-15,8300	0,1836	-0,005897	0,3534	-0,001811	-0,00012480
43-0761	-14,9900	0,1742	-0,006136	0,3518	-0,001862	-0,00018410
43-0601	-11,9100	0,1752	-0,005520	0,3322	-0,001377	0,00028760
43-0587	-17,3700	0,1809	-0,005901	0,3371	-0,001634	-0,00019350
43-0697	-9,9120	0,1519	-0,005507	0,3305	-0,001726	-0,00022760
43-0611	-8,6650	0,1912	-0,004831	0,3199	-0,001320	0,00089520
43-0754	-6,1640	0,1812	-0,004879	0,3405	-0,001581	0,00027690
43-0698	-6,2680	0,1561	-0,005173	0,3267	-0,001697	-0,00027660
43-0642	-8,7480	0,1560	-0,004916	0,2951	-0,001579	-0,00039440

Table J.2: calibration curves for 2nd order function of the humidity sensors

APPENDIX K   THE DCU CONFIGURATION AND READOUT

After powering up the electronics the DCUs must be reset and initialized by writing and setting the DCU registers to the right value (see description in Appendix XX). The 'SpecsServer' eases the task by hiding procedure to the users who just has to send the reset and initialize command and the server automatically performs the operation that consists of two writings to two different registers.

In the same way the 'SpecsServer' permits creating services that perform a DCU channel acquisition. By creating this type of service the users just needs to ask for service value update and the server will cope with the acquisition procedure. This procedure consists of writing to the DCU control register a value that indicates the channel that has to be read, the mode (LIR or HIR) and that launches the acquisition. After that, the digitized value is stored in two of the registers, so they have to be read out and then the valued is obtained by merging the reading of both registers. Implementing this whole procedure in the server eases the monitoring procedure and the number of operations and service requests that the client has to perform is reduced. As a consequence the network occupation is fewer and CPU computation load is moved to server side and consequently fairly reduced because the client commands run as an interpreted language while the server is native code.

APPENDIX L   IT SERVICE BOX TO DETECTOR MAPPING

STATIONS 1, 2 and 3 ACCESS

BOTTOM BOX

ACCESS BOX

BOTTOM BOX SVCE BOXES

ACCESS BOX SVCE BOXES

STATION 1 and 3 CRYO

CRYO BOX

TOP BOX

CRYO BOX SVCE BOXES

TOP BOX SVCE BOXES

STATION 2 CRYO

CRYO BOX

TOP BOX

CRYO BOX SVCE BOXES

TOP BOX SVCE BOXES
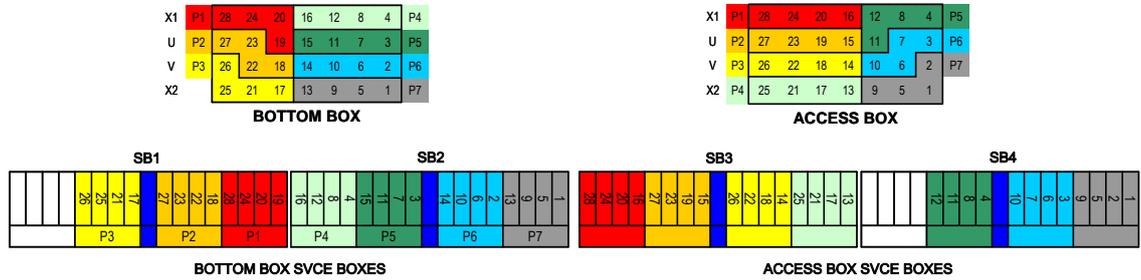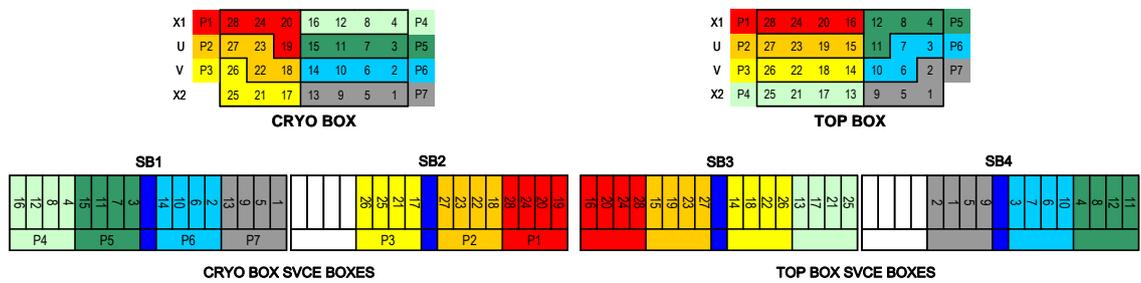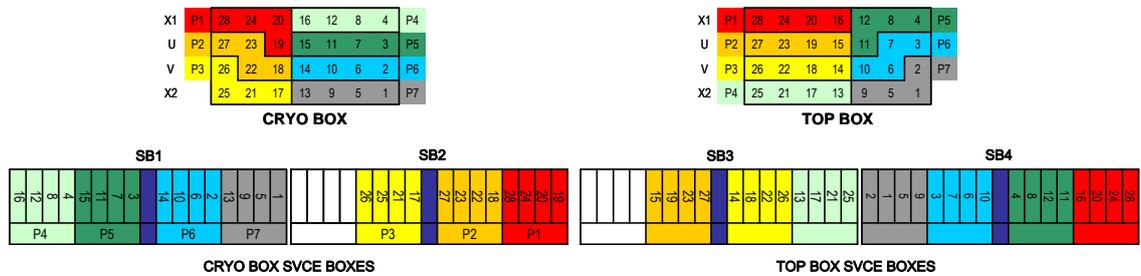
APPENDIX M TT SERVICE BOX TO DETECTOR MAPPING

ACCESS TOP

ACCESS BOTTOM

CRYO TOP

DETECTOR

SERVICE BOXES

CRYO BOTTOM

DETECTOR

SERVICE BOXES
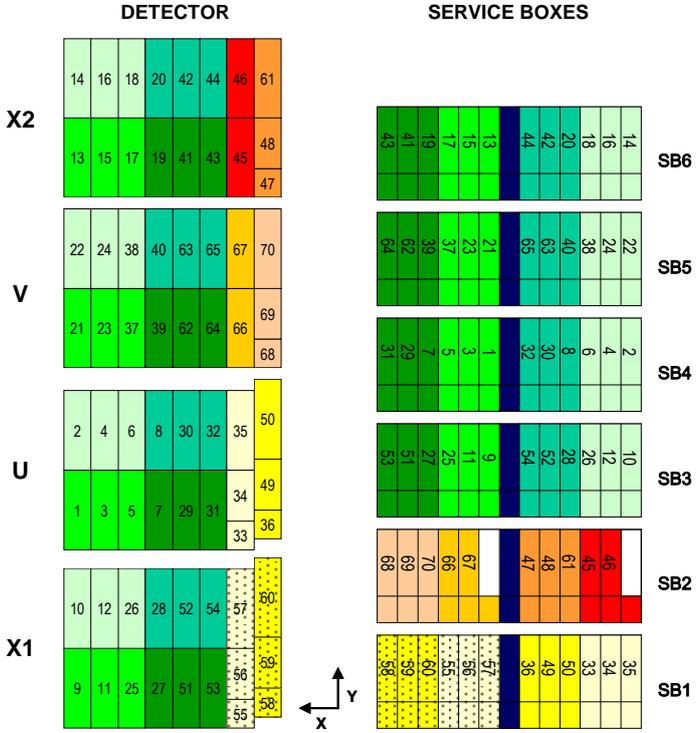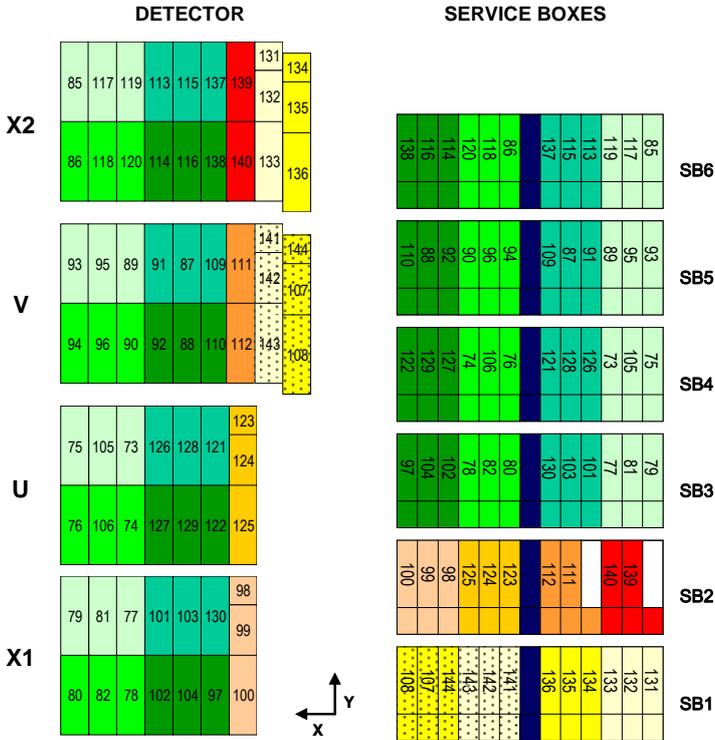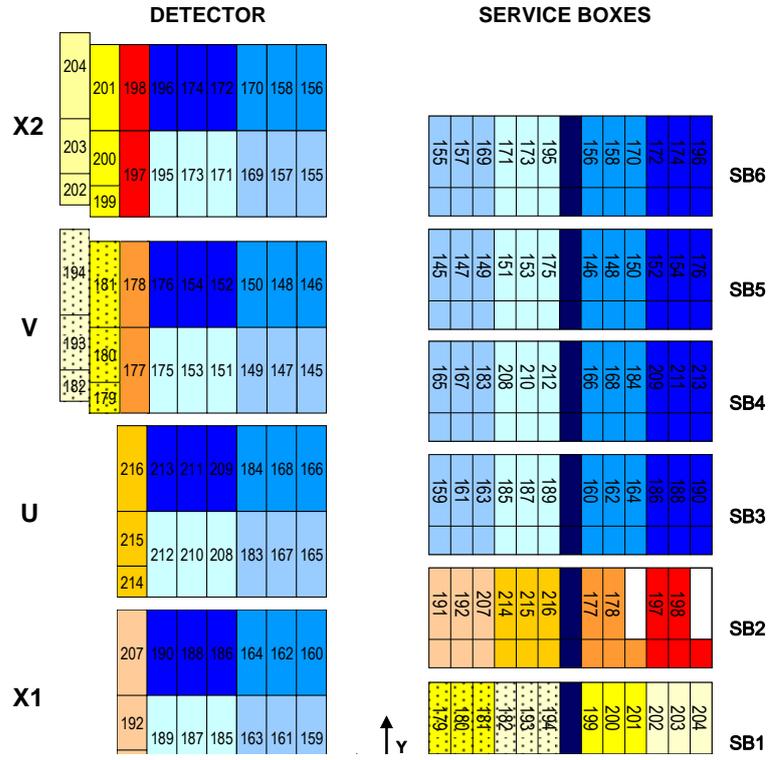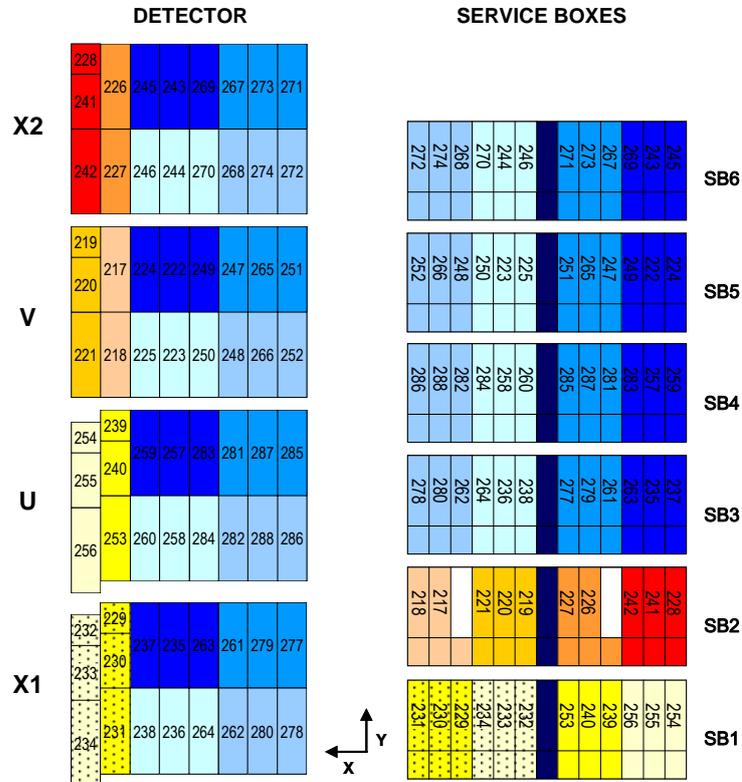
APPENDIX N   HARDWARE SPECIFICATIONS

Reg Nº	Reg Name	Range	Res LSB	Nominal value	Setting reg content	Description
0	*Itp*	0 - 2mA	7.8 µA	0 µA	0x00	test pulse bias current
1	*Ipre*	0 - 2mA	7.8 µA	600 µA	0x4C	preamplifier bias current
2	*Isha*	0 - 2mA	7.8 µA	80 µA	0x0A	shaper bias current
3	*Ibuf*	0 - 2mA	7.8 µA	80 µA	0x0A	front-end buffer bias current
4	*Vfp*	0 - 2.5V	9.8mV	0mV	0x00	preamplifier feedback voltage
5	*Vfs*	0 - 2.5V	9.8mV	0mV	0x00	shaper feedback voltage
6	*Icomp*	0 - 2mA	7.8 µA	40 µA	0x05	comparator bias current
7	*Ithdelta*	0 - 2mA	7.8 µA	—	—	current defining incremental comparator hreshold
8	*Ithmain*	0 - 2mA	7.8 µA	—	—	current defining common comparator threshold
9	*Vrc*	0 - 1.25	4.9mV	0mV	0x00	comparator RC time constant
10	*Ipipe*	0 - 2mA	7.8 µA	100 µA	0x0D	pipeamp bias current
11	*Vd*	0 - 2.5V	9.8mV	1 275mV	0x82	pipeamp reset potential
12	*Vdcl*	0 - 2.5V	9.8mV	1 030mV	0x69	pipeamp reference voltage
13	*Ivoltbuf*	0 - 2mA	7.8 µA	160 µA	0x14	pipeamp buffer bias current
14	*Isf*	0 - 2mA	7.8 µA	200 µA	0x1A	multiplexer buffer bias current
15	*Icurrbuf*	0 - 2mA	7.8 µA	800 µA	0x66	output buffer bias current
16	*Latency*	10 - 160	—	160	0xA0	trigger latency
17	*ROCtrl*	—	—			readout control
18	*RclkDiv*	0 - 255	—	0	0x00	ratio between Rclk and Sclk
19	*CompCtrl*	—	—			comparator control
20	*CompChTh*	0 - 31	—	—	—	comparator channel threshold shift register implementation.
21	*CompMask*	—	—	0	0x00	comparator mask shift register implementation
22	*TpSelect*	—	—	0	0x00	test pulse selection shift register implementation
23	*SEUcounts*	0 - 255	—	—	—	sum of Single Event Upsets

Table N.1: Registers 0−15 are bias registers for the analogue stages; register 16 defines the latency which has to be ≥ 10 and ≤ 160 for reliable chip operation (a change of the latency register is only made effective by applying a reset); the registers 17 and 19 select the chip's mode of operation; registers 20−22 (CompChTh, CompMask, TpSelect) are operated as shift-registers, being CompMask and TpSelect a 128-bit register each, segmented in 16 8-bit registers, and CompChTh establishes a 1024 (= 128 × 8) bit register.

Reg Nº	Reg Name	Default content (after reset)	Description
0	Fine Delay 1	0x00	Defines the clock1 phase in steps of 104 ps between 0 and 25 ns using a special encoding
1	Fine Delay 2	0x00	Defines the clock2 phase in steps of 104 ps between 0 and 25 ns using a special encoding
2	Coarse Delay	0x00	<3:0> Coarse delay 1; <7:4> Coarse delay 2
3	Control	0x93	<0> Enable Bunch Counter operation <1> Enable Event Counter operation <2> SelClock40Des2; <3> Enable Clock40Des2 output <4> Enable ClockL1Accept output <5> Enable Parallel output bus; <6> Enable Serial B output; <7> Enable (non-deskewed) Clock40 output
8	Single error count <7:0>	0x00	Counter of single bit errors recognised by the receiver's Hamming decoder
9	Single error count <15:8>	0x00	Counter of single bit errors recognised by the receiver's Hamming decoder
10	Double error count <7:0>	0x00	Number of double bit Hamming errors and frame error
11	SEU error count <15:8>	0x00	Counter of SEUs
16	ID<7:0>	0x00	Chip ID
17	MasterModeA<1:0>, ID<13:8>	0x00	TTCrx operational mode + chip ID
18	MasterModeB<1:0>, I2C_ID <5:0>	0x00	TTCrx operational mode + I2C ID
19	Config 1	0x1A	<2:0> dll_isel<2:0> Selects DLL current; <5:3> pll_isel<2:0> Selects PLL current <6> dll_sel_aux_1 Selects test input forphase shifter 1 <7> dll_sel_aux_2 Selects test input for phase shifter 2
20	Config 2	0x84	<2:0> mux_select<2:0>: Selects test outputs <3> cf_sel_test_PD: Selects external test signal for enabling the PLL phase detector. <4> cf_sel_inputA When 0 selects inputs from optical link, otherwise test_in<3,4> <5> cf_PLL_aux_reset: Assert PLL test reset line <6> cf_DLL_aux_reset: Assert DLL test reset line <7> cf_en_check_machineA: Enables Hamming checkmachine
21	Config 3	0xA7	<2:0> frequ_check_period<2:0>: Stop frequency detection phase after 2^(n+4) cycles without "frequ_low" detected <3> cf_dis_INITfaster: If 1 disables automatic frequ. increase after PLL reset <4> cf_dis_watchdog: If 1 disables watchdog circuit <5> cf_en_Hamming: Enables Hamming error detection/correction on incoming data stream <6> cf_en_testIO: Enables Test Input/Outputs <7> cf_en_check_machineB: Enables Hamming check machine
22	Status	0xE0	<3:0> - Always zero <4> auto_reset_flag: A 1 indicates that an automatic reset has occurred due to a timeout condition in the watchdog circuit <5> frame_synch: A 1 indicates that channel B is synchronized to the data stream <6> dll_ready: A 1 indicates that the High-Resolution phase shifters are working properly <7> pll_ready: A 1 indicates that the clock and data recovery circuit is locked on the incoming data stream
24	Bunch counter <7:0>	0x00	It is incremented by the received clock signal.
25	Bunch counter <15:8>	0x00	It is incremented by the received clock signal.
26	Event counter <7:0>	0x00	It is incremented upon reception of a trigger accept signal in channel A
25	Event counter <15:8>	0x00	It is incremented upon reception of a trigger accept signal in channel A
26	Event counter <23:16>	0x00	It is incremented upon reception of a trigger accept signal in channel A

Table N.2: TTCrx registers.

Reg Nº	Reg Name	Default content (after reset)	Description
0	CR0	0x00	<6> Enable channel output <5:0> defines the channel 0 delay
1	CR1	0x00	<6> Enable channel output <5:0> defines the channel 1 delay
2	CR2	0x00	<6> Enable channel output <5:0> defines the channel 2 delay
3	CR3	0x00	<6> Enable channel output <5:0> defines the channel 3 delay
4	CR4	0x00	<6> Enable channel output <5:0> defines the channel 4 delay
5	GCR	0x00	<6> IDLL: force the resynchronization of the DLL <5:3>: — <1:0> Mode: selects the clock frequency

Table N.3: Delay25 chip registers.

Reg Nº	Reg Name	Default content (after reset)	Description
0	REGOUT_MSB	0x0000	Write in Bit #i to control output pin #i or read in Bit #i to read the status of input pin #i. The
1	REGOUT_LSB	0x0000	
2	CONFREGOUT_MSB	0x0000	Set bit #i to 1 to configure pin #i in output mode (itis configured in input mode – value 0 -- by default)
3	CONFREGOUT_LSB	0x0000	
4	MEZZA_CTRL	0x0000	<10> *SCLSpyControl:Disable* SCL Spy Control. <9> *CalibChannelBDisable*: Disable decoding of Calibration Channel B command. <8> *L0ChannelBDisable:Disable* decoding of L0 Reset Channel B command. <7> *BckChannelBDisable:Disable* use of broadcast for Channel B decoding. <6> *EEPROMOE:*: EEPROM OE (Internal Use). <5> *EEPROMSerEn*: EEPROM SerEn and CE (Internal Use). <4> *ClkDivFact*: Divide the mezzanine clock by 2. <3> *StadynModeBit*: Mode of control of I2C_DE and JTAG_DE (1→static mode, 0 → dynamic mode) <2> *CmdOscillatorBit*: Source of the mezzanine clock (0 → local oscillator, 1→use external clock) <1> *OutputExternalResetBit*: Control of the reset pin of the mezzanine. <0> *GlobalSoftwareResetBit*: Global reset of the mezzanine.
5	MEZZA_STAT	0x0000	<9> *MastslaveBit*: Master/Slave bit: if 0, Mezzanine in Master mode, if 1, in slave mode. <8> *MSResistorBit*: Master/Slave resistor status: if 0, resistor, if 1, no resistor.
6	BUS_SEL	0x0000	In static mode (set by the *Stadyn_mode* bit of the control register), control i2cjtag_de lines.
7	INT_VECT	0x0000	<10> *IO7InterruptBit*: Mask or active I/O 7 interrupt. <9> *IO6InterruptBit*: Mask or active I/O 6 interrupt. <8> *IO5InterruptBit*: Mask or active I/O 5 interrupt. <7> *IO4InterruptBit*: Mask or active I/O 4 interrupt. <6> *IO3InterruptBit*: Mask or active I/O 3 interrupt. <5> *IO2InterruptBit*: Mask or active I/O 2 interrupt. <4> *IO1InterruptBit*: Mask or active I/O 1 interrupt. <3> *IO0InterruptBit*: Mask or active I/O 0 interrupt. <2> *I2CAckInterruptBitMask* or active I2C acknowledge error. <1> *ChecksumBit* Mask or active Checksum Error. <0> *UserInterruptBit* Mask or active User Interrupt.
8	DATE_PROM	0x0000	Date of the PROM in format MMDD.
9	DEFINT_VECT	0x0000	Same as INT_VECT
10	CHANB_DELAY	0x0000	Delay Calibration pulse type 0 of Channel B with 25 ns steps.

Table N.4: SPECS slave internal registers

Reg Nº	Reg Name	Default content (after reset)	Description
0	*Config 0*	0x33	<4:0> *Wait_time*: Defines the time to wait between the "LOCKED" state and "READY" state <7:5> *Loss_of_lock_time*: Defines the number of erroneous cycles the chip tolerates before it goes into the "OUT-OF-LOCK" state
1	*Config 1*	0x1F	<3:0> *PLL_lock_time*: Defines the time between the OUT-OF-LOCK state and the LOCKED state. <4> *en_soft_loss_of_lock*: If 1 then the GOL can tolerate a number of erroneous cycles defined by *Loss_of_lock* time, otherwise the state goes immediately to OUT-OF-LOCK after the first error. <5> *en_loss_of_lock_count*: If 1 then one TX_LOLC cycle is inserted between LOCKED and READY, where the number of loss-of-lock events is transmitted as a 16-bit data word. <6> *en_force_lock*: [Disables lock state machine if 1. Only used for testing/debugging.] <7> *en_self_test*: If 1, a running 16 bit counter generates transmission data
2	*Config 2*	0x10	<4:0> *PLL_current*: Defines the charge-pump current of the internal phase-locked loop (PLL). <6:5> *test_sel*: [Selects signal to appear on test_analog pad. Only used for testing/debugging.] <7> *en_flag*: Enables flag bits in G-Link mode
3	*Config 3*	0x20	<6:0> *LD_current / driver_strength:* Defines the bias current for the Laser Driver (LD) or the strength of the 50 Ohm line driver <7> *use_conf_regs:* When 1, the content of the Configuration registers 1 and 2 are used to define the value for *PLL_current* and *LD_current*. If 0, the values are derived from the encoded values in the pads.
5	*Status 0*	0x00	Number of "loss-of-lock" events since last reset.
6	*Status 1*	—	<7:6> *link_control_state_A*: Current state of link initialisation logic A <5:4> *link_control_state_B*: Current state of link initialisation logic B <3:2> *link_control_state_C*: Current state of link initialisation logic C <1> *conf_glink*: Reads value on *conf_glink* pin. If 1, then the chip is configured on G-Link mode, otherwise in Ethernet mode. <0> *conf_wmode16*: Reads value on conf_wmode16 pin. If 1then the chip accepts 16-bit wide data and transmits with 800 Mbit/s, otherwise 32 bit data and 1.6 Gbit/s speed.

Table N.5: GOL registers

APPENDIX N: HARDWARE SPECIFICATIONS

Control Board I/O register assignment[1]:
Slave1 bit-signal assignment:
21) TTCrq RESET (O) → resets the TTCrx.
22) TTCrq_QPLL RESET (O) → resets the TTCrq QPLL.
23) /QPLLs reset (O) → resets the all the QPLLs of the DBs in a service box. (Negative logic).
24) RST_GOLs (O) → resets all the GOLs in a SB.
Slave2 bit-signal assignment:
2) L0ACCEPT_DE_Ctrl (O) → Enables the *L0ACCEPT* LVDS driver.
7) DRV_DOWN_EN (O) → Enables the output of the *Clock40Des2* LVDS receiver.
12) DRV_UP_EN (O) → Enables the *Clock40Des1, L0ACCEPT …* LVDS receiver ouput.
15) TTCrqREADY (I) → Signal that indicates whether there is TFC signal at the TTCrq input
16) TTCrqLOCKED (I) → Indicates whether the TTCrq QPLL is locked or not.
17) TTCrqERROR (I) → Indicates if the TTCrq QPLL is in error.
Digitizer Board DCU channels:
Channel 1 → Dig Board QPLL status.

Table N.6: Control Board I/O line and Digitizer Board DCU channel assignment for DAQ purposes

1 Between parentheses is indicated whether the signal is an input (I) or output (O) line.

APPENDIX O   LHCB ST COMPUTER'S ASSIGNMENT

Computer name	IP address	Description
itdaq01.lbdaq.cern.ch	10.130.39.5	IT PC for: TELL1s DNS;TELL1s control project
itdaq02.lbdaq.cern.ch	10.130.20.25	IT PC for: SPECS masters; SPECS DNS; Svce Boxes control project
ttdaq01.lbdaq.cern.ch	10.130.20.15	TT PC for: TELL1s DNS;TELL1s control project
ttdaq01.lbdaq.cern.ch	10.130.20.28	TT PC for: SPECS masters; SPECS DNS; Svce Boxes control project

Table O.1: Server PCs

Computer name	IP address	Description
ittell*XX*.lbdaq.cern.ch	10.130.33.*XX*	IT TELL1 CCPC, *XX* from 01 to 42
tttell*XX*.lbdaq.cern.ch	10.130.34.*XX*	TT TELL1 CCPC, *XX* from 01 to 48

Table O.2: CCPCs IP addresses

Computer name	MAC address	Description
ittell*XX*.lbdaq.cern.ch	0x[00:CC:BB:13:XX:YY]	IT TELL1 CCPC, *XX* from 01 to 42 and *YY* is replaced by the port number 0..3
tttell*XX*.lbdaq.cern.ch	0x[00:CC:BB:10:XX:YY]	TT TELL1 CCPC, *XX* from 01 to 48 and *YY* is replaced by the port number 0..3

Table O.3: CCPCs ethernet addresses

APPENDIX P   THE DAQ CONFIGURATION

P.1   FRONT END CONFIGURATION

After powering up the FE DAQ electronics and having set the SPECS slaves I/O lines to the right mode operational mode several settings must be configured in the front-end:

- First, the I/O lines listed in the table XX of Appendix XX need to be programmed with the right values.

- The TTCrx has two programmable clock phase generators named *Clock40Des1* and *Clock40Des2* used to generate fine the phase delays within the 40 MHz clock cycle. It has two coarse delay pipelines that affect the main TTC signals (clock included). Due to specific restrictions in LHCb caused by the way in which the short broadcast is encoded, it is mandatory that the TTCrx is programmed such the broadcast bits are all synchronized to the *Clock40Des1*.

- The *Clock40Des1* and *Clock40Des2* phase values must be set in order to define the proper sampling point of the Beetles and ADCs respectively. The ADCs clock phase will depend on the data cable length (from DB to hybrid). The coarse delay value will define the synchronization of the detector with respect to the particle bunch.

- The SPECS slaves have to be configure to use the external clock, that is, the *Clock40Des1*.

- The Delay25 chip channel outputs must also be enabled otherwise the TTC signals won't be distributed to the FE and the delay values need to be set such no setup and hold timing violation takes place in the FE. These values should in principle be the same for all Control Boards.

- The SPECS channel B decoding needs to be enabled and the calibration output disable in case we do not plan to run in testing mode.

- Once all the previous is done, the last thing is resetting the Digitizer Board QPLLs and programming the Beetles and GOLs with the desired settings.

P.2   THE TELL1 CONFIGURATION

There are two ways for configuring the TELL1: using a configuration file that is loaded using a command shell command or using the "ccserv". The "ccserv" is the chosen option for remote control, so it must be always running and configured to use as DNS the IT or TT control computer.

The amount of configurable TELL1 options is quite huge and their meaning is explained in any of the configuration files distributed with the software releases. We will describe here the basic settings that one has to appropriately configure in order to have the TELL1 running in "physics" mode. The configurable settings are listed and briefly explained below:

- Gigabit IP and Ethernet protocol parameters: destination IP, destination MAC, source IP, source MAC. and other IP and Ethernet protocol settings

- TTC and processing operation: set the detector ID (ST); configure the TTC source to come from the TFC system; disable external trigger; enable the trigger type and the destination IP to come from the TTC system; disable the internal data generator; set the source id, which is unique for each TELL1; define the class for the zero suppressed, non zero suppressed, pedestal and error banks class.

- IF TELL1 is used to generate the L0 triggers, ie, for debugging, the trigger type must usually be set to work with NZS data.

In the ST specific part: the zero suppression, the header correction, the pedestal, the pedestal auto-update and the common mode suppression must be enabled; the non working optical links and analogue ports must be disabled; the hit thresholds, masks and other values for individual channels must be set as well.

APPENDIX Q THE TEMPERATURE CYCLING BOX SCHEMATICS
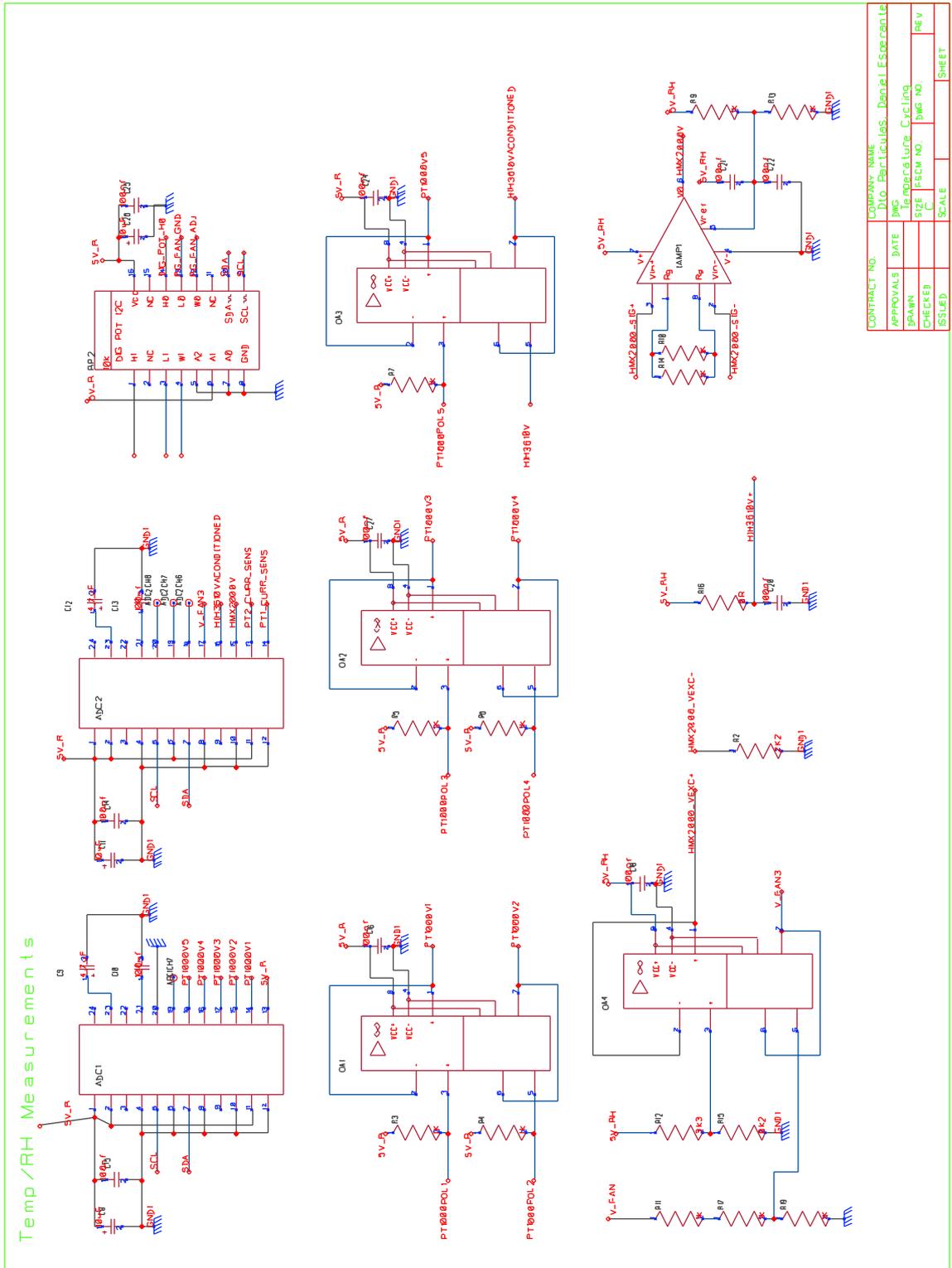
Figure Q.1: The Temperature Cycling Box Schematics

Figure Q.2: The Temperature Cycling Box Schematics

Figure Q.3: The Temperature Cycling Box Schematics

REFERENCES

[1] L. Evans and P. B. (editors), "LHC Machine", *Journal of Instrumentation*, vol. 3, no. 08, p. S08001, 2008.

[2] The ATLAS Collaboration, G. Aad *et al.,* "The ATLAS Experiment at the CERN Large Hadron Collider", *Journal of Instrumentation*, vol. 3, no. 08, p. S08003, 2008.

[3] The CMS Collaboration, S. Chatrchyan *et al.*, "The CMS experiment at the CERN LHC", *Journal of Instrumentation*, vol. 3, no. 08, p. S08004, 2008.

[4] The ALICE Collaboration, K. Aamodt, *et al.*, "The ALICE experiment at the CERN LHC", *Journal of Instrumentation*, vol. 3, no. 08, p. S08002, 2008.

[5] The LHCb Collaboration, A. Augusto Alves *et al.*, "The LHCb Detector at the LHC", *Journal of Instrumentation*, vol. 3, no. 08, p. S08005, 2008.

[6] The TOTEM collaboration, G. Anelli *et al.*, "The TOTEM Experiment at the LHC", *Journal of Instrumentation*, vol. 3, no. 08, S08007, (2008).

[7] The LHCf collaboration, O. Adriani *et al.*, "The LHCf Detector at the LHC", *Journal of Instrumentation*, vol. 3, no. 08, S08006, (2008).

[8] The LHCb Collaboration, "LHCb VELO Technical Design Report", CERN-LHCC/2001-011, CERN, May 2001.

[9] The LHCb Collaboration, "LHCb Reoptimized Detector Design and Performance Technical Design Report", CERN-LHCC/2003-030, CERN, Sept 2003.

[10] The LHCb Collaboration, "LHCb Outer Tracker Technical Design Report", CERNLHCC/2001-024, CERN, Sept 2001.

[11] The LHCb Collaboration, "LHCb RICH Technical Design Report", CERN- LHCC/2000-037, CERN, Sept 2000.

[12] The LHCb Collaboration, "LHCb Calorimeter Technical Design Report", CERN - LHCC/2000-036, CERN, Sept 2000.

[13] The LHCb Collaboration, "LHCb Muon Technical Design Report", CERN- LHCC/2001-010, CERN, May 2001.

[14] The LHCb Collaboration, "LHCb Addendum to the Muon System Technical Design Report", CERN-LHCC/2003-002, CERN, 2003.

[15] The LHCb Collaboration, "LHCb Second Addendum to the Muon System Technical Design Report", CERN-LHCC/2005-012, CERN, 2005.

[16] D. Esperante *et al.*, "LHCb Silicon Tracker DAQ and DCS Online Systems", CERN, Geneva, LHCb-CONF-2009-015, 2009.

[17] S. Lochner, "Development, Optimisation and Characterisation of a Radiation Hard Mixed-Signal Readout Chip for LHCb", Ph.D. dissertation, Max-Planck Inst., Heidelberg Univ., Germany, 2006.

[18] C. Lois, "Silicon sensor probing and radiation studies for the LHCb silicon tracker", *Nucl. Instrum. and Meth.*, vol A568, pp. 277-283, 2006.

[19] A. Vollhardt , "Neutron Irradiation Results for the LHCb Silicon Tracker Data Readout System Components", CERN, Geneva, Tech. Rep., LHCb-2003-049, 2003.

[20] A. Vollhardt, D. Esperante and Y.Ermoline, "Results of proton irradiation of components for the LHCb Silicon Tracker", CERN, Geneva, Tech. Rep., LHCb-2004-037, 2004.

[21] A. Vollhardt *et al.*, "A radiation tolerant fiber-optic readout system for the LHCb Silicon Tracker", Proceedings European Symposium on Semiconductor Detectors, Wildbad Kreuth, Jun 12-16, 2005, LHCb-2005-032.

[22] A. Vollhardt, "An Optical Readout System for the LHCb Silicon Tracker", Ph.D. dissertation, Physik-Institut, Zurich Univ., Switzerland, 2005.

[23] Haefeli et al., "The LHCb DAQ interface board TELL1", *Nucl. Instrum. and Meth.*, vol. A560, pp. 494-502, 2006.

[24] S. M. Schmeling, B. Flockhart, S. Luders, G. Morpurgo, "The Detector Safety System for LHC Experiments", *IEEE Trans. Nucl. Sci.*, vol. 51, no. 3, pp. 521-255, June 2004.

[25] D. Esperante, A. Vollhardt, "Design and development of the Control Board for the LHCb Silicon Tracker", CERN, Geneva, Tech. Rep., LHCb-2007-153, 2007.

[26] Z. Guzik, R. Jacobsson, B. Jost, "Driving the LHCb front-end readout", *IEEE Trans. Nucl. Sci.*, vol. 51, no. 3, pp. 508-512, June 2004.

[27] C. Gaspar et al., "An Integrated Experiment Control System, Architecture, and Benefits: The LHCb Approach", *IEEE Trans. Nucl. Sci.*, vol. 51, no. 3, pp. 513-520, June 2004.

[28] CERN Gigabit Optical Link (GOL) transmitter ASIC, http://proj-gol.web.cern.ch/proj-gol/

[29] CERN Quartz Phase-Locked Loop (QPLL) ASIC, http://proj-qpll.web.cern.ch/proj-qpll/

[30] CERN DCUF25 chip, http://cmstrackercontrol.web.cern.ch/CMSTrackerControl/documents/Magazzu/DCUF_User_Manual_v3.0.pdf

[31] D. Breton, D. Charlet, P. Robbe, I. Videau, "SPECS : A serial protocol for experiment control system in LHCb", Proceedings of *International Conference on Accelerator and Large Experimental Physics Control Systems*, Geneva, Switzerland, 10-14 Oct., 2005, WE1.5-4O.

[32] J. Christiansen, A. Marchioro, P. Moreira and T. Toifl, "TTCrx Reference Manual", http://ttc.web.cern.ch/TTC/TTCrx_manual3.11.pdf

[33] CERN Web Services, ST Microelectronics LHC Radiation Hardened Voltage Regulator, http://lhc-voltage-regulator.web.cern.ch/lhc-voltage-

[34] R. Jacobsson, B. Jost, Z. Guzik, "Readout supervisor design specifications", CERN, Geneva, Tech. Rep., LHCb-2001-012, 2001.

[35] F. Fontanelli et al., "Embedded Controllers for Local Board-Control", *IEEE Trans. Nucl. Sci.*, vol. 53, no. 3, pp. 936-940, June 2006.

[36] V. Radeka, Veljko, "Shielding and Grounding in Large Detectors", Proc. 4th *Workshop on Electronics for LHC Experiments*, CERN/LHCC/98-36, pp. 14-18 (1998).

[37] V. Bobillier, J. Christiansen, R. Frei, "Grounding, Shielding and Power Distribution in LHCb", CERN, Geneva, Tech. Rep, LHCb-2004-039, 2004.

[38] C. Bauer, D. Esperante, R. Frei, U. Straumann, P. Vázquez, A. Vollhardt, "Grounding, Shielding and Power Distribution for the LHCb Silicon Tracking", CERN, Geneva, Tech. Rep, LHCb-2004-101, 2005.

[39] Gafner, A. Vollhardt, "Characterization and sample testing of the LHC4913 positive voltage regulator for the LHCb Silicon Tracker", CERN, Geneva, Tech. Rep, LHCb-2003-128, 2003.

[40] LHC4913 series regulator datasheet. STMicroelectronics.

[41] P. Moreira; "TTCrq Manual"; version 1.5. CERN-EP/MIC. http://proj-qpll.web.cern.ch/proj-qpll/images/manualTTCrq.pdf

[42] http://ttc.web.cern.ch . Timing, Trigger and Control (TTC) Systems for the LHC.

[43] G. Magazzu, A. Marchioro, and P. Moreira, "The Detector Control Unit: an ASIC for the monitoring of the CMS Silicon Tracker", *IEEE Trans. on Nuclear Science*, vol. 51, issue 4,

pp. 1333-1336; "DCUF User Guide", CERN, http://cmstrackercontrol.web.cern.ch/cmstrackercontrol/documents/Magazzu/DCUF_User_Manual_v3.0.pdf

[44] L6225, "DMOS DUAL FULL BRIDGE DRIVER" datasheet. STMicroelectronics.

[45] HMX2000, Hygrotron-MEMS Hygro/Thermal Sensor, www.hygrometrix.net.

[46] Agapito et al., "Instrumentation Amplifiers and Voltage Controlled Current Sources for the LHC Cryogenic Instrumentation", Proceedings of 6th *Workshop for LHC Experiments*, 2000.

[47] Hagerman, Jim, "Two transistors form bidirectional level translator", EDN, Nov 7, 1996, pp. 114.

[48] M. Needham, O. Steinkamp, "Updated channel numbering and readout partitioning for the Silicon Tracker", CERN, Geneva, Tech. Rep., LHCb-2007-137.

[49] MARATON Power Supply System Technical Manual, Wiener 20 July 2006.

[50] H. J. Ziock er al., "Temperature dependence of the radiation induced charge of depletion voltage in silicon PIN detectors", *Nucl. Instr. and Methods*, vol. A342, pp. 96-104, 1994.

[51] F. Lehner, M. Stodulski, "The liquid cooling system of the LHCb Inner Tracker: Design Constraints and Considerations", CERN, Geneva, Tech. Rep., LHCb-2002-066, 2002.

[52] A. Büchler, "Thermal and Mechanical Characterization of the TT Detector for the LHCb Experiment", Master Thesis, Universität Zürich, Switzerland, 2007.

[53] K. Bösinger, F. Lehner, S. Steiner, S. Strässle, "Design, Construction and Thermal Measurements on a Detector Box for the Inner Tracker of the LHCb Experiment", CERN, Geneva, Tech. Rep., LHCb-2002-059, 2002.

[54] D. Sonntag, "Important new values of the physical constants of 1986 vapor pressure formulations based on the ITS-90 and psychrometer formula", Z. Meteorol. 70, pp. 340-344, 1990.

[55] M. Schwerin, A. Tsirou, G. Verdini, M. Weber; "The humidity sensors for the CMS Tracker", CMS internal note.

[56] S. Löchner, M. Schmelling, "The Beetle Reference Manual", CERN, Geneva, Tech. Rep., LHCb-2005-105, 2005.

[57] J. Christiansen, "Requirements to the L0 front-end electronics", CERN, Geneva, Tech. Rep., LHCb-2001-014, 2001.

[58] P. Moreira *et al.*, "GOL Reference Manual Version 1.5", http://proj-gol.web.cern.ch/proj-gol/manuals/gol_manual.pdf

[59] The I2C-Bus Specification, Version 2.1, January 2000, http://www.nxp.com/acrobat_download/literature/9398/39340011.pdf

[60] LHCb TFC system. http://lhcb-online.web.cern.ch/lhcb-online/TFC/default.html

[61] C. Gaspar *et a.l*, "An Integrated Experiment Control System, Architecture and Benefits: the LHCb Approach", *IEEE Transactions on Nuclear Science*, vol. 51, no. 3, June 2004.

[62] C. Gaspar and B. Franek, "Tools for the Automation of Large Distributed Control Systems", *IEEE Trans. Nucl. Sci.*, vol. 53, no. 3, June 2006.

[63] M. Gonzalez-Berges, "The Joint COntrols Project Framework", Proceedings of *Computing in High Energy and Nuclear Physics*, California, March 2003, THGT006.

[64] PVSS-II, http://www.pvss.com

[65] P. Burkimsher and H. Milcent, "Applying Industrial Solutions to the Control of HEP Experiments". Presented at *ICALEPCS* 1999, 1999.

[66] The OPC foundation, http://www.opcfoundation.org/

[67] C. Gaspar, M. Donszelmann, "DIM –A Distributed Information Management System for the DELPHI Experiment at CERN", Proceedings of the 8th *Conference on Real-Time Computer applications in Nuclear, Particle and Plasma Physics*, Vancouver, Canada, June 1993.

[68]   B. Franek and C. Gaspar, "SMI++ Object-Oriented Framework for Designing and Implementing Distributed Control Systems", *IEEE transactions on Nuclear Science*, vol. 52, nº4, August 2005.

[69]   Object Management Group – UML, http://www.uml.org

[70]   J. Barlow, B. Franek, M. Jonker, T. Nguyen, P. Vande Vyvre, and A. Vascotto, "Run control in MODEL: The state manager", *IEEE Trans. Nucl. Sci.*, vol. NS-36, pp. 1549-1553, Oct. 1989.

[71]   P. Aarnio *et al.*, "The DELPHI detector at LEP", *Nucl. Instrum. and Meth.*, vol. A303, pp. 233-276, 1991.

[72]   C. Gaspar, "Hierarchical Controls. Configuration & Operation", CERN, Geneva, 2004.

[73]   F. Calheiros, P. Golonka, F. Varela, "Automating the configuration of the Control Systems for the LHC experiments", presented at *ICALEPCS 2007*, Knoxville, Tennessee, USA, 2007.

[74]   L. Abadie *et al.*, "The LHCb configuration database", presented at 10th *ICALEPCS*, Geneva, Switzerland, 10-14 Oct., MO4A.2-7O.

[75]   F. Alessio *et al.*, "LHCb Online event processing and filtering", International Conference on *Computing in High Energy and Nuclear Physics*, *J. Phys. Conf. Ser.*, vol. 119, part 2, 2008.

[76]   LHCb Experiment Control System, http://lhcb-online.web.cern.ch/lhcb-online/ecs/default.htm

[77]   Pierre-Yves Duval, C. Gaspar, "Guide for ECS FSM design in LHCb sub-detectors", EDMS CERN document, EDMS 655828, https://edms.cern.ch/document/655828

[78]   C. Gaspar, "LHCb ECS Integration Guidelines", EDMS CERN document, EDMS 732486, https://edms.cern.ch/document/732486

[79]   L. Roy, "Detector Safety System for the Trigger Tracker (doc file, status table, Patch Panel connection)", EDMS CERN document, EDMS 783228, https://edms.cern.ch/document/830676

[80]   L. Roy, "Detector Safety System for the Inner Tracker (doc file, status table, Patch Panel connection)", EDMS CERN document, EDMS 783228, https://edms.cern.ch/document/830675

[81]   L. Roy, "LHCb Infrastructure DSS Alarm-Action Matrix", EDMS CERN document, EDMS 783228, https://edms.cern.ch/document/783228

[82]   E. Bogatin, "Signal Integrity-Simplified", Prentice Hall, September 2003.

[83]   H. Johnson, M. Graham, "High Speed Digital Design: A Handbook of Black Magic", Prentice Hall, April 1993.

[84]   H. W. Ott, "Noise Reduction Techniques in Electronic Systems", Wiley-Interscience, 2nd edition, March 1988.

[85]   P. Horowitz and W. Hill, "The art of electronics", Cambridge University Press, 2nd edition, July 1989.