Unterschrift der Betreuerin

Diplomarbeit

System Integration of the Global Muon Trigger for the CMS Experiment at CERN

Ausgeführt am
Atominstitut der Österreichischen Universitäten

unter der Leitung von

Univ.-Doz. Dr. Dipl.-Ing. Claudia-Elisabeth Wulz
Institut für Hochenergiephysik der Österreichischen Akademie der Wissenschaften

eingereicht an der Technischen Universität Wien
Fakultät für Physik

von

Tobias Nöbauer
Matrikelnummer 9926179, Studienkennzahl E810
Beheimgasse 40/6, 1170 Wien

Wien, 08.10.2007

Unterschrift des Studenten

Abstract

The Global Muon Trigger is a processing board which forms part of the Level 1 trigger system of the CMS experiment at CERN. The purpose of the L1 trigger system is to reduce the incoming LHC event rate of 40 MHz to 100 kHz by reconstructing physics objects (such as muons) and their types, energies/momenta and directions from coarse detector data and selecting 'interesting' events that satisfy well-defined trigger conditions. It has to do so within a small latency time and at maximum efficiency. The L1 trigger is implemented as a massively parallel custom hardware system using programmable logic technologies.

Based on FPGA technology, the Global Muon Trigger receives up to 16 muon candidates from the Regional Muon Triggers, evaluates algorithms for duplicates matching, ghost suppression and sorting and delivers the best four muons found in the three complementary muon detectors of CMS to the Global Trigger.

In the course of this thesis, the Global Muon Trigger was tested and integrated with the neighboring hardware boards that it exchanges data with. Its control software library was expanded and integrated into the trigger-wide distributed control software Trigger Supervisor.

This document gives an introduction to the Large Hadron Collider machine (chapter 1), the physics characteristics of which dictate the need for efficient and powerful triggering. The CMS detector (ch. 2) and its trigger and data acquisition systems (ch. 3) are described in detail, and the Global Muon Trigger board is introduced (ch. 4). Finally, the integration tasks carried out in the hardware (ch. 5) and software realm (ch. 6) are outlined.

Kurzfassung

Der Globale Müonentrigger ist eine Prozessorkarte im Level 1 Triggersystem des CMS Experiments am Europäischen Kernforschungszentrum CERN. Die Aufgabe des Level 1 Triggers ist es, die LHC-Kollisions- und damit Ereignisrate von 40 MHz auf höchstens 100 kHz zu reduzieren um die nachfolgenden Datenspeicher nicht vollends zu überlasten. Dazu werden aus grobkörnigen Meßdaten physikalische Objekte (z.B. Müonen) mit Teilchentyp, Energie/Impuls und Propagationsrichtung rekonstruiert. Wenn diese Objekte festgelegten Triggerbedingungen entsprechen wird das Ereignis voll ausgelesen, ansonsten werden die Daten verworfen. Der Level 1 Trigger ist als speziell entworfenes, massiv paralleles Elektroniksystem ausgeführt um den strengen Anforderungen von niedriger Latenzzeit und hoher Effektivität genügen zu können.

Der Globale Müonentrigger basiert auf programmierbaren Logik-Chips (FPGA-Technologie) und empfängt bis zu 16 Müonkandidaten von den Regionalen Müonentriggern, sucht nach doppelten oder falschen Kandidaten und sortiert sie. Die besten vier Müonen in den drei komplementären Müonendetektoren von CMS werden an den Globalen Trigger weitergeleitet.

Im Zuge dieser Diplomarbeit wurde der Globale Müonentrigger gemeinsam mit seinen Nachbarsystemen getestet, die Programmbibliotheken zu seiner Steuerung wurden erweitert und in die Trigger Supervisor Software integriert, die als verteiltes System den gesamten Level 1 Trigger steuert.

Dieses Dokument beschreibt zunächst den Large Hadron Collider (Kapitel 1), dessen physikalische Charakteristika eine hocheffiziente Ereignisauswahl notwendig machen. Der CMS-Detektor (Kap. 2) und seine Subsysteme für Trigger und Datennahme (Kap. 3) werden detailliert dargestellt, ebenso wie der Globale Müonentrigger (Kap. 4). Schließlich wird dessen Integration in das umgebende Hard- (Kap. 5) und Softwaresystem (Kap. 6) beschrieben.

Contents

Abstract **3**

Kurzfassung **5**

1 The Large Hadron Collider **9**
1.1 Introduction . . . 9
1.2 Design considerations and basic description . . . 9
1.3 Physics characteristics and goals . . . 17

2 The CMS Experiment **23**
2.1 Principal Design and Magnet System . . . 23
2.2 Tracking . . . 27
2.3 Calorimetry . . . 29
2.3.1 Electromagnetic Calorimeter . . . 29
2.3.2 Hadron Calorimeter . . . 31
2.4 Muon System . . . 33
2.4.1 Drift Tubes . . . 35
2.4.2 Cathode Strip Chambers . . . 35
2.4.3 Resistive Plate Chambers . . . 37

3 CMS Trigger and Data Acquisition **41**
3.1 Overview of the CMS Trigger and DAQ System . . . 41
3.1.1 First Level Trigger . . . 43
3.1.2 Data Acquisition . . . 45
3.1.3 High Level Trigger and LHC Computing Grid . . . 49
3.2 L1 Calorimeter Trigger . . . 49
3.3 L1 Muon Trigger . . . 53
3.3.1 Drift Tube Local Trigger . . . 53
3.3.2 DT Track Finder . . . 55
3.3.3 CSC Local Trigger . . . 57
3.3.4 CSC Track Finder . . . 59
3.3.5 RPC Trigger . . . 61
3.4 L1 Global Trigger . . . 61

4 Global Muon Trigger **67**
4.1 Conceptual Design and Functions . . . 67
4.1.1 Matching and Pairing . . . 67
4.1.2 Ghost-busting in the Overlap Region . . . 69
4.1.3 MIP/Iso Assignment . . . 71
4.1.4 Conversion of Parameters . . . 73
4.1.5 Sort Rank Assignment . . . 73
4.1.6 Merging . . . 75
4.1.7 Sorting . . . 75
4.2 Implementation . . . 77
4.2.1 Hardware Overview . . . 77
4.2.2 Firmware Development . . . 79

5 GMT Hardware Tests and Integration **81**
5.1 Software Libraries and PC–Trigger Board Communication . . . 81
5.1.1 Overview . . . 81
5.1.2 GMT Online Software . . . 81
5.2 Software/Firmware Improvements and Extensions . . . 83
5.2.1 MIP/Quiet Assignment FPGA Test Firmware . . . 83
5.2.2 MIP/Quiet Bit Tests . . . 83
5.2.3 GMT Interface Bit Error Test Class . . . 87
5.2.4 Block Transfer Test Class . . . 87
5.2.5 Standalone Emulator Tool . . . 87
5.3 Tests Performed . . . 87
5.3.1 GMT Internal Tests . . . 87
5.3.2 Interconnection Tests with PSBs . . . 87
5.3.3 Interconnection Tests with the Regional Muon Triggers . . . 89

6 GMT Software Integration **91**
6.1 Software Frameworks . . . 91
6.1.1 XDAQ . . . 91
6.1.2 Trigger Supervisor . . . 95
6.1.3 Contributions to the TS Framework . . . 99
6.2 GMT TS Cell . . . 101
6.2.1 Main Classes . . . 101
6.2.2 Commands . . . 103
6.2.3 Operations . . . 105
6.2.4 Auxiliary Classes . . . 107
6.3 Configuration Database and File Repository . . . 109
6.4 Integration Tests . . . 109

Bibliography **115**

List of Figures **120**

List of Tables **121**

Acknowledgements **123**

Chapter 1

The Large Hadron Collider

1.1 Introduction

The Large Hadron Collider (LHC) is a superconducting storage ring for two counter-rotating beams of protons (or heavy ions) currently being constructed at CERN, with first collisions scheduled for the end of 2008. The LHC will be the largest and most powerful particle accelerator ever built and is expected to provide unique insights into the nature of elementary particles and their interactions. This chapter will briefly outline the considerations relevant for the basic design of the LHC and describe its layout (section 1.2). A summary of selected physics properties and goals will be given (section 1.3).

1.2 Design considerations and basic description

A discovery machine

As Carlo Rubbia, Nobel laureate and former CERN director-general pointed out in his opening speech at the Aachen LHC Workshop in 1990, "most of our experimental knowledge has come from two complementary methods, namely (1) high-precision studies of particles and forces and (2) hunting for new particles and interactions." The first method, the "high-precision frontier" requires the production of great quantities of certain particles with a low background ("factories", eg. for the B, top or Z particles) whereas the second one, the "high energy frontier" strives for the highest collision energies attainable in combination with - as explained by Rubbia - correspondingly higher luminosities. This is due to the fact that the size of cross sections typically goes like $1/E^2$. (cf. [51])

The Livingston plot for colliders (Fig. 1.1) shows clearly that 'type 2' discovery machines, i.e. those that have first reached a given collision energy, have always been hadron colliders. This is mainly due to the fact that energy loss by synchrotron radiation P for a given particle energy E strongly decreases with particle rest mass m_0:

$$P = \frac{2q^4\gamma^2B^2}{3c^5m_0^2}v_\perp^2,$$

with q being the electric charge of the particle, B the magnetic flux density, γ the relativistic gamma factor for the particle, c the speed of light in vacuum and $v_\perp$ the particle's velocity component perpendicular to the direction of the magnetic field.

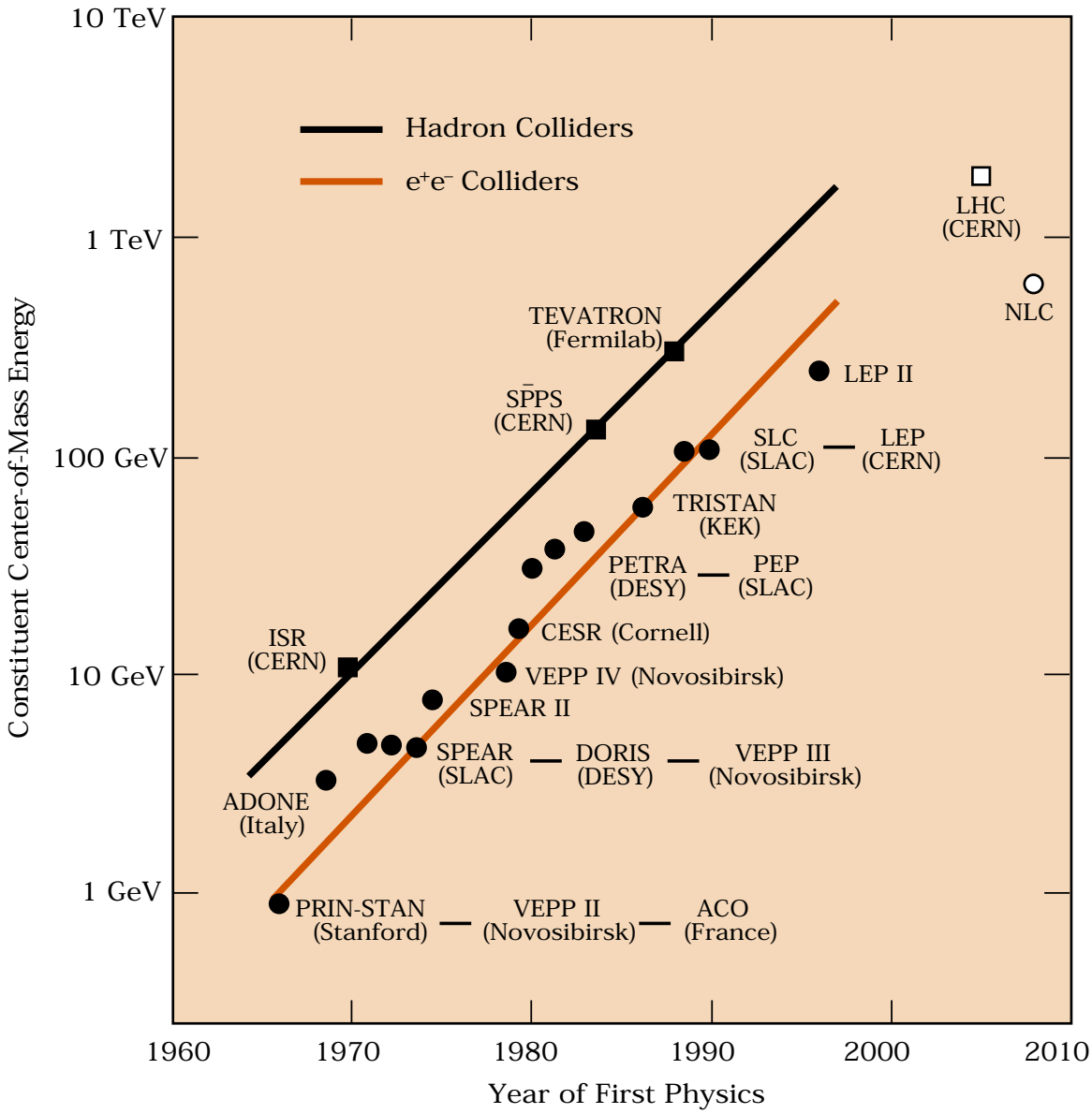


Figure 1.1: Livingston plot showing colliders only (Source: [49])

For a given particle energy $E = \gamma m_0 c^2$ and assuming $v_\perp \approx c$, this yields

$$P = \frac{2q^4 B^2 E^2}{3c^7} \left(\frac{1}{m_0^4} \right)$$

So, for electrons and protons compared:

$$\frac{P_e}{P_p} = \left(\frac{m_p}{m_e} \right)^4 \approx 2000^4 = 1.6 \times 10^{13}$$

For a proton machine like the LHC, the main limiting factor for the achievable energy is the magnetic flux density B required to bend particle tracks of given momentum p onto a given circle. Since B decreases linearly with the radius ρ of the circle, designing the accelerator ring with a radius as large as possible is a way of reducing required magnet strength (and thus radiation losses). Given the pragmatic decision to construct the LHC in the existing 27 km circumference LEP tunnel at CERN, the maximum particle momentum can be roughly estimated as follows:

$$p\,[\mathrm{GeV/c}] \approx 0.3\ B\,[\mathrm{Tesla}]\ \ \rho\,[\mathrm{Meters}]$$

For $\rho = 4200$ m and a maximum feasible flux density using superconducting magnets of $B \approx 10$ T, this allows for $p \approx 12$ TeV/c. However, an accelerator ring cannot entirely consist of bending dipole magnets: straight sections are required for acceleration, beam cleaning and tuning, insertion and extraction, monitoring instrumentation and the experiments at the interaction points. Moreover, higher order multipole magnets are interleaved with dipoles to facilitate beam focusing. In the case of the LHC, the main arc dipoles have been designed to provide a nominal flux density of $B = 8.33$ T, enough to achieve a per-proton nominal energy of 7 TeV. The LHC will accelerate two separate proton or heavy ion beams rotating in opposite directions within the same cryostat assembly, providing a center-of-mass collision energy of

$$\sqrt{s} = 14 \text{ TeV}$$

for protons.

Luminosity

As mentioned already, luminosity L is another key parameter of a particle collider, since it determines at what rate a given process will take place, and therefore the time needed to accumulate a sufficiently high level of confidence for measurements. The number of events of a specific process generated per second is given by

$$N_{event} = L\sigma_{event}$$

where σ_{event} is the cross section of the process under study. Machine luminosity L is given by

$$L = \frac{N_b^2 n_b f_{rev} \gamma}{4\pi \epsilon_n \beta^*} F$$

where N_b is the number of particles per bunch, n_b the number of bunches per beam, f_{rev} the revolution frequency, γ the relativistic gamma factor, ϵ_n the normalized transverse beam emittance, β^* the beta function at the collision point and F the geometric luminosity reduction

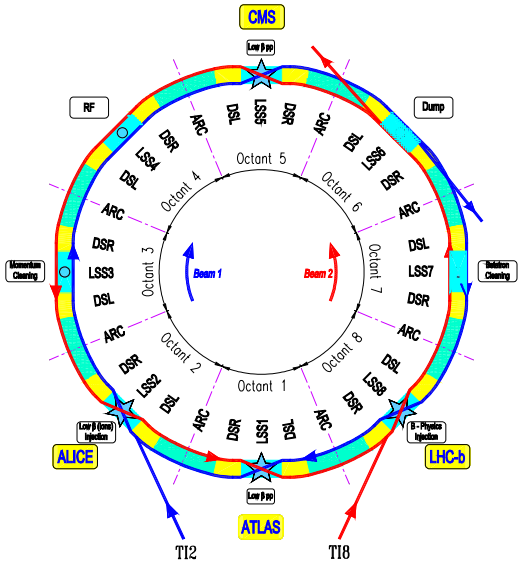


Figure 1.2: Schematic layout of the LHC (Source: [15])

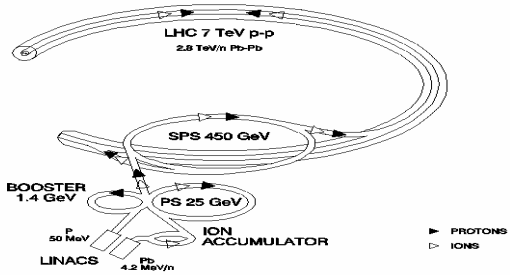


Figure 1.3: LHC injector chain (Source: [14])

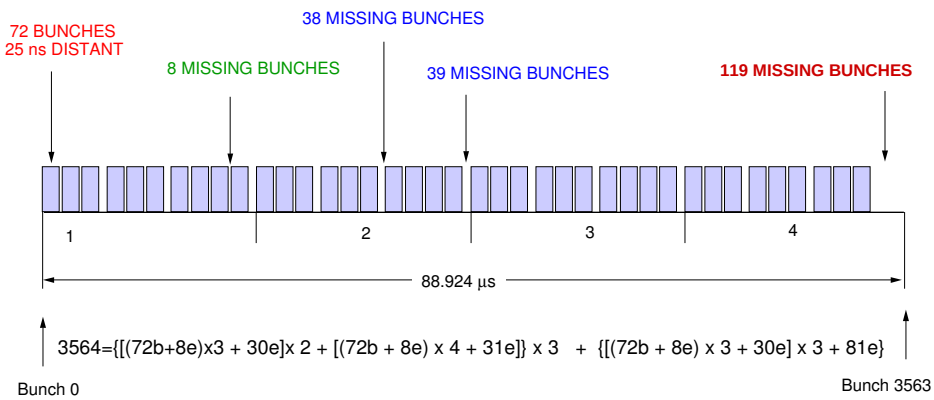


Figure 1.4: Beam structure of the LHC (Source: [59])

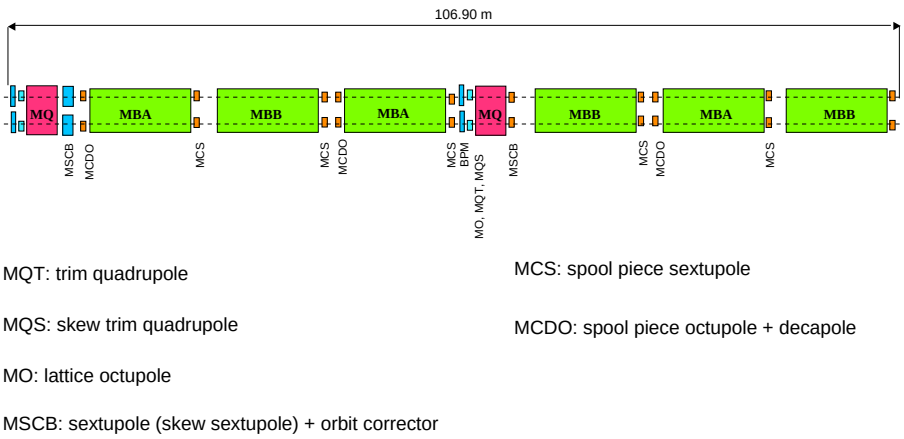


Figure 1.5: Schematic layout of the LHC arc cells. MBA and MBB denote the type A and B main dipole assemblies (Source: [15])

factor due to the crossing angle at the interaction point. Assuming equal beam parameters for both circulating beams, F is given by

$$F = 1/\sqrt{1 + \left(\frac{\theta_c \sigma_z}{2\sigma^*}\right)^2}$$

where θ_c is the full crossing angle at the interaction point ('IP'), σ_z the RMS bunch length and σ^* the transverse RMS beam size at the IP. (cf. [15]). Table 1.2 gives a selection of LHC parameters, including those relevant for luminosity.

Beam data		
Proton energy	[GeV]	**7000**
Relativistic gamma γ		7461
Number of particles per bunch N_b		1.15×10^{11}
Number of (filled) bunches n_b		2808
Circulating beam current	[A]	0.582
Stored energy per beam	[MJ]	362
Longitudinal emittance (4σ)	[eV s]	2.5
Transverse normalized emittance ϵ_n	[μm rad]	3.75
Geometry		
Ring circumference	[m]	26658.883
Ring separation in arcs	[mm]	194
Main magnets		
Number of main bending dipoles		1232
Length of main bends	[m]	14.3
Field of main bends	[T]	8.33
Synchrotron radiation		
Instantaneous power loss per proton	[W]	1.84×10^{-11}
Synchrotron radiation power per ring	[W]	3.6×10^3
Energy loss per turn	[eV]	6.71×10^3
RF system		
Revolution frequency	[kHz]	11.245
RF frequency	[MHz]	400.8
Total RF voltage	[MV]	16
Peak luminosity related data		
RMS bunch length (for Gaussian distribution) σ_z	[cm]	7.55
RMS beam size at IP 1+5 (Atlas, CMS) σ^*	[μm]	16.7
β^* at IP 1+5 (CMS, Atlas)	[m]	0.55
Half crossing angle at IP 1+5 $\theta_c/2$	[μrad]	$\pm$ 142.5
Geometric luminosity reduction factor F		0.836
Peak luminosity at IP 1+5 (CMS, Atlas)	[cm^{-2} s^{-1}]	$\mathbf{1.0 \times 10^{34}}$

Table 1.1: Overview of LHC nominal parameters (Source: LHC Design Report [15])

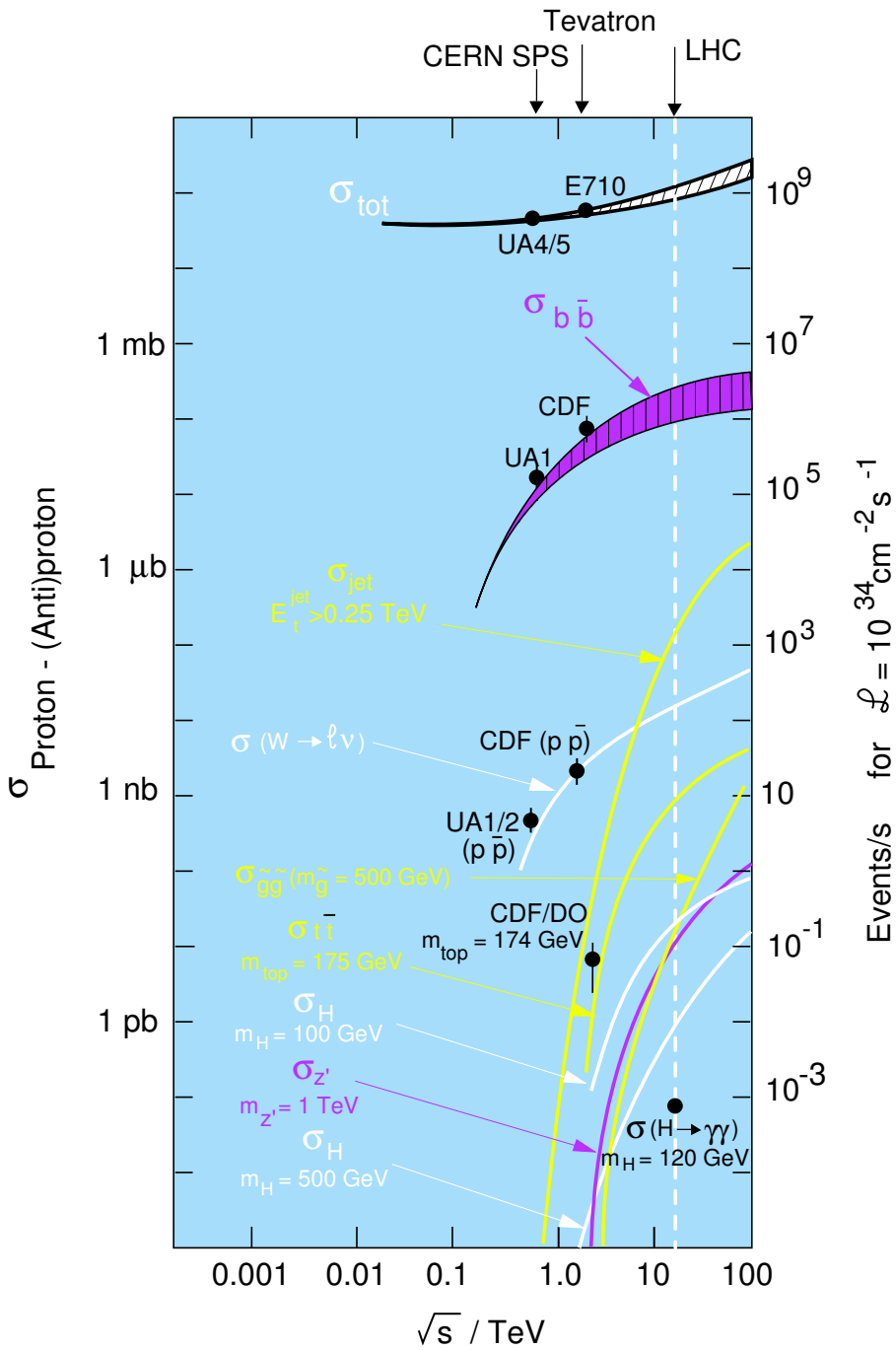


Figure 1.6: Cross sections and event rates as a function of center-of-mass energy (Source: [34])

Injector chain and beam structure

Protons for the LHC start their journey from a simple hydrogen bottle attached to a 100 keV duoplasmatron ion source. They then are injected into a 4-vane RFQ (radio frequency quadrupole) 750 keV linear accelerator (linac), followed by a 50 MeV linac ('Linac2'), the 1.4 GeV proton synchrotron booster (PSB), the 25 GeV proton synchrotron (PS) and the 450 GeV super proton synchrotron (SPS) before finally being transferred to the LHC and injected into its two beam pipes in opposite rotational directions. Heavy ions will be pre-accelerated using the same chain of machines with the exception of a separate ion source and linac. A filling, ramp-up, ejection and ramp-down cycle of the PS with a beam designed for the LHC (25 ns bunch gaps, required bunch size and distributions) will take 3.6 s. Two, three or four such PS batches (ie one filling of the PS) are injected into the SPS and ramped up to the SPS ejection energy within an SPS supercycle period of 21.6 s. 24 SPS supercycles are required to fill both LHC rings, resulting in a (theoretical) LHC filling time of about 9 minutes (real total filling time will be approximately 16 minutes due to the time required for machine fine-tuning and ring switching). Rise times of the various injection/ejection kicker magnets and capacity ratios of the interconnected accelerators result in a beam structure within the LHC as outlined in Figure 1.4; this information can be used for synchronizing various bunch crossing counters (e.g. within the LHC experiments) with the LHC orbit.

Lattice layout

The LHC follows the geometry of the LEP tunnel and has eight arcs and long straight sections (LSS). Figure 1.2 gives a schematic overview of the lattice layout with exaggerated length proportions: each LSS is about 528 m long and can house experiment or utility insertions. Each arc (∼2459 m) is made of 23 regular arc cells which are 106.9 m long and consist of two arc half-cells (53.45 m). A half-cell is made of one short straight section (SSS) cold mass (for quadrupole, octupole and higher order corrector magnets) and three bent dipole cold masses (14.3 m) with sextupole and decapole spool piece corrector magnets attached to their ends in two different configurations (type A and B dipoles). At each transition between an arc and an LSS, a dispersion suppressor section (∼200 m) adapts the LHC reference orbit to the geometry of the LEP tunnel and cancels the horizontal dispersion arising in the arc.

Straight sections, Experiments

The 8 long straight sections contain the four experiment sites with the two high-luminosity detectors Atlas and CMS located at diametrically opposite positions (LSS1+5) and insertions for accelerating RF cavities (LSS4), beam dump (LSS6) and beam cleaning (LSS3+7). Injection lines meet the LHC orbit from below the LHC reference plane within the straight sections for the two special experiments Alice (heavy-ion collisions, quark-gluon plasma, LSS2) and LHCb (B-physics, CP violation, LSS8). At the four experiment locations (interaction points), the two beams swap beam pipes and of course can be focused to collide within an area as small as possible. Both Atlas and CMS share the interaction points with smaller experiments in their forward regions: LHCf [32] consists of two small sampling calorimeters and silicon trackers ±140 m from interaction point 1 (Atlas) designed to measure the energy spectrum of very forward photons and neutral pions in order to provide data for the understanding of high energy cosmic ray air shower development. The TOTEM, CASTOR and ZDC detectors

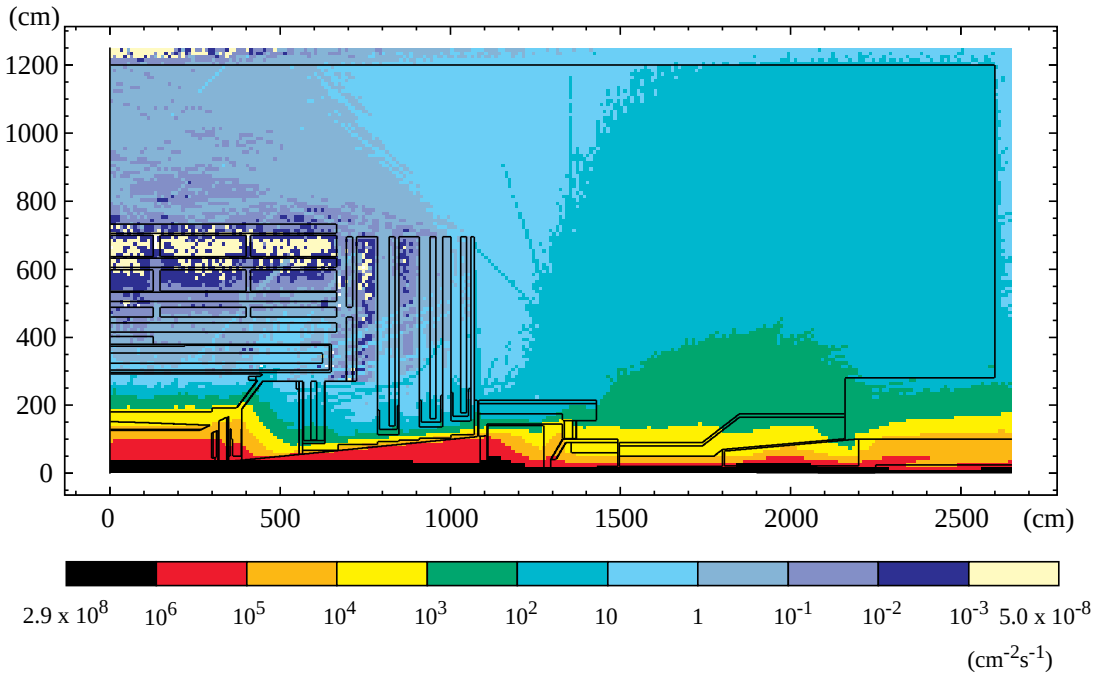


Figure 1.7: Charged particles (hadron and muon) fluxes in the CMS detector and experimental area. Data from Fluka simulation for $\mathcal{L} = 10^{34}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$(Source: [23])

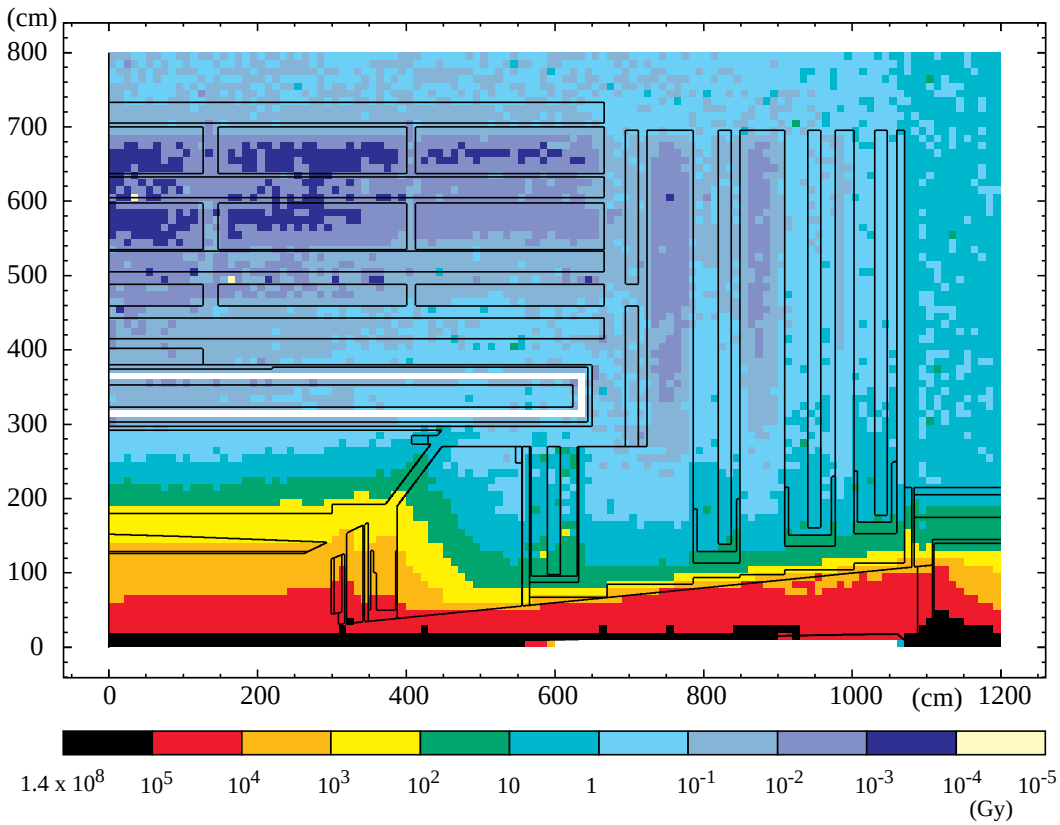


Figure 1.8: Absorbed dose (in Gy) in the CMS detector. The values correspond to an integrated luminosity of $5 \times 10^5\,\mathrm{pb}^{-1}$, as expected to be accumulated during the first ten years of LHC operation. (Source: [23])

Particle type	Multiplicity/event.
$p, \bar{p}$	9.0
$n, \bar{n}$	8.4
$\pi^{\pm}$	71.7
$K^{\pm}$	7.1
γ	1.3
π^0	40.7
other neutral	10.6
other charged	3.8

Table 1.2: Average particle multiplicities (as used for radiation environment simulation in [20])

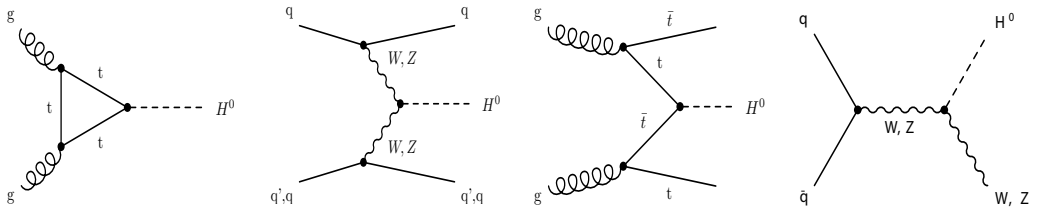
accompany CMS in the forward regions of IP 5 and will provide exceptionally high coverage for very forward particles, both neutral and charged (see next section).

1.3 Physics characteristics and goals

Including several extrapolation methods, the total cross section for proton-proton collisions σ_{tot}^{pp} at LHC energies is estimated to be 110 ± 20 mb (cf. [34]). At peak LHC luminosity $\mathcal{L} = 10^{34}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$, this gives a p-p collision rate of $R = \sigma_{tot}^{pp} \cdot L = 1.1 \pm 0.2$ GHz. The elastic cross section σ_{el}^{pp} is expected to amount about 30 mb or 27% of σ_{tot}^{pp}, diffractive (ie. soft inelastic scattering) processes will account for another 20 mb. Diffractive scattering denotes processes where only little momentum is exchanged between two colliding protons, but for instance enough for one proton to dissociate into a system of the same intrinsic quantum numbers (single-diffractive collision, cf. [12]). The TOTEM experiment at the CMS interaction point 5 will detect particles with very low transverse momenta (which originate from elastic or soft inelastic events) that otherwise would escape down the beam pipe in order to measure σ_{tot}^{pp} (and other parameters that have some importance to cosmic ray physics; cf. [41]).

The remaining inelastic non-diffractive cross section (hard collisions) which is mainly interesting for physics discoveries is expected to be about 65-70 mb. In the LHC high luminosity scenario ($\mathcal{L} = 10^{34}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$), this results in about 8×10^8 inelastic p-p events per second, creating an extremely harsh radiation environment around the interaction points. Neutral and charged pions as well as (anti-) protons and neutrons are the main components of background radiation, with especially neutrons causing practical problems due to their activation of otherwise stable nuclei. Electronic circuits that are exposed to high radiation doses have to be designed 'radiation hard' and especially the vertex detectors close to the interaction points will be severely damaged after a few years of operation and will have to be replaced. Table 1.3 gives the order of magnitudes of particle multiplicities in a minimum bias p-p collision, figure 1.8 shows expected background radiation doses for the CMS detector and the surrounding cavern.

Due to the composite nature of the proton, its total energy is distributed among the 'partons' (quarks and gluons) that make up the proton. The parton distribution function ('PDF') describes the probability of finding a parton with a certain longitudinal momentum



(a) gluon gluon fusion (b) vector boson fusion (c) $t\bar{t}$ associated production (d) W, Z associated production

Figure 1.9: Feynman diagrams for main Higgs production channels at the LHC (Source: [54])

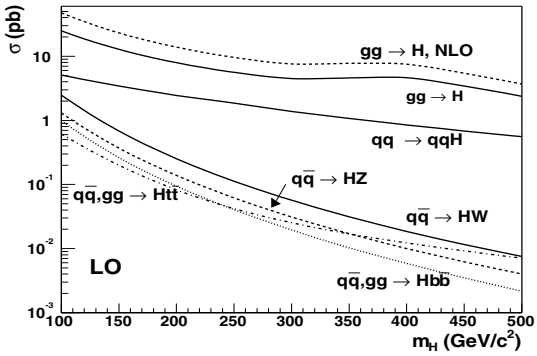


Figure 1.10: Calculated (leading order) cross sections for the SM Higgs boson as a function of m_H (Source: [9])

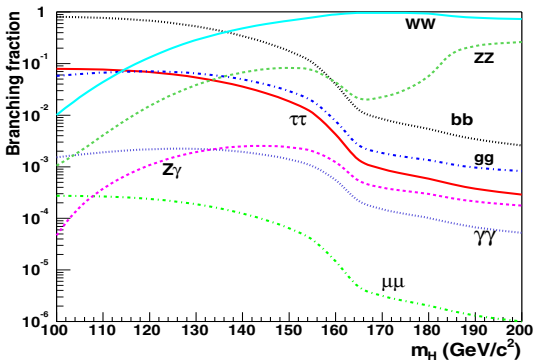


Figure 1.11: Calculated branching ratios for the SM Higgs boson as a function of m_H (Source: [9])

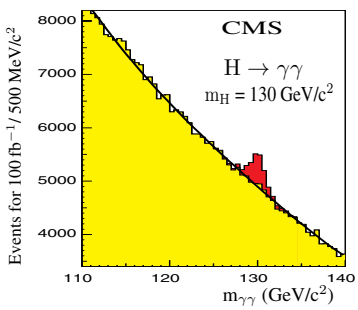


Figure 1.12: Reconstructed di-photon invariant mass distribution of the H $\rightarrow \gamma\gamma$ signal (dark) and the background (light) with $m_H = 130 GeV/c^2$ for $100 fb^{-1}$ (Source: [9])

Process	Events/s	Events/year
$W \to e\nu$	40	4×10^8
$Z \to ee$	4	4×10^7
$t\bar{t}$	1.6	1.6×10^7
$b\bar{b}$	10^6	10^{13}
$\tilde{g}\tilde{g}$ (m = 1 TeV)	0.002	2×10^4
Higgs (m = 120 GeV)	0.08	8×10^5
Higgs (m = 120 GeV)	0.08	8×10^5
Higgs (m = 800 GeV)	0.001	10^4
QCD jets $p_T > 200$ GeV	10^2	10^9

Table 1.3: Approximate event rates of some physics processes at the LHC for a luminosity of $\mathcal{L} = 2 \times 10^{33}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$. One year is equivalent to an integrated luminosity of 20 fb^{-1}. (Source: [30])

fraction x and momentum transfer Q^2; the PDF currently cannot be calculated analytically (due to the inherent non-perturbative effect in QCD bound states) nor numerically (due to limitations in lattice QCD), the various known PDFs have been derived from experimental data and are used to generate simulated collision events for physics and detector simulation (cf. [30]). Unlike for leptons, a hadron beam has to be regarded as a 'broadband' beam of quarks and gluons with many internal degrees of freedom, which poses additional difficulties for data analysis.

As can be seen from Figure 1.6, the LHC can be considered a 'factory' for $b - \bar{b}$ and $t - \bar{t}$ pairs as well as W and Z bosons. Jets (visible as mostly hadronic showers in calorimeters) of great energy will be observed at unprecedented rates, imposing great demands on triggering and event reconstruction. Table 1.3 gives an overview of event rates for selected physics processes.

The average number of inelastic non-diffractive interactions per bunch crossing will be 25 at design luminosity, leading to 'pile-up', i.e. the superposition of secondary particles originating from distinct collisions within the detector. In order to maintain signal quality, the detector has to have a fast response time (so that not too many subsequent bunch crossing get superimposed) and fine granularity (so that the probability for two particles to hit the same detector channel is small).

Search for the Higgs boson

The most prominent and important objective for the LHC and its general purpose detectors is the search for the Higgs boson, the last so-far unseen particle predicted by the Standard Model (SM) and origin of the electroweak symmetry breaking mechanism. In proton-proton collisions at LHC energies, the Higgs particle can be produced by several processes as illustrated in figure 1.9: gluon-gluon fusion through a top quark loop (a) has the largest cross section over the entire range of possible Higgs masses m_H(100 to ~1000 GeV/c^2). The cross section for vector boson fusion (b) is roughly one order of magnitude smaller over most of the mass range and becomes important only at high m_H. Associated $\mathrm{t\bar{t}}$ (c) and W/Z (d) production processes occur significantly more rarely (cf. figure 1.10) but will be more promising processes for extracting a Higgs signal from the background due to the decays of the t and W/Z particles.

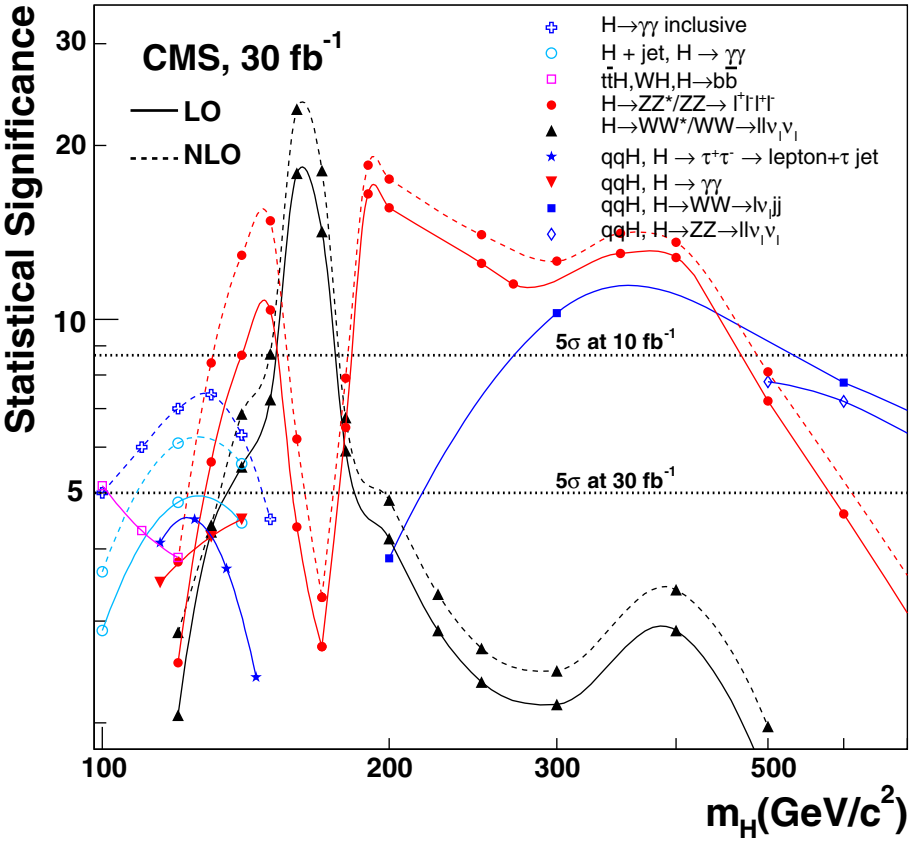


Figure 1.13: Expected statistical significance with 30 fb^{-1} for the SM Higgs boson as a function of $\mathrm{m_H}$ (Source: [9])

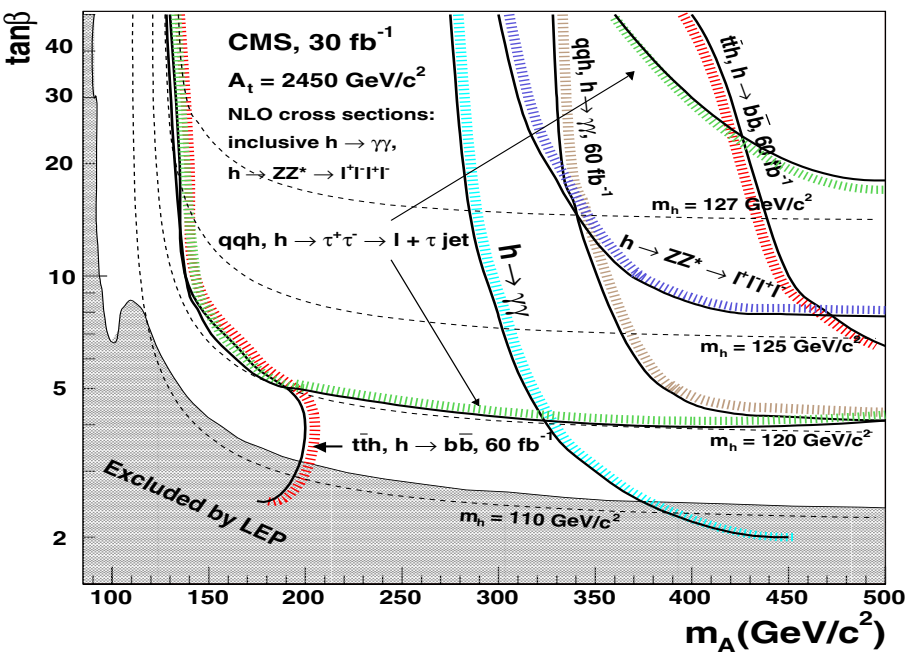


Figure 1.14: The 5σ-discovery potential for the lighter scalar MSSM Higgs boson as a function of $\mathrm{m_A}$ and $\tan\beta$ for $30\mathrm{fb}^{-1}$ with $\mathrm{A_t} = 2450\mathrm{GeV/c}^2$ (Source: [9])

Figure 1.11 gives an overview of the decay channels of the Higgs particle for $m_H < 200$ GeV/c^2.

In order to outline a search strategy for the Higgs boson, detailed simulation studies have been carried out for the LHC detectors (for CMS c.f. [9], [30]), incorporating all major production and decay channels as well as detector resolution, signal-to-background ratios and possibilities for selecting appropriate event signatures in the trigger. The main discovery channels can be outlined as follows (cf. [30], [54], [9]):

$\mathbf{m_H < 130\,GeV/c^2}$: Since the dominant decay channel $\mathrm{H} \to \mathrm{b\bar{b}}$ in this mass range has a very high background from QCD processes, $\mathrm{H} \to \gamma\gamma$ becomes the most promising channel, putting great demands on the electromagnetic calorimeter ('ECAL') in terms of energy resolution (since photons leave no trace in the tracker or other sub-detectors). Events can be selected using single- and diphoton triggers, requiring isolated photon candidates (i.e. no additional energy deposited in the tracker/ECAL within a cone around the track of the photon) improves selection efficiency. The $\mathrm{H} \to \mathrm{b\bar{b}}$ channel remains useful only if the Higgs is produced in association with a $\mathrm{t\bar{t}}$ pair or W boson.

$\mathbf{130\,GeV/c^2 < m_H < 2\,m_Z}$: The $\mathrm{H} \to \mathrm{ZZ}^* \to \ell^+\ell^-\ell^+\ell^-$ ($\ell \ldots$ electron or muon) channel is dominant in most of this region, with the exception of the area around $m_H \approx 2\,\mathrm{m_W}$, where the channel $\mathrm{H} \to \mathrm{WW} \to \ell\nu\ell\nu$ has a higher branching ratio. Since that channel involves neutrinos which are not detected directly but only due to missing energy in the whole detector, the background has to be known very well in order to be able to derive a clear mass measurement. Both channels can be selected using di- and multi- electron or muon triggers.

$\mathbf{2\,m_Z < m_H < 700\,GeV/c^2}$: $\mathrm{H} \to \mathrm{ZZ}^* \to \ell^+\ell^-\ell^+\ell^-$ gives a very clean signal in this mass range, with hardly any background. Since for $m_H > 500$ GeV/c^2 the cross sections for this decay channel are decreasing, more complicated channels involving jets and neutrinos will have to be used which require a well-understood background and more complex trigger conditions.

Top-quark Physics

Due to its high luminosity and energy, the LHC will produce many heavy particles of the Standard Model ('SM') at unprecedented rates, such as bottom and top quarks, $\mathrm{W}^\pm$ and Z^0. The LHC physics program therefore includes precision studies of many SM processes, which will help in verifying or extending the SM; most importantly, the t quark mass, its detailed branching ratios and spin properties will be under investigation. The t quark will be produced at an estimated production cross section of 830pb mainly through gluon-gluon fusion (90%) and quark-anti-quark annihilation (10%) as $t\bar{t}$ pairs. Within the SM, the t decays mostly into a W boson and a b quark. The W in turn may decay into leptons ($e^-\bar{\nu}_e, \mu^-\bar{\nu}_\mu, \tau^-\bar{\nu}_\tau$ and charge conjugates), or into quarks, $u\bar{d}', c\bar{s}'$ and c.c. A fully leptonic decay of the W^+W^- system (branching ratio 11.1%) therefore offers a very clear signature to the trigger (at least two jets and at least 2 oppositely charged leptons of high transverse momenta), S/N ratio can be improved by requiring lepton isolation and missing transverse energy (due to neutrinos in leptonic and semi-leptonic decay modes) of $E_T^{\mathrm{missing}} > 40$GeV. Event selection possibilities for the other (semi-leptonic and hadronic) decay modes in the case of CMS are discussed in detail in [30].

CP-Violation, b-quark Physics

CP violation is known to occur in the neutral kaon system (which has been studied in detail e.g. by the NA48 experiment at CERN, c.f. [46], [38]) and in the system of neutral B mesons. The large production cross section of 500 μb for b quarks at the LHC (i.e. $\mathcal{O}(10^6)$ $b\bar{b}$ pairs per second at low luminosity $\mathcal{L} = 2 \times 10^{33}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$) allows for extensive study of this system, with the LHCb experiment ([31]) specially designed for B-physics measurements; the two general purpose detectors CMS and Atlas ([19]) however will also be able to take relevant data for the determination of the parameters of the Cabibbo-Kobayashi-Maskawa (CKM) matrix. Event selection will be difficult due to large pileup (especially in the high luminosity scenario) and rather soft leptons in many final states. Decay channels involving muons (especially $B_s^0 \to J/\Psi\,\phi \to \mu^+\mu^-K^+K^-$ which allows for measurement of the Wolfenstein parameter η, c.f. [57]) will offer a lower background and better selection efficiencies.

Heavy Ion Physics

The LHC will be capable of accelerating heavy ion nuclei such as Pb to energies as high as 1150 TeV (for Pb–Pb) in the center-of-mass system (5.5 TeV per nucleon-nucleon cm-system), generating a concentration of $\mathcal{O}(10\,\mathrm{TeV})$ of energy within a volume of $\mathcal{O}(100\,\mathrm{fm}^3)$. This will allow the study of many-body dynamics of strongly interacting matter, especially a (predicted) new form of matter, the Quark-Gluon-Plasma. The Alice experiment ([18]) has been constructed in order to study heavy ion collisions, the general purpose experiments also have developed corresponding programs. Fundamental problems such as the nature of quark confinement and aspects of early universe cosmology will be addressed in these experiments.

Supersymmetry, Beyond the Standard Model

It unfortunately is beyond the scope of this thesis to go into the details of theories extending the Standard Model, of which Supersymmetry ('SUSY') is the most prominent one. It should however be noted that tests of these (possible) new theories and measurements of their parameters comprise the most exciting possibilities of the LHC, opening a door to an entirely new domain of physics.

Chapter 2

The CMS Experiment

The Compact Muon Solenoid ('CMS') experiment is one of the four large particle detectors currently under construction for the LHC accelerator at CERN. A general purpose discovery experiment, CMS was designed for "a very good and redundant muon system, the best possible electromagnetic calorimeter, a high quality central tracking and a financially affordable detector" [20]. Since the first presentation of the CMS concept at the LHC workshop in Aachen 1990 [33], the CMS collaboration has grown to 2300 members from 159 institutions in 37 countries, including many non-CERN member states from four continents. The 500 million CHF detector will be ready for first collisions by the end of 2007.

2.1 Principal Design and Magnet System

The design goal of very good momentum resolution for muons ($\Delta p/p \approx 10\%$ at $p = 1\,\mathrm{TeV/c}$), electrons and other charged particles without too stringent demands on alignment and spatial resolution of the muon systems led to the choice of a strong solenoidal field in the central region, which is returned by a saturated iron yoke. The original plan for additional forward toroids was dropped soon for reasons of feasibility and cost. A superconducting solenoidal coil of 13 m length and a free inner diameter of 5.9 m provides a nominal magnetic field of 4 T (on the beam axis) and is large enough to fully accommodate inner tracking and calorimetry. This field configuration parallel to the beam bends charged particle tracks in the transverse plane (as opposed to toroidal configurations), which eases muon identification and triggering on kinematic cuts. Furthermore, a solenoid magnet allows for a more compact detector, since full track bending starts at r = 0, while for toroidal configurations the field is very small close to the beam axis (which might make necessary an additional smaller solenoid as in the case of Atlas). Table 2.1 gives the main parameters of the coil, which is made up of 45 km of NbTi/Cu Rutherford-type superconducting cable (made of 36 strands) enclosed in a high purity aluminium stabiliser and an aluminium alloy mechanical reinforcement. The self-supporting conductor was wound in four layers using a custom winding machine. A massive cryogenic system is necessary to cool the 220 t cold mass to liquid helium temperature, along with an 800 kVA DC power supply system which delivers up to 20000 A at 16 V. Figure 2.2 shows a schematic rendering of the magnet coil inside the cryostat assembly.

The cylindrical shape of CMS is made up of five dodecagonal barrel wheels (YB-2, YB-1, YB0, YB+1, YB+2) and two times three endcap discs (YE±3, YE±2, YE±1), all made of steel. These elements act both as magnet return yoke and as mechanical support for all the

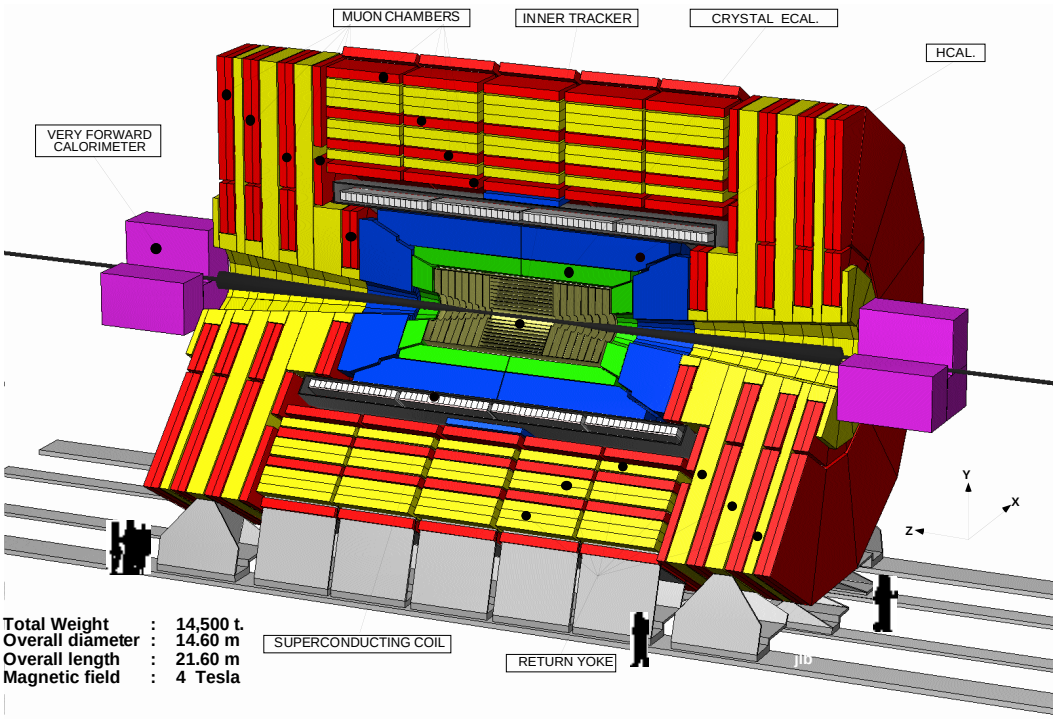


Figure 2.1: Schematic elevation view of CMS

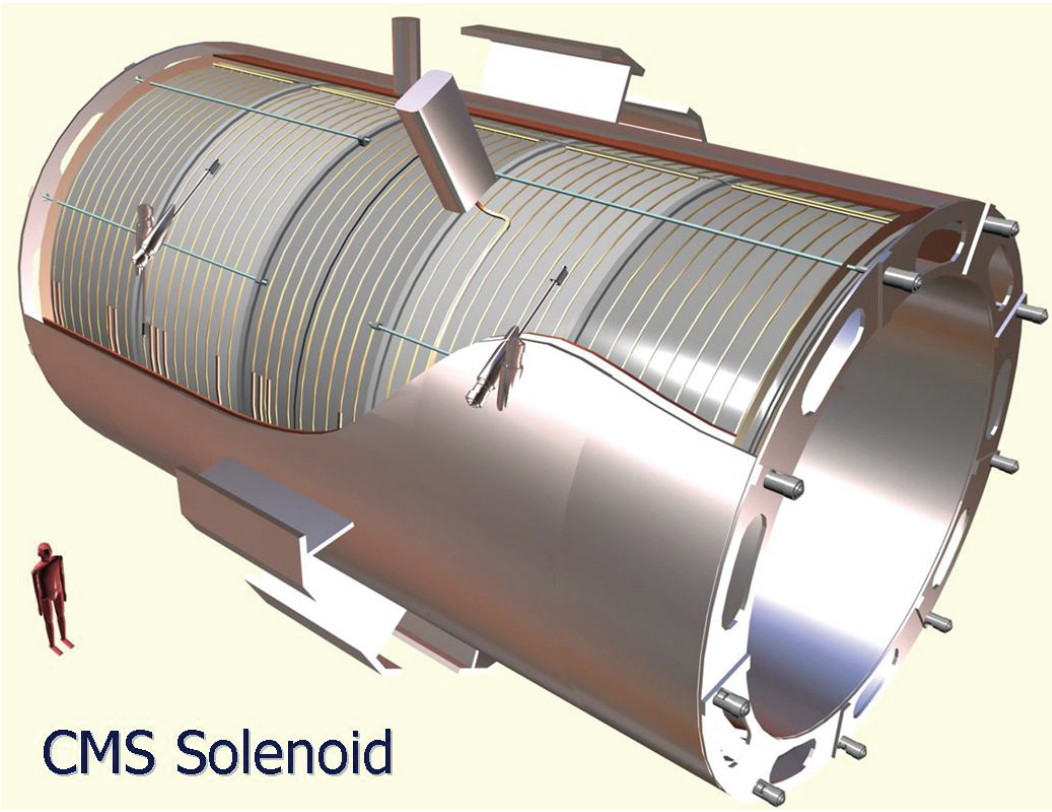


Figure 2.2: Schematic view of the CMS magnet inside the cryostat assembly

Field	4 T
Inner Bore	5.9 m
Length	12.9 m
Number of Turns	2168
Current	19.5 kA
Stored energy	2.7 GJ
Hoop stress	64 atm

Table 2.1: Parameters of the CMS superconducting solenoid. (Source: [29])

other detector elements. The central wheel YB0 fully supports the magnet inside its vacuum tank, which in turn supports the hadronic and electromagnetic calorimeters (barrel sections only; HB± and EB±) and the inner tracking assembly. The steel plates which make up the wheels are interleaved with four stations of muon chambers (drift tubes and resistive plate chambers). These plates additionally act as absorbers for particles that 'punched through' the calorimeters in order to ensure that only muons get detected in the muon chambers. Each ring is 2536 mm long and has an outer diameter of 13990 mm and an inner diameter of 7600 mm. The four gaps between the five rings (two times 120 mm, two times 200 mm) are designed to exit the cables and services from the central detectors. This results in an overall barrel yoke length of 13320 mm. The two innermost endcap discs (YE±1) support the forward parts of HCAL and ECAL (HE± and EE±), the two outermost ones carry the very forward calorimeters. All endcap discs are covered with cathode strip muons chambers (CSCs) and resistive plate chambers (RPCs). All wheels and discs have been assembled in the surface hall SX5 at the CMS site at LHC interaction point 5 and at the time of writing of this document consecutively are being lowered into the underground experimental hall UXC5 at LHC level (about 100 m below ground) using a massive gantry crane. Table 2.1 gives an overview of the masses of the 15 major parts of CMS. For detailed information on the magnet and the return yoke, cf. [24].

The coordinate system used by the CMS collaboration is depicted in Figure 2.1: x points to the center of the LHC ring, y upwards and z consequently points tangentially to the ring, roughly toward the west (Jura side); since the LHC orbit is inclined by 1.23% with respect to the horizontal plane, so are y and z. For physics discussions often a (pseudo-) spherical coordinate system with coordinates r (radial distance to beam line) ϕ (azimuthal angle, measured from the positive x-axis in the $x-y$ plane) and η (pseudorapidity) is used, where

$$\eta = -\ln\left(\tan\frac{\theta}{2}\right),$$

with the polar angle θ measured from the positive z-axis. Pseudorapidity is an approximation for rapidity Y, which is defined by

$$Y = \frac{1}{2}\ln\frac{E+p_z}{E-p_z},$$

where E and p are energy and momentum of a particle. Rapidity is especially useful because a shift of the center-of-mass in a collision in z direction away from a nominal interaction point only adds a constant and leaves the distribution dN/dY invariant. Such a shift is typical

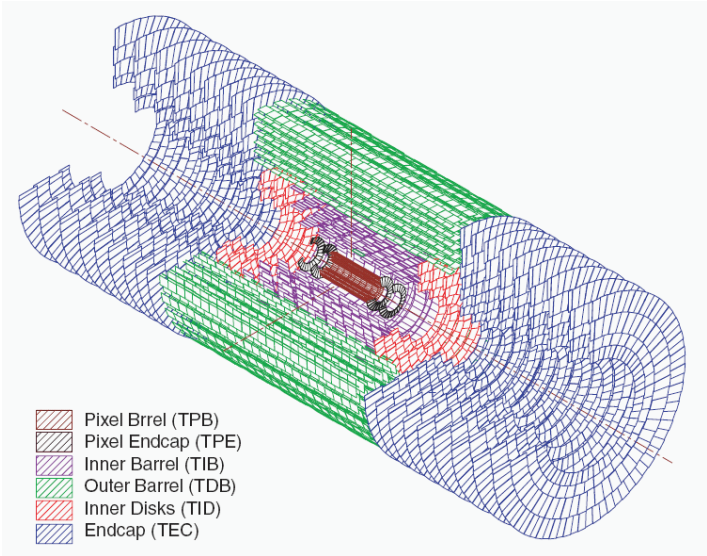


Figure 2.3: Schematic elevation view of the CMS tracking system

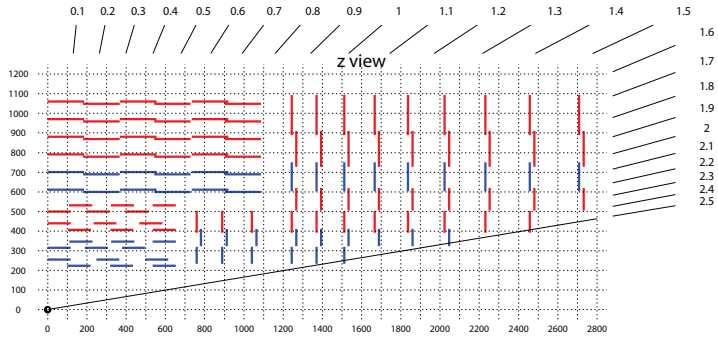


Figure 2.4: Projected view of one quarter of the CMS tracking system (without vertex pixel detector). Red lines: single-sided modules, blue lines: double-sided modules. Angle coordinate is pseudorapidity; x and y in mm

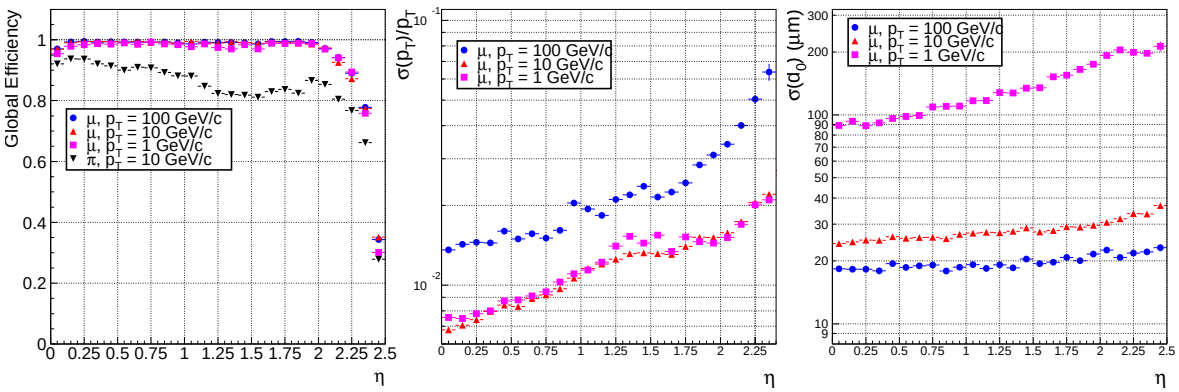


Figure 2.5: Performances of the CMS tracker reconstruction of charged particles. Left: track reconstruction efficiencies as a function of pseudo-rapidity $|\eta|$ for muons and pions of different tranverse momentum p_T. Center: track transverse momentum resolutions for muons as a function of $|\eta|$ and for different p_T values. Right: track impact parameter resolutions for muons as a function of $|\eta|$ and for different p_T values. (Source: [10])

Designation	Mass in tonnes
HF+	250
YE+3	410
YE+2	880
YE+1	1310
YB+2	1250
YB+1	1250
HB+	700
YB0	1920
HB-	700
YB-1	1250
YB-2	1250
YE-1	1310
YE-2	880
YE-3	410
HF-	250

Table 2.2: Masses of the major parts of CMS

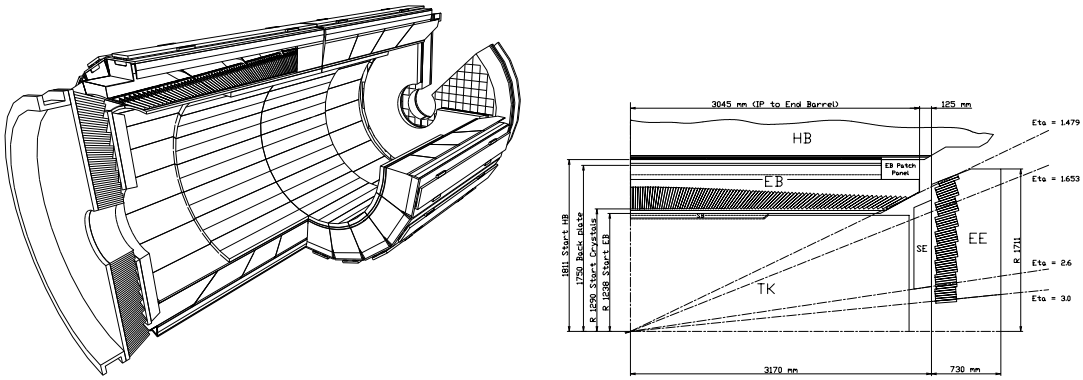
for hadron collisions, since the interacting parts are the partons, among which energy is not distributed equally. Pseudorapidity is a good approximation for rapidity for $p \gg m$ and $\theta \gg 1/\gamma$.

2.2 Tracking

The purpose of the tracking system is to determine vertices of particle decays and measure particle trajectory parameters – such as impact parameter (defined as the shortest distance between the origin and a particle track) and sagitta (in order to calculate particle momentum) – of charged particles (especially electrons, muons, charged mesons) and photons. Identifying lepton and photon isolation (i.e. no other energy deposited around a particle track) also plays a crucial role in many particle searches. Since the strong magnetic field forces soft particles onto helical tracks of rather small radius (typically < 60 cm), particle density is very high up to that radius, requiring especially high pixel/strip granularity in order to minimize occupancy (i.e. the ratio of pixels/strips with a particle hit) and electronics designed to withstand this hard radiation environment. Granularity along with the number of measured points per track and well-known alignment are crucial factors for momentum and vertex resolution. Figure 2.5 gives an overview of expected tracker performance and resolution.

After initial plans to design the tracking system with a silicon pixel vertex detector, a silicon microstrip tracker and an outer micro-strip gas chamber tracker ('MSGC') [25], the CMS collaboration decided in 1999 to alter the design, dropping the MSGC in favor of an all-silicon layout [26], replacing the MSGC with an additional outer barrel silicon strip tracker. Figures 2.3 and 2.4 show the principal layout: the tracker is divided into a barrel part with cylindrical layers and two forward (endcap) parts with disc layers; its overall length is about 270 cm.

The vertex detector (overall length: 92 cm) consists of two barrel layers and two times two



(a) 3D view

(b) side view of one quarter

Figure 2.6: Layout of the CMS ECAL

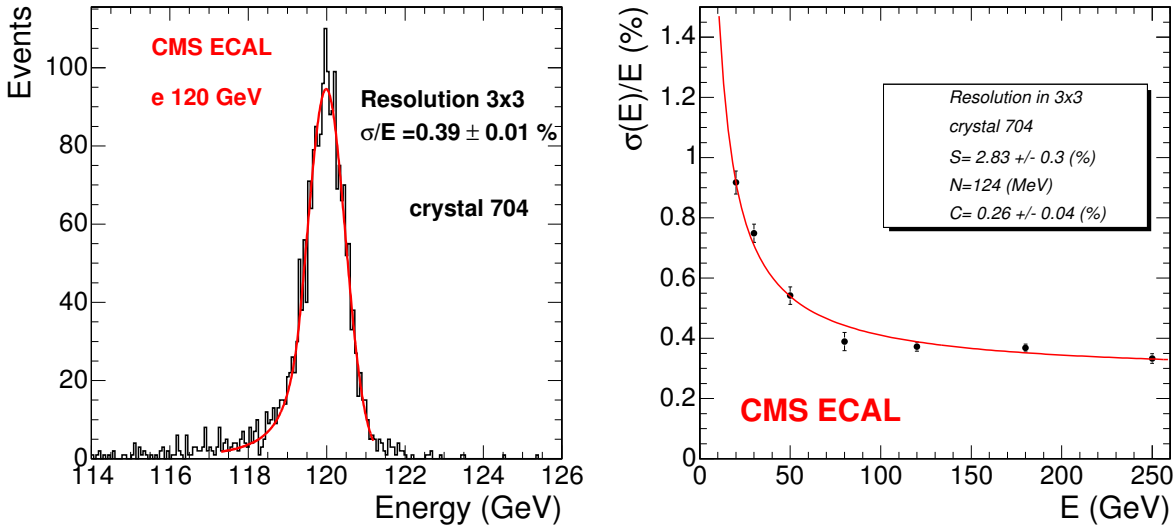


Figure 2.7: Left: Distribution of energy reconstructed in a 3×3 crystal matrix for electrons at 120 GeV. Right: Resolution as a function of the energy (Source: [63])

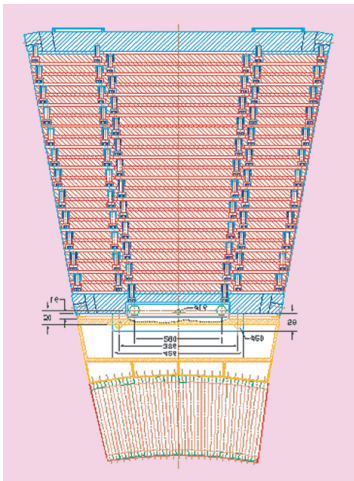


Figure 2.8: Cross section in transverse plane of one 20° wedge of the CMS HCAL

forward discs on which the silicon modules (n-on-n type) are mounted. Pixels are rectangular with $150 \times 100\,\mu$m edge length and a width of 250 μm, which makes possible a hit resolution of 10–15 μm using interpolation between the charges distributed in adjacent pixels. The endcap layers are mounted at $z = \pm 325$ mm and $z = \pm 465$ mm, for the barrel layers there are two different configurations for the high/low luminosity scenarios respectively (layers at $r = 4.3$ and 7.2 cm for low luminosity; $r = 7.2$ and 11.0 cm for high luminosity). In the high luminosity configuration, the vertex detector will have an active surface of about one square meter with 66 million pixels, delivering data on 40×10^6 readout channels.

The inner and outer trackers consist of an overall ten barrel layers and nine end cap discs on each side, extending to a radius of 110 cm and providing track hits up to $|\eta| = 2.5$. They will be composed of 15148 detector modules with one or two sensors each which are supported by a carbon-fiber frame. Module dimensions (thickness 320–500 μm, pitch size 60–270 μm) and active strip area densities vary among the different parts of the tracker, with an overall expected occupancy of 1% in the high luminosity scenario. All parts of the tracking system are operated at -10 °C and 30% relative humidity in order to avoid reverse annealing of the silicon modules and protect the sensors from increased leakage current due to radiation damage. Some 60 kW of thermal power will have to be extracted from the front-end electronics.

2.3 Calorimetry

2.3.1 Electromagnetic Calorimeter

The CMS electromagnetic calorimeter (ECAL, [21]) uses a total of 83000 $PbWO_4$ crystals as both absorbing and scintillating material (i.e. a 'homogeneous' calorimeter as opposed to a 'sampling' one) with avalanche photodiode (barrel) and vacuum photo-triodes (endcap) photodetectors. $PbWO_4$ crystals have been chosen because of their short radiation length and small Molière radius – which allows for a compact calorimeter that fits within the CMS magnet – and because it is a rather fast scintillator, which helps in minimizing pile-up from subsequent bunch crossings in the high luminosity LHC environment. (Table 1.1 sums up characteristic properties of common calorimetry crystals.) Barrel crystals have a 22×22 mm front face ($\Delta\eta \times \Delta\phi = 0.0175 \times 0.0175$), matching the Molière radius (i.e. lateral shower dimension) and are 23 cm thick, an equivalent of 26 radiations lengths, which ensures that practically all showers are fully absorbed within the detector volume. In order to maximize effective thickness, the crystals in the barrel part point toward the interaction point, with an additional tilt of 3° in η or ϕ to increase geometric coverage. The crystals in the endcaps ($1.48 < |\eta| < 3.0$) have the same size front face but become wider toward the outside to a maximum granularity of $\Delta\eta \times \Delta\phi = 0.05 \times 0.05$. There are additional preshower detectors at the insides of the endcap crystal ECALs, consisting of two layers of Pb absorber interleaved with silicon strip detectors. The preshower detector improves spatial resolution for photons to about 300 μm at 50 GeV, which helps in $\pi^0 - \gamma$ separation. Test beam measurements e.g. with 3×3 crystals have resulted in an energy resolution $\sigma(E)/E = 0.39$ for 120 GeV ([63]) electrons, which satisfies design specifications. Figure 2.7 shows the interpolated energy resolution function for 20–250 GeV electrons.

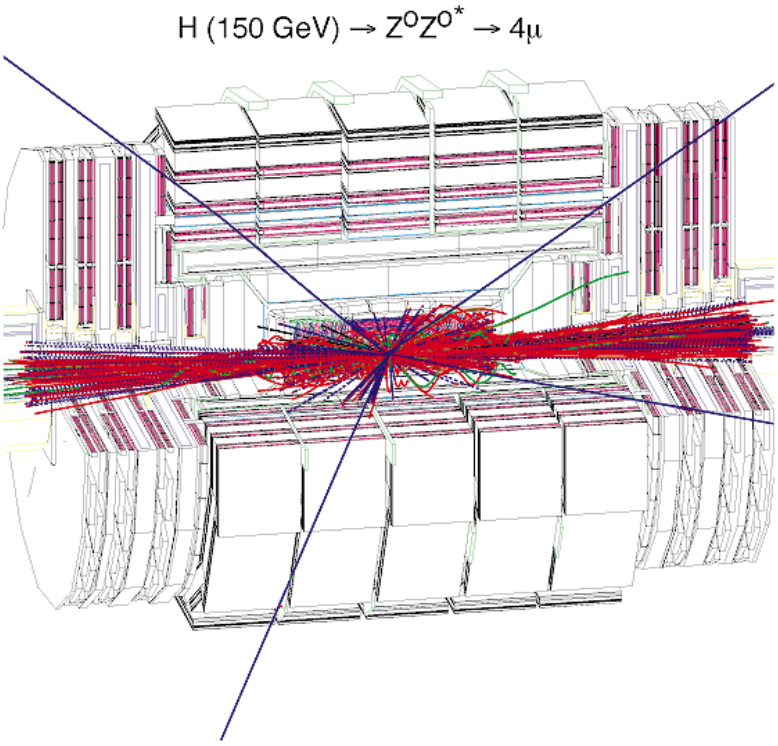


Figure 2.9: A 150 GeV Higgs event decaying into four muons in the CMS detector (Source: [23])

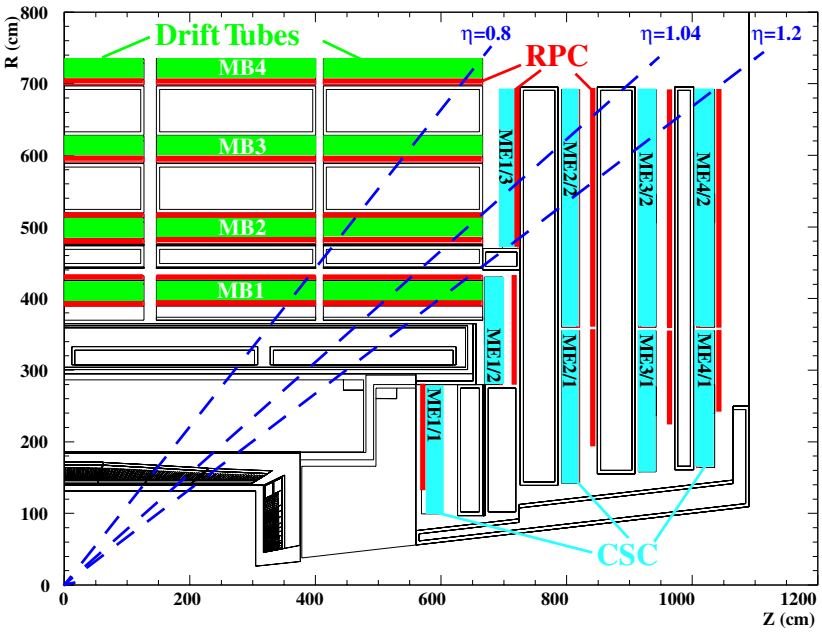


Figure 2.10: Side view of one quarter of CMS, showing positions of the muon systems (Source: [54])

	NaI(Tl)	BGO	CSI	PbWO4
Density [g/cm^3]	3.67	7.13	4.51	8.28
Radiation length [cm]	2.59	1.12	1.85	0.89
Molière radius [cm]	4.80	2.33	3.50	2.19
Light decay time [ns]	230	60 300	16	5 (39%) 15 (60%) 100 (1%)
Relative light output	100	18	20	1.3

Table 2.3: Comparison of properties of various crystals (Source: [21])

2.3.2 Hadron Calorimeter

As opposed to the CMS ECAL, a 'sampling' design was chosen for the Hadron Calorimeter ('HCAL', [22]), which consists of a central part (split into barrel and endcap modules, 'HB' and 'HE', $|\eta| < 3.0$ and a forward part ('HF', $3.0 < |\eta| < 5.3$). In the barrel part, there are 13 layers of 50 mm copper absorber plates interleaved with 4 mm thick plastic scintillator, plus two layers of stainless steel absorber at the innermost and outermost positions for structural strength. Readout is done using wavelength-shifting plastic fibers and hybrid photodiodes. HB is about 79 cm deep, which is equivalent to 5.15 interaction lengths for $|\eta| = 0$. The endcap parts of the central calorimeter consist of 19 copper absorption layers with the same active elements. Both HB and HE are segmented radially (into absorption-scintillation layers, which are grouped into sections HB1, HB2, HE1, HE2, HE3) and in η and ϕ with a granularity of $\Delta\eta \times \Delta\phi = 0.087 \times 0.087$, which is important for the separation of two jets close together in space (dijet events). Since the depth available for the HCAL within the magnet coil is not sufficient to fully absorb high-energy jets and measure late shower development, an additional Outer Calorimeter ('HO') with one (two in the central ring YB0) coarse absorber layers has been included, which is situated outside the magnet coil after the first iron return yoke which is used as an absorption layer as well. This setup results in a total thickness of 11 absorption lengths. Since accurate measurement of missing transverse energy E^T_{miss} (i.e. energy carried by weakly interacting neutral particles such as neutrinos) is a key requirement for the study of the top quark as well as Standard Model and SUSY Higgs searches, the HCAL has been designed for very good hermeticity by adding forward calorimeters outside of the outermost endcap discs, extending coverage up to $|\eta| = 5.3$. The radiation environment is extremely harsh in this very forward region close to the beam pipe: an average of 760 GeV per event are expected to hit HF, compared to a total of 100 GeV for the whole rest of CMS. Therefore an entirely different design has been chosen: each HF – located at 11.1 m from the interaction points at both sides – consists of a large block of steel absorber with embedded quartz fibers parallel to the beam. Incident particles cause showers, of which some charged particles will have momenta above the Cherenkov threshold and cause detectable Cherenkov light in the fibers. HCAL energy resolution is expected to be $\sigma_E/E = 65\%/\sqrt{E} \oplus 5\%$ for HB, $\sigma_E/E = 85\%/\sqrt{E} \oplus 5\%$ for HE and $\sigma_E/E = 100\%/\sqrt{E} \oplus 5\%$ for HF, with E given in GeV ([54]).

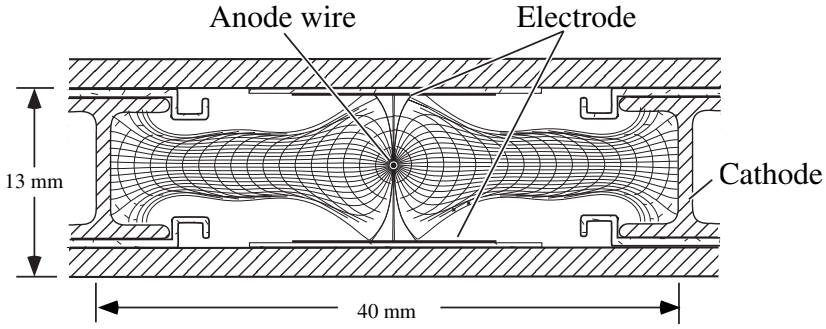


Figure 2.11: Cross section of a drift cell with electric field lines, without magnetic field (Source: [23])

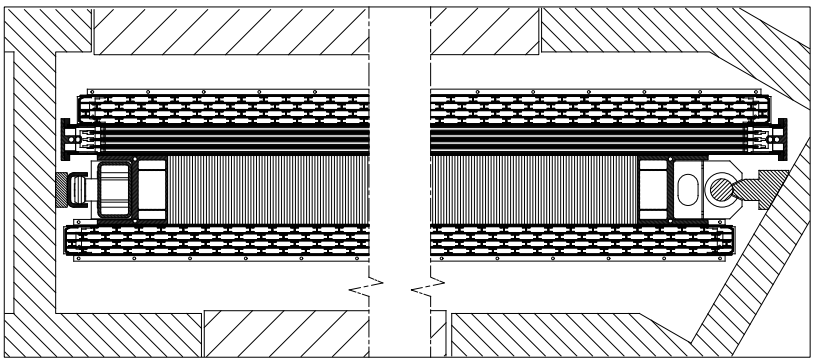


Figure 2.12: Cross section in the $r - \phi$ plane of a drift tube chamber inside the iron yoke (Source: [23])

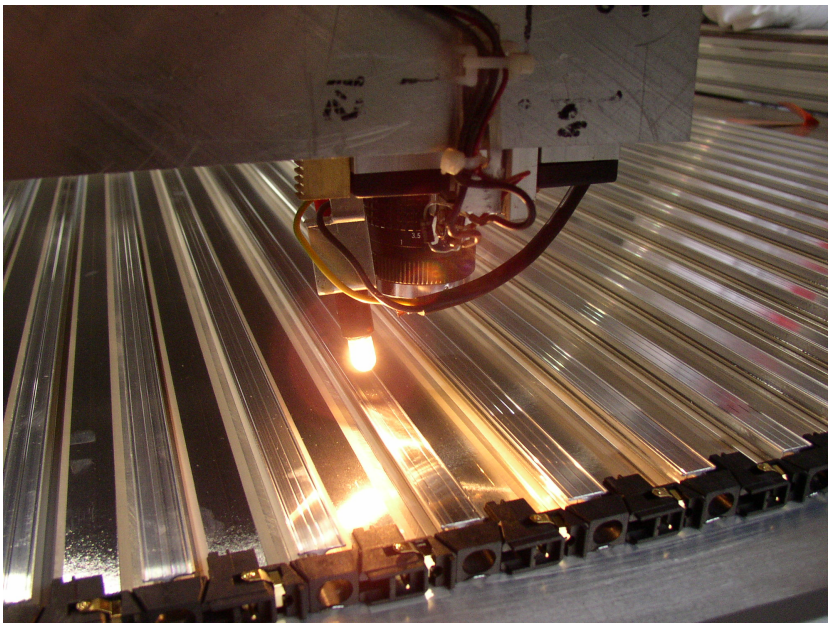


Figure 2.13: Drift tube superlayer during assembly. Anode wires can be seen between the aluminium cathodes (Source: CERN CDS)

2.4 Muon System

Since muons will play an essential role in identifying (i.e. triggering) interesting events in many physics processes (such as leptonic decays of W/Z bosons originating from Higgs particles), the CMS detector has been designed for excellent muon spectrometry resolution, redundancy and time response. Muon chambers are the outermost subdetectors in most particle physics experiments, since only muons (and neutrinos) have interaction lengths long enough to pass through calorimetry, the magnet coil and the return yoke, which guarantees low background and thus relatively easy muon identification.

In the barrel part, there are four layers of rectangular muon stations (MB1–MB4) interleaved with the return yoke toward the outside of the 12-sided CMS barrel cylinder. Each station combines several layers of Drift Tubes ('DT') with Resistive Plate Chambers ('RPC') and is in the plane tangential to the barrel cylinder and thus roughly orthogonal to muon trajectories. In each endcap part, there are four stations of trapezoidal Cathode Strip Chambers ('CSC', labels ME1–ME4) arranged in concentric rings around the beam pipe (three rings labeled ME1/1 through ME1/3 on the inside of the innermost endcap disk YE±1 and two for the other stations). RPCs have been added in the endcaps as well, providing worse spatial resolution but superior time response, they can be considered dedicated triggering detectors. Barrel DTs cover the region $0 \leq \eta < 1.3$, endcap CSCs are available for $0.9 < \eta < 2.4$, while RPCs are installed for $0 \leq \eta < 2.1$. Table 2.4 gives an overview of the key parameters of the three muon detector technologies, which will be described in more detail in the following sections.

Detector	Drift Tubes	Cathode Strip Chambers	Resistive Plate	
Function	Tracking p_T trigger BXID	Tracking p_T trigger BXID	BXID p_T trigger Resolve tracking ambiguities	
η region	0.0–1.3	0.9–2.4	0.0–2.1	
Stations	4	4	Barrel 6	Endcap 4
Layers	rϕ 8, Z 4	6	2	
Chambers	250	540	360	252
Channels	195000	Strips 273024 Wire groups 210816	80640	80642
Spatial resolution (σ)	per wire 250 μm rϕ (6/8 pts) 100 μm Z (3/4 pts) 150 μm	$r\phi$ (6 pts) 75 μm (outer CSCs) 150 μm R(6 pts): $(15-50)/\sqrt{72}$ μm	Cell size	
Time resolution	5 ns	6 ns	3 ns	
Within 20 ns window	$> 98\%$ (station) (no parallel B field)	$> 92\%$ (station)	98%	

Table 2.4: CMS muon chamber properties and statistics (Source: [23])

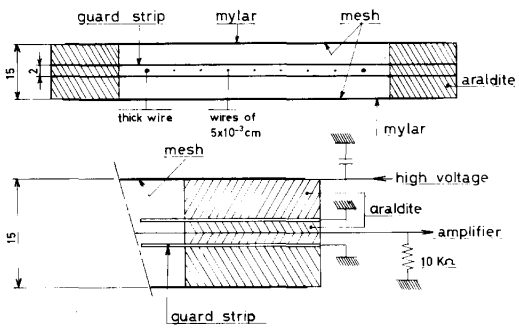
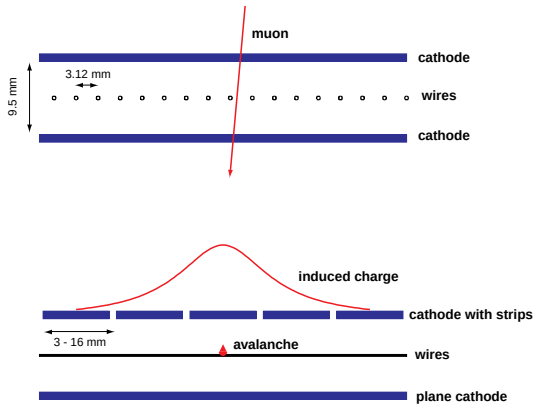


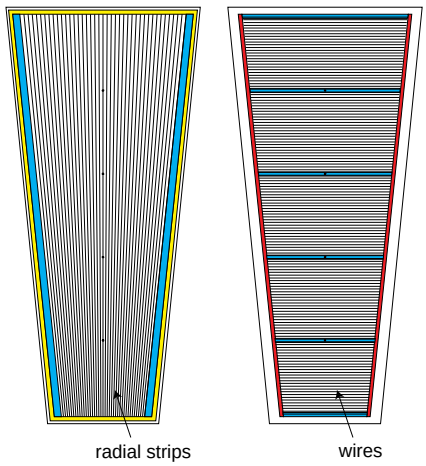
Fig. 1. Some details of the construction of the multiwire chambers.

(a) Original drawing from a 1968 Charpak publication illustrating a MWPC (Source: [17])

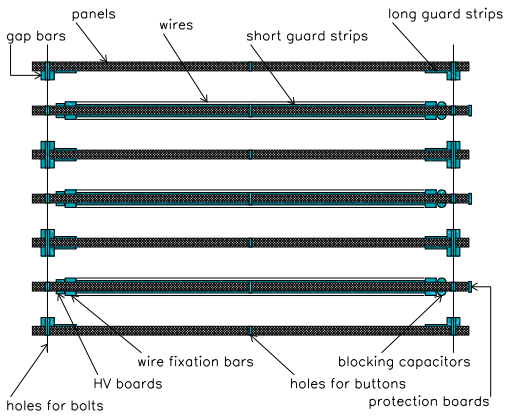


(b) Principle of coordinate measurement with a CSC

Figure 2.14: Principles of MWPC and CSC



(a) top view



(b) exploded view from the side

Figure 2.15: Schematic drawings of the CMS Cathode Strip Chambers design (Source: [23])

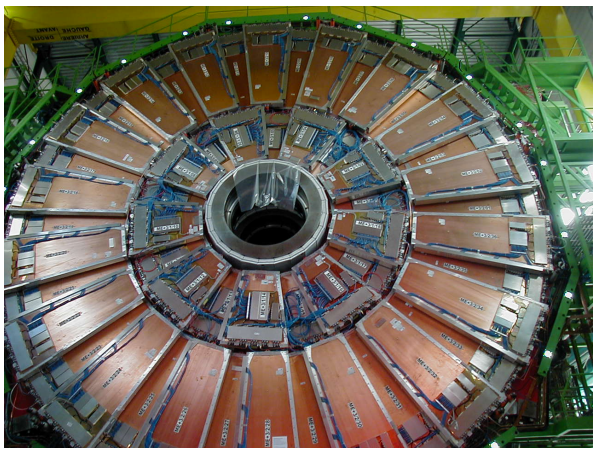


Figure 2.16: CMS endcap wheel YE2 fully equipped with CSCs (Source: CERN CDS)

2.4.1 Drift Tubes

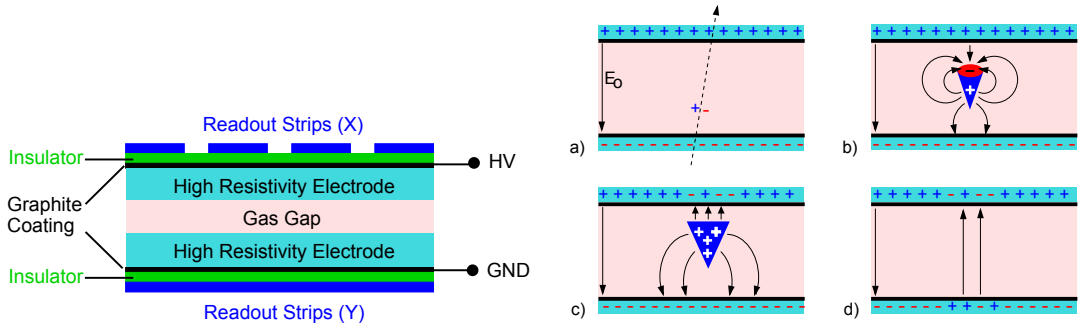
Drift chambers were chosen as the main muon tracking detectors in the barrel part because of the low expected rate, low radiation background (in fact this part of the detector has the lowest background of all, as can be seen from Fig. 1.8) and low magnetic field (the magnetic flux is almost entirely returned by the iron yoke) – and for cost reasons, due to the large area to cover. Figure 2.11 shows the cross section of one drift cell: in the middle of a 40 mm $\times$ 13 mm $\times$ 2–4 m cell volume, a 50 μm diameter wire of stainless steel acts as the anode (positive voltage), the cathode consists of two I-shaped beams of aluminium (negative voltage) which also separate a cell from the neighboring ones. The top and bottom walls are made of 2 mm aluminium plates, with an insulation layer of 0.5 mm polycarbonate plastic (Lexan) glued between the walls and the aluminium cathode. Two additional strips of copper are glued to the top and bottom (as shown in 2.11) of the cell, insulated from the walls by a 0.1 mm thick mylar tape. They have a positive bias (i.e. a positive voltage lower than the wire) and squeeze the electric field lines, improving drift time linearity and the resolution of the cell. The wires are under a tension of 2.9 N (corresponding to about 70% of the rupture value), which results in a wire sagging of 35 μm (135 μm) for 2 m (4 m) wires. A gas mixture of about 15% CO_2 and 85% Argon (at a pressure of 1020 mbar) has been chosen, ensuring both sufficient quench protection (gas discharges) and drift velocity saturation. Voltages are in the order of +3700 V for the wire, −1200 V for the cathode and +1800 V for the copper strips, the precise definition of the working point being an issue of optimization. Drift velocity is about 5.6 cm/μs, resulting in a maximum drift time of about 380 ns.

As can be seen from Figure 2.13, drift cells are assembled next to each other onto a common plate of aluminium (2 mm thick, varying size), forming a so-called layer. Four layers are combined to a Super-Layer, with the odd layers staggered by a half-cell width with respect to the even layers in order to resolve left-right ambiguities. Three Super-Layers in turn are arranged into one rigid structure called a drift tube chamber, which is finally mounted onto the iron yoke. Figure 2.12 shows the cross section of a drift tube chamber: the two outer Super-Layers have their wires oriented parallel to the beam and the magnetic field, thus measuring the trajectory in the CMS bending plane, while the inner Super-Layer is oriented perpendicularly to the others, providing measurements of the track position along the beam line. One SL is separated from the other two using an aluminium honeycomb plate, which ensures mechanical stiffness and improves angular resolution within the chamber. The outermost muon station of CMS contains drift tube chambers with only two Super-Layers measuring the ϕ coordinate.

Test beam measurements have shown a chamber efficiency of 98% (with hits onto the aluminium beams being the main cause of inefficiency); other important engineering issues include (mis-)alingment, cell aging and gas leakage (with especially nitrogen from the atmosphere affecting drift velocity). Quality tests have however confirmed the design resolution of 250 μm for a single cell and 100 μm for a full chamber.

2.4.2 Cathode Strip Chambers

In the endcap regions, particle rate (and magnetic field are considerably higher than in the barrel part, which led to the choice of Cathode Strip Chambers ('CSC') as muon detectors. CSCs are Multi-Wire Proportional Chambers (which were invented at CERN by Nobel laureate Georges Charpak in 1968, cf. [17]) with the cathode plane segmented into isolated strips.



(a) Schematic drawing of an RPC (b) Principle of operation of an RPC

Figure 2.17: Principles of RPCs. Quoting [42]: "a) Some gas atoms are ionized by the passage of a charged particle. An avalanche is started. b) The avalanche size is sufficiently large to influence the electric field in the gas gap. c) The electrons reach the anode. The ions drift much slower. d) The ions reach the cathode. The charges in the resistive layers influence the field in a small area around the position where the avalanche developed" (Source: [42])

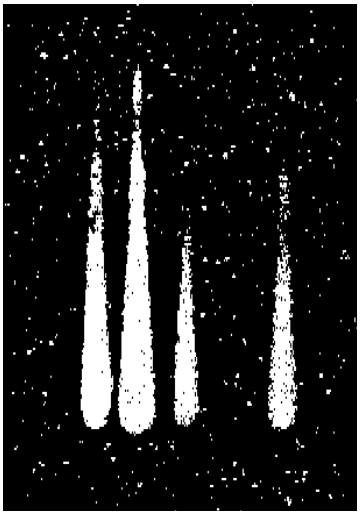


Figure 2.18: A cloud chamber photograph of electron avalanches in a parallel plate counter (Source: [42])

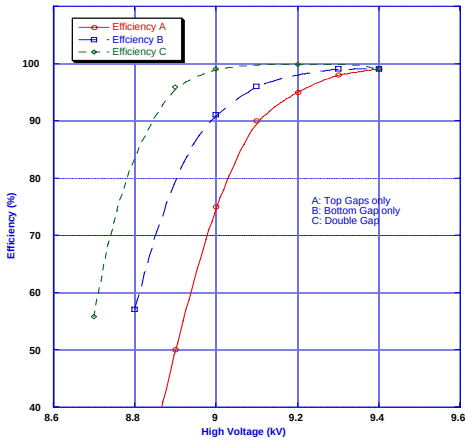


Figure 2.19: RPC efficiency as measured in a test beam (Source: [11])

As illustrated by Figure 2.14(b), an avalanche developing close to a wire induces a charge distribution in the cathode, the shape of which can be calculated precisely using electrostatics. By interpolating the fractions of charge measured in these strips, it is possible to measure the track position along the wire with a precision of 50 μm or better, thus obtaining one track coordinate (ϕ in the case of CMS) – another one (r) can of course be determined by measuring the charge reaching the wires, but resolution is rather coarse (typically 0.5 cm).

As mentioned already, the CMS forward CSC system consists of four stations per endcap. The innermost one has three concentric rings of chambers, the others two. The various chambers differ in design, since the are placed in quite different environments: while the chambers of station ME1/1 operate in an axial magnetic field of up to 3 Tesla and station ME1/2 in a highly non-uniform field of up to 1 T, the other stations experience a magnetic flux much lower. In rings 2–4, there are 36 chambers for the outer radius, each covering a sector of 10° in ϕ; the inner radius is covered by 18 chambers of 20°. Chambers overlap in ϕ to avoid dead areas, the radial gaps between the rings are not projective, thus close to 100% of the solid angle in question is covered by three or more subsequent CSC stations. In total, there are 270 chambers per endcap.

Each chamber consists of six layers of 9.5 mm gas gaps with wires and one-sided cathode plates, the cathodes being segmented into radially milled strips, 3–16 mm (2–5 mrad in ϕ) wide, depending on chamber position. Three out of the six plates, the so-called anode layers, carry wires on both sides. These 50 μm diameter wires of gold-plated tungsten at a baseline spacing of 3.12 mm are operated at a nominal voltage of 4.1 kV under a tension of about 250 g. Despite wire lengths of up to 1.5 m, no intermediate supports are needed. A wire spacing of about 3.16 mm is a compromise between stability and response time, which is needed for correct bunch-crossing assignment. Wires are grouped into bunches of 16–50 mm in width for readout. The gas mixture is Ar (40%) + CO_2 (50 %) + CF_4 (10%). The chambers at station ME1/1 are tilted by 25° in order to compensate for altered drifting in the high magnetic field and generally have smaller strip width, wire diameter, wire spacing and gas gap pitch in order to deal with the exceptionally high rate in this area (up to 1000 Hz/cm^2).

2.4.3 Resistive Plate Chambers

In order to improve muon trigger efficiency, an additional low-cost muon detection system with very fast time response was included into the CMS detector concept, namely Resistive Plate Chambers ('RPC') in both the barrel and endcap parts. RPCs are gaseous detectors consisting of two parallel plates of high resistivity material – phenolic resin (bakelite), 10^{10}–10^{11} Ωcm – separated by a gas gap of about 2 mm. The outer surfaces of the bakelite plates are coated with graphite paint, forming the HV and ground electrodes. Thereby the electrode resistivity mainly determines the rate capability, while the gap width determines the time performance. Readout is done using Al strip electrodes separated from the HV layers by a PET film. Figure 2.17(b) illustrates the principle of operation: an ionizing particle passing through the gas gap produces clusters of electrons and ions, giving rise to avalanche multiplication and drift toward the electrodes. Drift velocity for electrons is much higher than for ions, yielding a 'fast' signal in the form of an induced current on the readout electrodes. Depending on the HV voltage applied, an RPC can operate in 'streamer' or 'avalanche' modes: a larger HV voltage in streamer mode creates much stronger avalanche multiplication in the gas gap but requiring more time to collected the charges in the resistive plates, resulting in large dead time and limited rate capacity (up to 100 Hz/cm^2). In avalanche mode, as used in CMS,

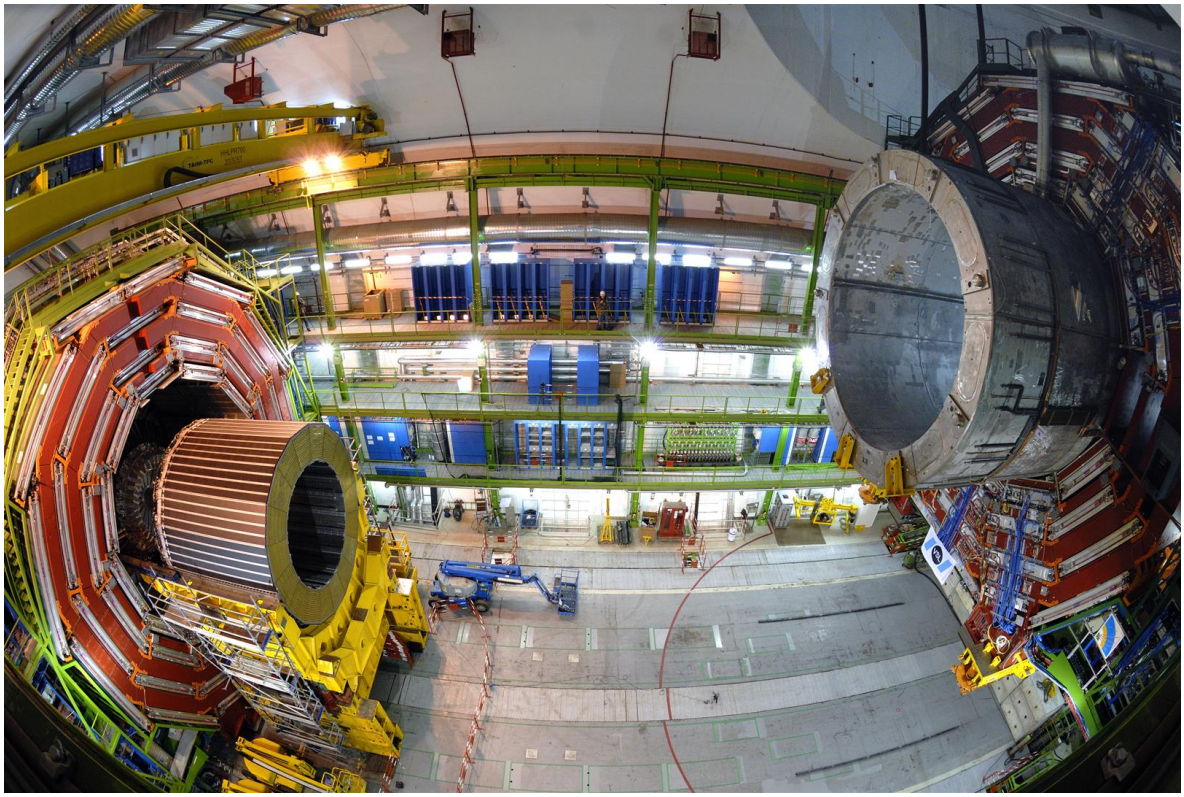


Figure 2.20: Fisheye view of the CMS experimental cavern during the lowering of the central barrel wheel YB0 on Feb. 28, 2006. YB+1 and the HB+ HCAL supermodule can be seen on the left [Source: CERN CDS]

charge multiplication is much weaker, requiring low-noise amplifiers and read-out electronics, but dead time is short enough to cope with hit rates up to 1000 Hz/cm^2, sufficient for the LHC environment.

CMS uses double-gap RPCs, composed of two gas gaps, each sandwiched between bakelite plates, with two HV electrodes on the outside and a common pick-up strip in the middle. HV voltage is 9.5 kV and a gas mixture of 95% $C_2H_2F_4$ and 5% $i - C_4H_{10}$ has been chosen. As shown in figure 2.10, two layers of RPCs have been mounted onto the barrel DT chambers at station MB1 and MB2, and one layer each for MB3 and MB4. Strips are running in parallel to the beam axis, divided into two (or three for MB4) parts and a strip width covering 5/16° in ϕ. In the endcaps, there is one RPC layer attached to each of the four CSC stations. Strips run in the radial direction here, covering the same ϕ widths.

Chapter 3

CMS Trigger and Data Acquisition

Given the expected data rate of unfiltered events at the CMS detector outputs (roughly 60 TB/s) on the one hand and the input capacity of the off-line processing and storage farms (225 MB/s) on the other, the need for efficient triggering (i.e. the selection of interesting events) becomes obvious. The CMS trigger is designed to perform the desired event rate reduction in two stages: the First Level (L1) and High Level (HLT) triggers. This chapter outlines the L1 trigger, and brief introductions are given on the High Level Trigger and the Data Acquisition (DAQ) system, which collects and stores precision data from the subdetectors when a Level-1-Accept signal is issued.

3.1 Overview of the CMS Trigger and DAQ System

At LHC design luminosity, an average of about 22 inelastic p–p interactions will take place at every bunch crossing. With a final bunch crossing frequency of 40 MHz, this results in $\mathcal{O}(10^9)$ collisions per second, each bunch crossing producing about 1 MB of zero-suppressed data to be collected from $\mathcal{O}(10^8)$ channels. Given the permanent storage input capacity mentioned above, an overall event rate reduction factor of $\mathcal{O}(10^7)$ is required, with excellent efficiency on selecting interesting physics events at very low rates. In the past, up to four layers of triggering were employed, each layer dealing with less data than the previous ones but evaluating more complex algorithms. However, more layers need more complex control logic and integration efforts, which is why in the case of CMS a two-layer architecture has been chosen: the Level 1 Trigger, a pipelined, low-latency system of custom hardware processes coarse data at an incoming event rate of 40 MHz in real time and yields a positive trigger decision (Level-1-Accept, L1A) at a maximum rate of 100 kHz (cf. [27]). The High Level Trigger (also called Event Filter), implemented as a large farm of commercial general-purpose processors ($\mathcal{O}(10^3)$ CPUs required), receives only the events accepted by the L1 trigger but has access to full detector data. The HLT runs rather complex reconstruction algorithms with a processing time ranging between 40 ms (average) and 1 s (maximum) and reduces the event rate from 100 kHz (L1-Accept rate) to 100 Hz (permanent storage input capacity), requiring a computing power of 10^6 SpecInt95 units. This means that the data acquisition system has to be capable of reading out, sorting and routing 100 GB/s from 700 on-detector readout modules to about 1000 processing units in the HLT farm. Other crucial requirements for the trigger and DAQ system include small dead time ($< 2\%$), reliability as well as scalability and flexibility under changing experimental and technological conditions.

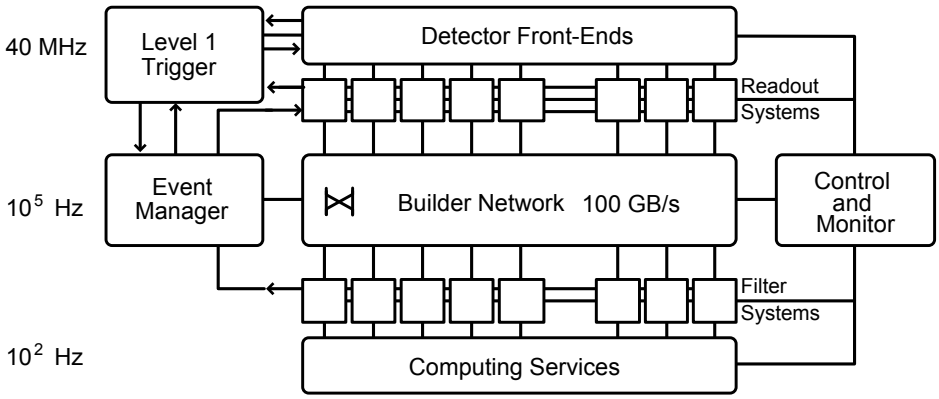


Figure 3.1: CMS trigger and data acquisition architecture (Source: [28])

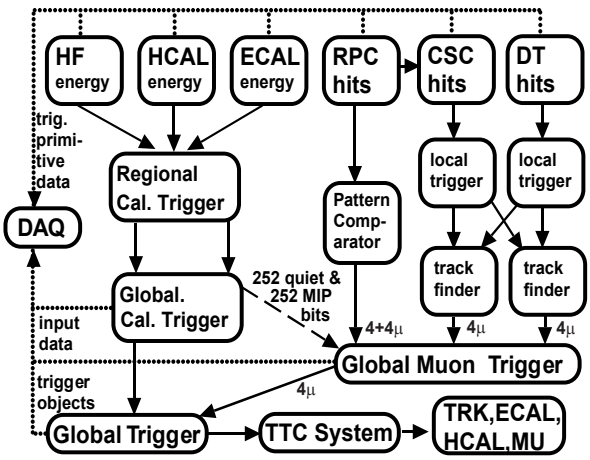


Figure 3.2: CMS Level 1 Trigger architecture and sub-units (Source: [27])

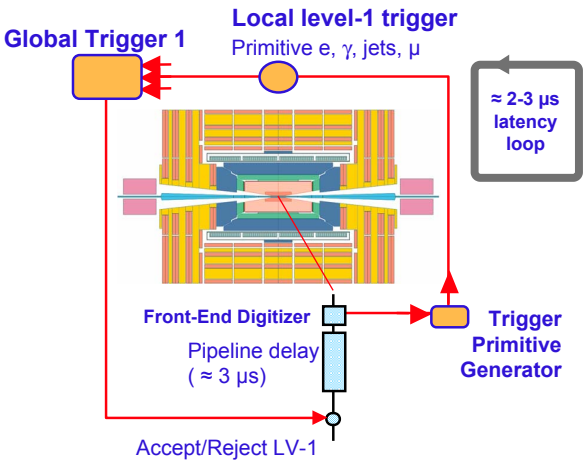


Figure 3.3: Latency loop detector front-end – L1 Trigger – detector

3.1.1 First Level Trigger

Figure 3.1 shows the position of the L1 trigger in the CMS trigger and DAQ architecture: it receives digitized detector raw data from the read-out units of the ECAL, HCAL, HF, DT, CSC and RPC subdetectors (i.e. no pixel, tracker and preshower data available to L1) and computes the L1-Accept decision in real time (i.e. every bunch crossing is being processed, processing takes constant time) within a maximum latency time (i.e. total computation and signal transfer time) of 3.2 μs or an equivalent of 128 bunch crossings ('bx'). This latency budget is determined by the size of the output buffers on the read-out boards: if no positive trigger decision arrives for 128 bx, data is discarded. Of this latency budget, 300 ns are taken up by the signal propagations delay in optical fibres from the detector to the trigger electronic racks in the neighboring counting room cavern and back again (as illustrated in Fig. 3.3, another 400 ns are needed to wait for the DT drift time before the signal can be collected. Various other delays add up to about 2 μs, leaving 1 μs for computation.

Since it is impossible to deal with such huge amounts of data within that latency budget using generic processors, the L1 trigger has been designed as a mostly custom-made, massively parallel pipelined electronic system. Most of the logic is implemented in ASICs (Application Specific Integrated Circuits, i.e. microchips with a custom but constant design) and FPGAs (Field Programmable Gate Arrays, i.e. microchips with re-programmable logic structures and freely assignable I/O pins), with ASICs being faster and radiation-hard but requiring longer design cycles. FPGAs have limited numbers of gates and cannot be clocked too fast, although their specifications have evolved very fast during the past few years. However their great flexibility through firmware re-flashing is a very convenient feature, making them the technology of choice in the trigger where radiation hardness is not an issue. Wherever possible processing time is being saved by the use of LUTs (Lookup Tables, i.e. memory tables with pre-computed output values for every input value possible). Most logic cards and crates don't use buses to transfer data (which would need several clock cycles for a transfer and extra bus logic), but rather (multiplexed) point-to-point links on the motherboards and crate backplanes (into which the motherboards are plugged). Wherever enough space is available for large connectors, parallel connections using copper wires are preferred to serial buses, since the latter require extra latency for parallel-to-serial conversion, error correction and high-frequency transfer overhead. The entire L1 system is synchronized using a common clock signal of 40.08 MHz that is distributed using optical fibers along with status signals throughout the entire detector from the LHC control center. The entire L1 trigger consists of an estimated 4000 motherboards in 200 crates and 40 racks, located in the counting room cavern USC55 right next to the detector cavern. Mostly 9U crates with custom backplanes are being used, VME64x bus interfaces allow for configuration, testing, monitoring and debugging from a networked crate controller PC in the rack. Every rack is equipped with water cooling and a temperature/voltage monitoring system for safety. The L1 trigger is divided into three major subsystems (cf. figure 3.2), the Calorimeter, Muon and Global Triggers which are described in more detail later in this chapter. It should be noted that the data at the L1 trigger inputs is already the result of analog and digital processing, which includes pulse shaping, amplification, noise suppression, application of local trigger thresholds for individual detector cells, analog-to-digital conversion and possibly translation to some definable scale. At the trigger input stage, these values represent local quantities like the energy deposited in all the individual calorimeter towers (blocks of three times three crystals) or hit coordinates or vectors in the muon chambers. The regional triggers subsequently apply coarse track

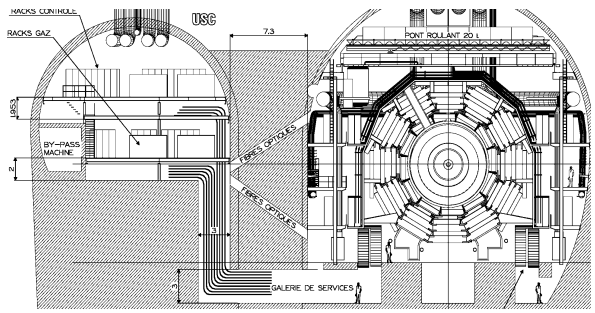


Figure 3.4: Cross section of the CMS experimental cavern UXC55 and the neighboring counting room USC55. Trigger data is transmitted through two direct but inclined tunnels (to absorb radiation) between the caverns to minimize cable length (Source: [27])

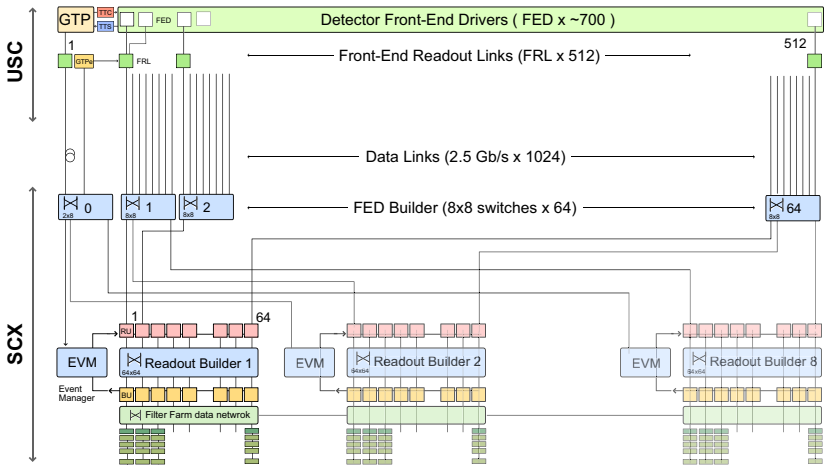


Figure 3.5: The CMS DAQ and event building system

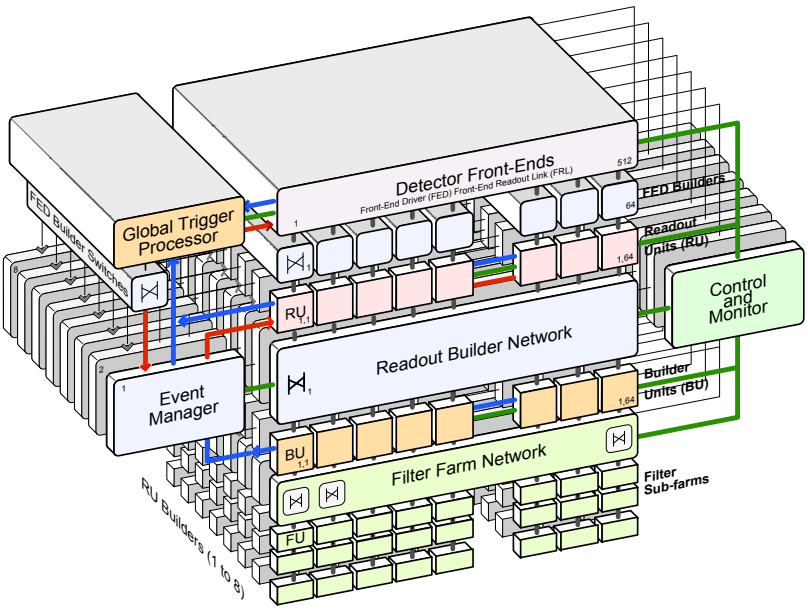


Figure 3.6: 'Three-dimensional' view of the full Event Builder system (Source: [28])

fitting and jet finding algorithms, computing 'physics' objects like muons, photons, electrons or jets with assigned transverse momentum (p_T) or energy E_T, sign of charge, and location information (η and ϕ coordinates). The global calorimeter (GCT) and global muon triggers (GMT) receive these objects from the regional triggers, try to combine objects that have been identified by two redundant subsystems, assign quality codes and deliver a sorted set of objects to the Global Trigger (GT). The GT is in principle a large programmable AND-OR network which compares these objects to conditions in a trigger menu, requiring e.g. two muons with momenta larger than some threshold and back-to-back trajectories. If the trigger conditions are satisfied, the GT issues a L1-Accept signal to all the sub-detector readout boards, the trigger itself and the DAQ system, which then retrieves all the detector data from the local buffers, including some L1 trigger intermediate results for monitoring purposes.

3.1.2 Data Acquisition

During L1 processing time, about 700 on-detector buffers store the full event data in analog or digital form (cf. figure 3.7); on L1-Accept, data is digitized (if necessary) and passed on from the on-detector front-end drivers (FEDs) to the the first layer of the DAQ system, 512 front-end readout link (FRL) cards, which may combine data from up to two FEDs. Two 2.5 Gb/s Myrinet optical fibers per FRL transmit these event fragments to one of 64 FED builder units in the surface counting room. A FED builder is basically an 8×8 Myrinet switch which pre-assembles fragments from 8 FRLs and sends them to one or more readout units (RU). The RUs again buffer data and interface with the 'readout builder' switching network, the task of which is to assemble event fragments from the readout columns (one readout column corresponds to one FRL) into full event records in the memory of the so-called builder unit (BU) PCs at the outputs of the switch. These BUs in turn are connected to the filter farm network (commercial Gigabit Ethernet using twisted-pair copper wires) that distributes full events to filter unit (FU) PCs. These FUs form the High Level Trigger to be described in the next chapter. As illustrated by figure 3.6, the DAQ system has been designed in a modular and extensible way that can be pictured as a 3D structure: at LHC startup with low luminosity, only one 'slice' of RUs, readout builder switches, BUs and FUs will be available. Until design luminosity is reached a few years after startup, all the slices will be subsequently installed using up-to-date hardware. The 8 outputs of each of the 64 FED builder switches will be connected to one RU in each slice in order to distribute data as more than one slice becomes available.

Tests and simulations have been carried out on a preseries system, which corresponds roughly to one slice of the final DAQ and consists of over 100 dual 2.6 GHz Xeon PCs with dual-port Myrinet or quad-port Gigabit Ethernet NICs (or both in case of the BU PCs which form bridges between the two physical networks). Results show that the design throughput of 240 MByte/s can be achieved using Gigabit-Ethernet in the filter farm network [50], using optimizations in GNU/Linux TCP/IP networking, such as altered frame sizes and an asynchronous TCP package to avoid head-of-line blocking when several hosts are trying to send data to the same network interface of another host.

The data flow protocol in the DAQ system follows a trigger-driven ('push') principle from the FEDs to the RUs: when a L1A signal is issued to the FEDs, event fragments are transferred to the RUs and written into buffers there. To avoid buffer overflow, there is a backpressure loop to the Global Trigger: if buffer occupancy in an FED passes over some 'high water mark', the FED will issue a warning via the synchronous trigger throttling

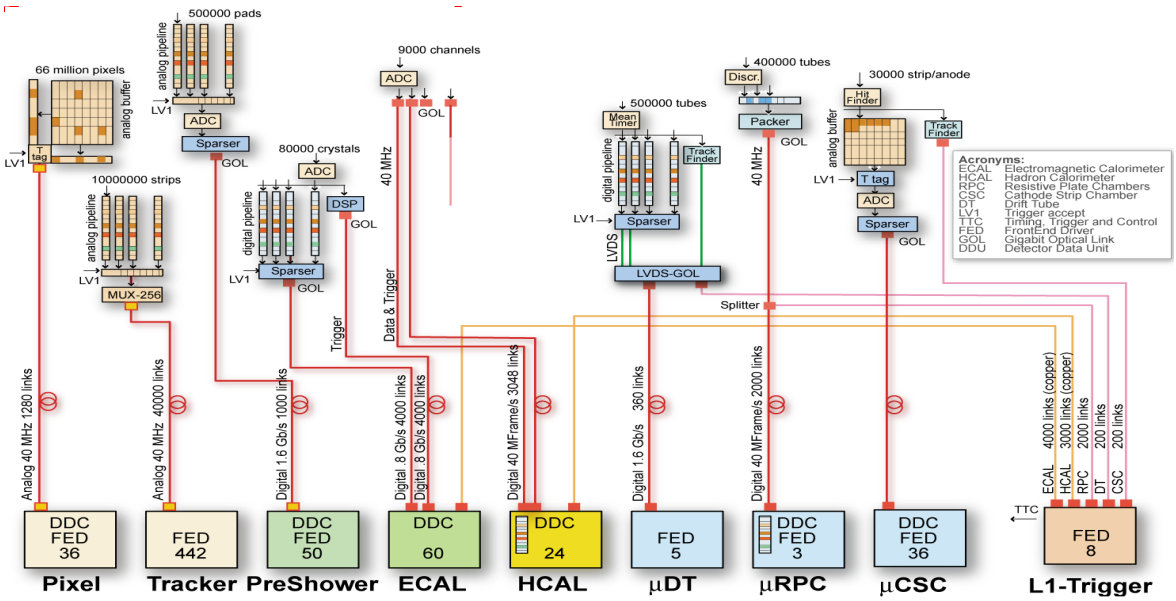


Figure 3.7: Various on-detector buffer and readout architectures (Source: [60])

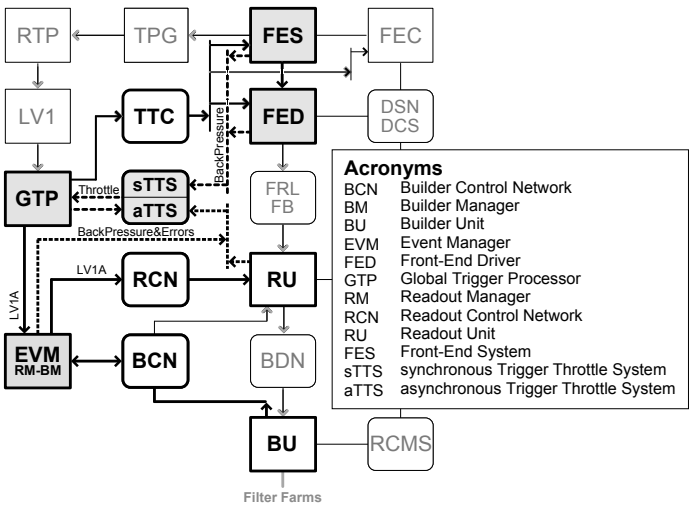


Figure 3.8: sTTS and aTTS integration with major data flow components (Source: [28]

system (sTTS) to the GT, which in turn will reduce the trigger rate until the warning is cleared. The sTTS is clocked synchronously to the L1 Trigger, thus allowing for very fast feedback. The RU computers (in the memories of which event super-fragments are assembled out of the FED event fragments) in turn are connected to the aTTS (asynchronous TTS; not clocked synchronously with the trigger) system, which transmits similar messages to the GT, again causing trigger rate reduction, but with higher latency than the sTTS. This is sufficient because of the large buffer sizes in the RUs.

From the RUs to the BUs (and filter farm), data flow follows a 'pull' principle: when a BU is ready to receive a new event, it issues an 'allocate' message to the Event Manager system (EVM), which replies with the logical event id of an event available from the RUs. The BU then requests the event from the RU, assembles it in its memory and passes it on to a filter sub-farm. The EVM supervises all event traffic in the RU builder: it receives bunch crossing, orbit number, L1A number and GPS time (and other data) from the Global Trigger and associates this data with a logical event id, thus providing resistance to a reset or wrap-around of the event number, which is maintained in local counters in the FEDs and the L1 trigger. The event id is distributed to the RUs via the Readout Control Network and passed on to a BU upon 'allocate' request as described above. The EVM is in charge of allocating each event id to one BUs in normal operation or more BUs for debugging purposes.

In case any part of the DAQ system detects a synchronisation error (e.g. caused by a buffer overflow or counter error) by comparing the event numbers of the fragments to be assembled, the GT is notified via TTS, and a resync-command is sent to all components of the trigger and the FEDs via the Trigger Timing and Control (TTC) system, an electro-optical network distributing clock, bunch counter reset, L1A and other control commands to the trigger and subdetector electronics. It is also possible to partition the Trigger/DAQ system for commissioning and testing: in this mode, the 32 (fixed) detector partitions (i.e. parts of subdetectors) can be grouped into 8 partition groups (also called TTC partitions), which operate independently and can receive different trigger signals (e.g. periodic 'technical' triggers at arbitrary rates; start/stop run signals) and can be assigned to one of 8 DAQ partitions, which further provide independent data flow until the BUs. In normal data taking mode however, all components of the detector will belong to the same single partition.

Software development constitutes a large part of the DAQ project: The XDAQ framework has been developed as a basis for event building and monitoring and is based on a distributed, event-driven architecture; it is described in more detail in section 6.1.1. The Run Control and Monitor System (RCMS) [13] is in charge of controlling and monitoring the entire CMS experiment during data taking, providing (remote) access to configurations of high and low voltages, cooling, gas and liquid services, safety systems and the like, traditionally referred to as 'Slow Controls'. The RCMS sub-systems in charge of theses services is called Detector Control System DCS. During data taking, RCMS will act as a master to DCS, the event builder and event filter subsystems as well as the Trigger Supervisor, which controls the L1 trigger subsystem (see section 6.1.2). RCMS also interfaces with the computing services to log all the calibration, configuration and run condition data. Communication between these various subsystems is based on the XDAQ framework and a peer-to-peer messaging architecture. Cf. the DAQ Technical Design Report [28] for more details on the CMS DAQ system.

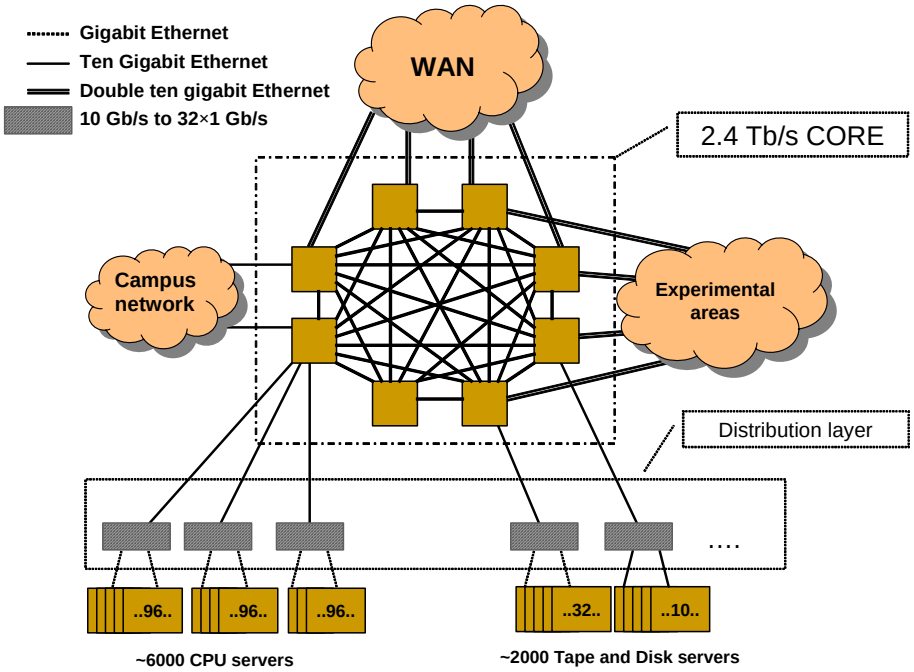


Figure 3.9: Layout of the LHC Computing Grid Tier-0 Network (Source: [40])

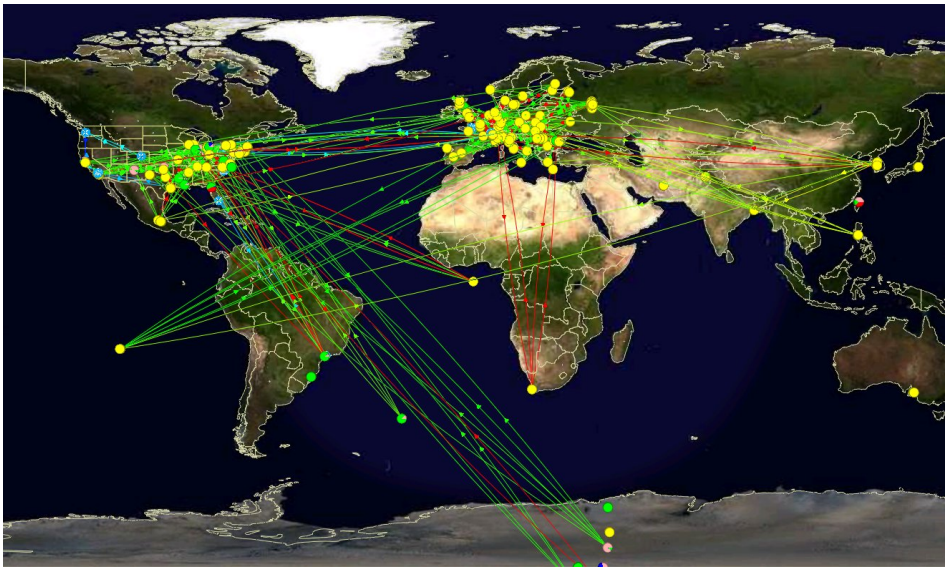


Figure 3.10: Sample of LCG node activity and data transfers (Source: screenshot of monALISA monitoring application http://monalisa.caltech.edu/

3.1.3 High Level Trigger and LHC Computing Grid

As mentioned earlier, for reasons of flexibility and scalability, the CMS collaboration has chosen to implement all higher-level triggers as software running on an $\mathcal{O}(10^3)$ CPU GNU/Linux cluster, called the filter farm, in real time. Although there are no explicit Level 2/3 triggers, the HLT software is designed to discard an event as quickly as possible: in a first pass, only calorimeter and muon detector data is considered in the event reconstruction algorithms. If the event is not discarded, partial and eventually full tracker and vertex data is being included. However it is perfectly possible (and intended) for these steps to be interchanged or replaced entirely in the future should new algorithms with altered performance characteristics be developed. A key requirement for the HLT is to be inclusive, i.e. not to discard events that might be potentially useful in 'exotic' searches for unexpected physics events: if the HLT discards an event, it is lost forever (apart from a very small number that is kept for HLT performance analysis). HLT and offline analysis (i.e. full complexity & accuracy physics analysis, not in real-time) are based on the same software packages and reconstruction algorithms, namely the CMSSW framework. The HLT however has no knowledge of the precise detector conditions (misalignment, exact temperatures, voltages etc.). Electrons, photons, muons, jets, neutrinos, t-leptons and b-jets are the physics objects to be identified and reconstructed at this level, with special conditions identifying 'gold-plated' i.e. prototypical and unambiguous events which are suited for immediate analysis by the physicists on shift in the control room via an 'express line' channel.

The HLT accepts events (raw event data size is about 1.5 MB) at an average output rate of 150 Hz, resulting in a sustained 225 MB/s data stream (not including calibration and other non-event data) to a multi-petabyte object database at the 'Tier-0' computing center (located at CERN), with an intermediate 50 TB disk buffer at the experiment site to prevent data loss in case of uplink interruptions. The Tier-0 center is the heart of the LHC Computing Grid (LCG), a world-wide distributed computing system that is used by all the other LHC experiments as well. Tier-0 will eventually provide about 6000 CPU servers and 2000 disk/tape servers, running first-pass reconstruction algorithms. Tier-0 will also distribute a second copy of all data to the ∼6 CMS Tier-1 centers (11 altogether, most used by more than one experiment), large computing facilities that take responsibility of the data and run complex analysis jobs that require access to large amounts of data. About 100 Tier-2 centers will provide capacity for end-user analysis jobs and Monte Carlo event generation and simulation and request data as required from the Tier-1 centers. The Tier-3 layer finally consists of small clusters or individual PCs at the collaborating institutes, which will as well contribute to analysis and reconstruction. Requirements on the LCG are estimated to sum up to 140 million SPECint2000 CPU capacity units [6], to about 60 PB of disk storage and 50 PB of mass storage and are expected to extend considerably during the 15 years of scheduled LHC operation [40]. However, it is unfortunately beyond the scope of this thesis to further go into the details of online and offline reconstruction and analysis or the LCG.

3.2 L1 Calorimeter Trigger

The purpose of the L1 Calorimeter Trigger is to find electron, photon and jet candidates in coarse data from the calorimeters (HB, HE, HF, EB, EE) and calculate the total deposited transverse energy and missing transverse energy. It sorts these candidates (primarily based on E_T, but may include other factors such as isolation) and delivers the best four of each

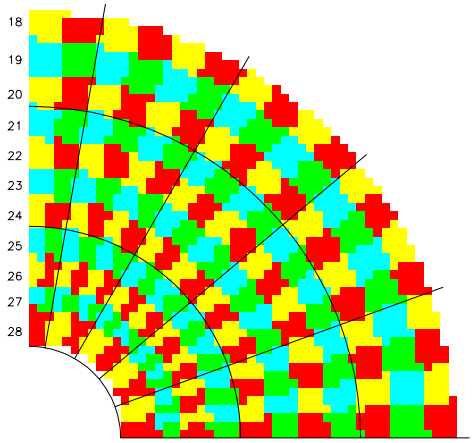


Figure 3.11: Calorimeter trigger tower layout in the ECAL endcap (Source: [27])

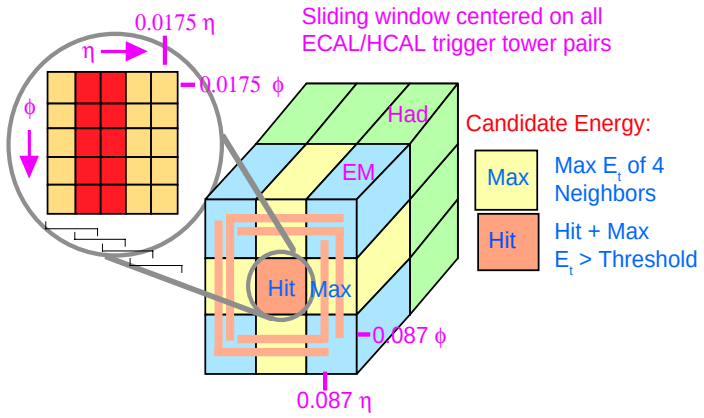


Figure 3.12: RCT electron/photon trigger algorithm (Source: [27])

(a) (b)

Figure 3.13: A full RCT crate without covers, front and rear view

kind to the Global Trigger. It additionally sends arrays of bits to the Global Muon Trigger indicating the passage of a minimum-ionizing particle (MIP) and minimum energy deposit (Quiet) in the corresponding region. Since tracker data is not available to the L1 trigger, it is impossible at this stage to distinguish photons from electrons. The Calorimeter Trigger performs these tasks in three major subsystems:

Trigger Primitive Generator: These on-detector electronics boards are tightly integrated with the calorimeter readout systems and generate the digital data available to the rest of the trigger by analyzing the 40 MHz samples from calorimeter transverse energy deposits. Since these detector pulses span several bunch crossing intervals, the TPGs first filter the pulse and finds its peak, thus assigning an unambiguous bunch crossing to the measurement. Next, energy sums are computed over so-called trigger towers to reduce data granularity (mainly for reasons of cost and complexity of the subsequent electronics). For the ECAL, one tower consists of 5×5 crystals in the barrel (covering $\Delta\eta \times \Delta\phi = 0.087 \times 0.087$) and a varying number of crystals in the endcap (see figure 3.11), where the crystals follow a $x - y$ geometry: up to $\eta \approx 2$ a tower covers $\Delta\eta \times \Delta\phi = 0.087 \times 0.087$, the η dimension of the towers grows with increasing η, the number of crystals per tower varies between 25 (at $|\eta| \approx 1.5$) and 10 (at $|\eta| \approx 2.8$). Up to $|\eta| = 1.74$, the trigger towers match the HCAL physical towers in size, above the HCAL towers have twice the dimension in ϕ, and transverse energy is divided by two and assigned to the two corresponding trigger towers. Two lateral segments of the HCAL (three in some parts of the endcap) are included into the energy summation for the trigger. Apart from tower energy sums, trigger primitive data also includes fine grain bits used for background rejection which indicate the lateral dimension of a shower in the ECAL. In the HCAL, that bit instead is used to indicate passage of a minimum ionizing particle, defined an energy deposit within some programmable window. Energy sums are converted to programmable non-linear scales to improve trigger resolution while keeping data widths low. TPG output data finally is transferred to the Regional Calorimeter Trigger in the adjacent underground counting room via high-speed serial copper wire links.

The *Regional Calorimeter Trigger* consists of 20 regional processor crates containing seven rear mounted receiver cards, seven front mounted 'electron isolation cards' and one front mounted 'jet summary card', as well as a 'timing and control card' and a VME-PCI interface (cf. figure 3.13). It identifies electrons or photons (which can't be distinguished in the trigger) by finding large energy deposits in one or two adjacent ECAL trigger cells and examining the lateral (fine grain bits) and longitudinal (e.g. HCAL to ECAL energy deposit less than 5%) shower dimensions in a 3×3 tower windows that is slid across all the towers (see figure 3.12). The best four electrons/photons as well as the best four isolated ones (i.e. low energy deposit in surrounding towers) are identified per wedge ($\Delta\eta \times \Delta\phi = 3 \times 0.7$) and forwarded to the Global Calorimeter Trigger. Additionally, the best four central, forward and τ-tagged jets are identified using similar algorithms and forwarded. In the RCT, the trigger towers are combined to form calorimeter regions, which constitute the granularity of the η and ϕ coordinates for the trigger objects at the Calorimeter Trigger output (including MIP/Quiet bits to GMT): up to $|\eta| < 3$, 4×4 towers constitute a region; above, one HF tower also acts as a full region. Transverse energy (E_T) sums are computed for each region, and their MIP (if all HCAL towers in the region have the fine grained bit set) and Quiet bits (if E_T for each tower in the region doesn't exceed some threshold, e.g. 5 GeV) are set.

The *Global Calorimeter Trigger* (one crate) finally receives all these objects, sorts the electron/photon candidates and sends the best four isolated and non-isolated ones to the Global Trigger. MIP and Quiet bits are forwarded to the Global Muon Trigger, and total E_T

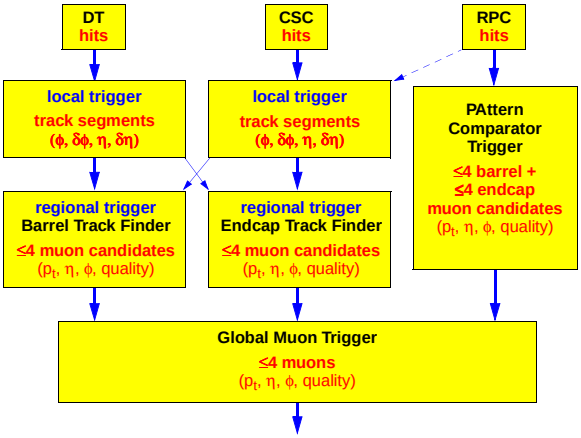


Figure 3.14: Muon trigger data flow (Source: [27])

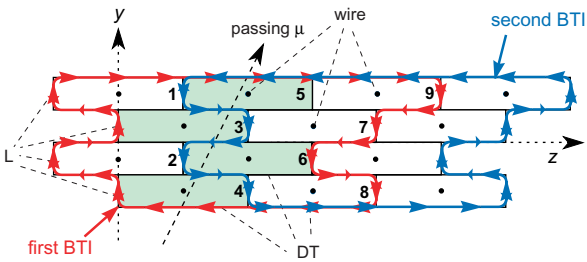


Figure 3.15: DT local trigger BTI wire groups (Source: [16])

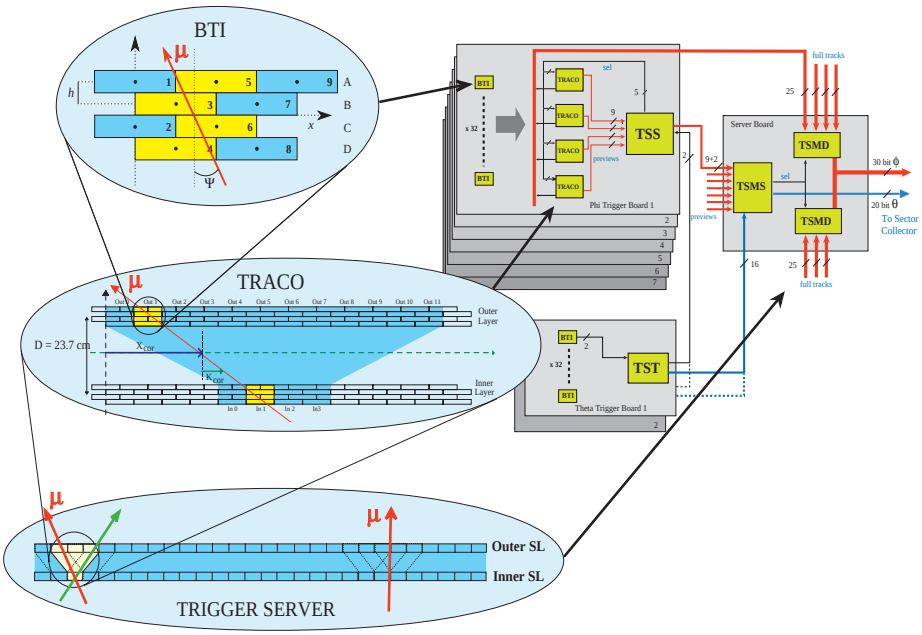


Figure 3.16: Overview of the DT Local Trigger: functions of BTIs, TRACOs and TS (Source: [27])

as well as missing E_T is calculated and sent to GT, as well as the total transverse energy of all jets (H_T).

3.3 L1 Muon Trigger

The highly redundant muon system of CMS is exploited by 3 separate trigger data paths, one for each muon sub-detector, the results of which are combined by the Global Muon Trigger (see chapter 4), taking into account the respective reconstruction quality measures. In combination, this results in robust background suppression, low ghosting and good efficiency over a wide η-range: while the wire chamber systems (DT, CSC) provide good position and thus momentum resolution and background rejection (due to their multi-layered structure), the fast and highly granular RPCs allow for unambiguous bunch crossing assignment and operation in a high-rate environment. Another advantage of these complementary systems (muon chambers vs. dedicated trigger detectors) is the possibility for cross-checking their results and quickly detecting problems. Figure 3.14 illustrates the two different approaches for muon candidate finding: while the DT and CSC subsystems first find track segments in the various muon stations (i.e. hit coordinates with momentum vectors) and then try to combine these segments into tracks, the RPC trigger follows a pattern comparator approach which is based on the spatial and time coincidence of hits compatible with a possible muon track. In the following sections, the various Local and Regional Muon Trigger components will be outlined, the Global Muon Trigger – the main subject of this thesis – is described in great detail in chapter 4.

3.3.1 Drift Tube Local Trigger

Mounted onto the sides of the Drift Tubes in the barrel, the DT Local Triggers detect charged particle track segments within one DT chamber. The θ and ϕ super-layers (cf. section 2.4.1) are processed separately, thus yielding track segments in the $r-\phi$ and $r-\theta$ planes. Processing starts with the *Bunch and Track Identifier* stage (BTI): one BTI ASIC is connected to a group of nine drift cells within the same superlayer (as shown in figure 3.15), two neighboring ASICs overlap. The BTIs look for coincident hits in all four (high quality trigger HTRG) respectively three layers (low quality trigger LTRG) of a superlayer that lie approximately within a straight line, using a mean-timer technique; they also assign a parent bunch crossing of the candidate track. These track candidates are sent to the *Track Correlator* (TRACO) logic blocks, which combine results from the two ϕ superlayers in one DT chamber: each TRACO is connected to four BTIs of the inner superlayer and 12 BTIs of the outer one. It compares combinations of track segments with pre-computed patterns for compatibility with a muon crossing angle, improving resolution by exploiting the 23.7 cm lever arm between the two superlayers; the TRACOs sort track candidates by closeness to the radial direction (i.e. high momentum) and convert the local coordinates to the CMS reference frame. The *Trigger Server* (TS) receives these track candidates and direct inputs from the θ-superlayer BTIs, which are ORed together in groups of nine tubes, resulting in 7 bits indicating a segment in the respective group. The TS also selects the two highest p_T (lowest angle against radial direction) track candidates in the chamber and forwards them to the *Sector Collector* (SC) boards, which are housed in a crate attached to the outermost muon stations and collect all the track segments from a 30° sector of one of the barrel wheels. The SCs in turn convert these LVDS (Low Voltage

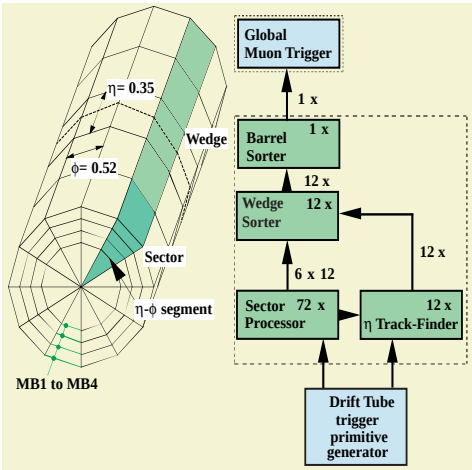


Figure 3.17: Logical segmentation and block diagram of the DT Track-Finder system (Source: [27])

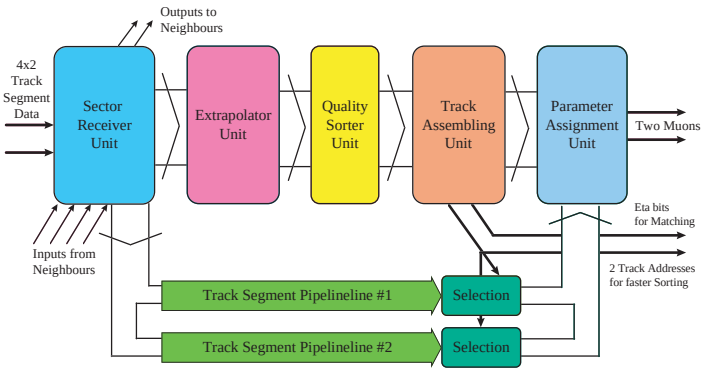


Figure 3.18: DT Track Finder Processor block diagram (Source: [27])

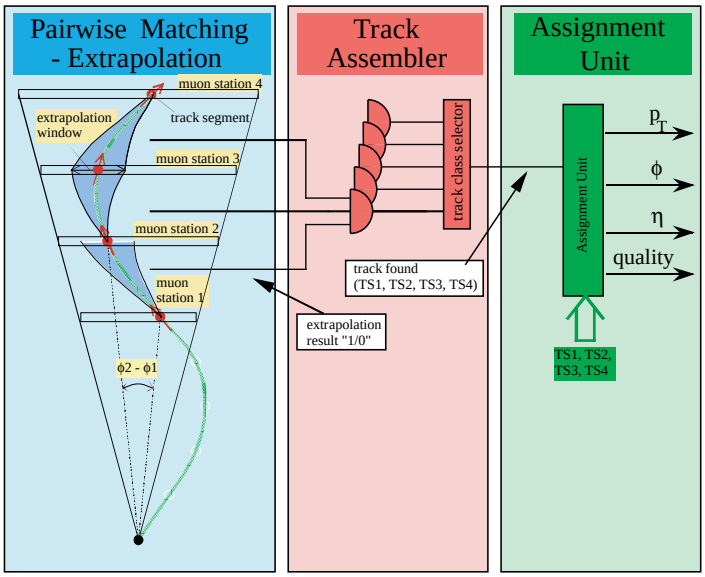


Figure 3.19: Principle of the DT Track Finder algorithm (Source: [27])

Differential Signal) signals to optical fibre signals and transmit them to the DT Regional Triggers in the adjacent counting room.

3.3.2 DT Track Finder

The DT Track Finder System (DTTF) [16, 39, 35] connects the track segments delivered from the DT Local Trigger into full muon tracks, combines tracks spanning more than one sector, assigns transverse momentum p_T, sign of charge, η and ϕ coordinates and a data word indicating reconstruction quality. In the DT–CSC overlap region ($0.9 \leq |\eta| \leq 1.3$), it also considers segment data delivered from the CSC Local Trigger.

The DTTF is organized into 12 wedges, each consisting of the five wheel sectors with the same ϕ ranges (cf. figure 3.17), each wedge corresponding to one crate in the underground counting room. Each wedge crate contains six *DTTF Processor Boards* (one for each wheel, with the central wheel logically split into two to simplify the design and account for tracks in $+z$ and $-z$ directions separately) which receive the track segments from the on-detector Sector Collector boards and yield up to two best muon tracks in the $r-\phi$ plane for the sector. One *Eta Track Finder* board per wedge crate similarly finds tracks in the $r-z$ plane. All these muons (fully reconstructed tracks with physics and quality parameters assigned) are sent to the *Wedge Sorter Board* (one per crate), which matches duplicates and delivers the best two muons per wedge to the final *Barrel Sorter Board* (housed in one of the wedge crates). There tracks found in two adjacent wedges are again merged, and the best four muons overall (ranked by reconstruction quality and p_T) are selected and delivered via LVDS channel link parallel cables to the GMT.

The DTTF Processor Boards perform all the track assembly logic, they are divided into the following logic units (see figure 3.18): The *Sector Receiver Unit* receives track segments via optical links, synchronizes them, forwards them to the neighboring Sector Receiver Units and in turn receives segments from the neighboring sectors. Next, the *Extrapolation Unit* links track segments as illustrated in figure 3.19: using lookup tables, position and direction of track segments from one station are extrapolated to a target station (in the combinations 1→2, 1→3, 1→4, 2→3, 2→4 and 4→3) . If a segment from the target station is found within some window $\Delta\phi$, the extrapolation is considered successful. Since muon trajectories can cross section and wheel boundaries, each source segment is compared to the two target segments of the six chambers surrounding the source chamber: two in the same sector, the two adjacent ones in ϕ and the three neighbors in the next wheel (away from the interaction point; since bending is in the $r-\phi$ plane, no muon track can be bent back towards the interaction point in z direction). Due to the magnetic field configuration the bend angle has a zero crossing near station 3; therefore that station is not used as a source. Instead, track segments are projected backwards from the outermost stations of the next wheel to the station 3 segments in each wheel. The sector processors for the outermost wheels slightly differ from the others in that they receive segment data from endcap CSC station 1/3 to handle the overlap between the two muon systems. These segments are treated as if they were coming from neighboring DT stations but with different extrapolations to account for the highly inhomogeneous field in the endcaps.

Since it is highly unlikely for a source segment to fit more then two target segments (but two times six extrapolations are performed, the subsequent *Quality Sorter Unit* selects the best two extrapolated target segments for each source segment by comparing the quality bits of the track segments and the deviation from the extrapolated target position. Next, the

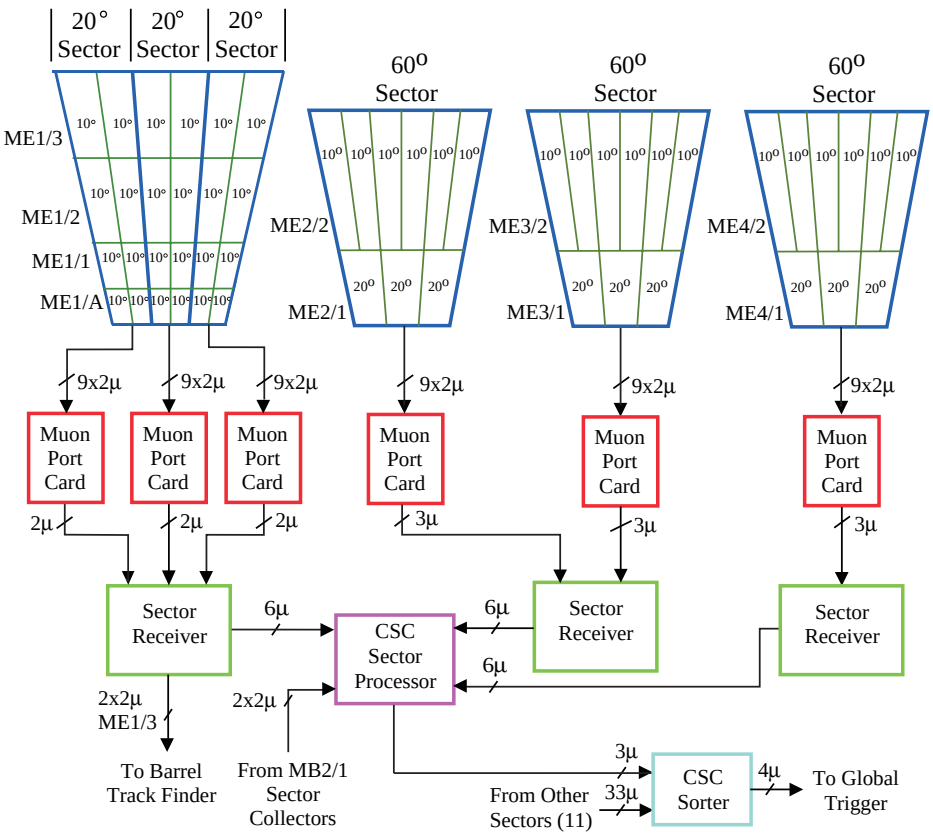


Figure 3.20: The data mapping from CSC chamber-level LCT information into 60° sector information for the CSC Track Finder (Source: [27])

Track Assembling Unit attempts to link series of subsequent extrapolations (e.g. a successful 1→2 extrapolation with successful 2→3 and 3→4 ones). Other conditions are applied as well, such as the requirement for a 1→3 extrapolation compatible with subsequent 1→2 and 1→3 ones (in order to suppress background). Tracks are assigned one of 11 quality classes (such as 'T1234', indicating successful extrapolations to all stations). As a next step in the pipelined architecture, the best track is selected and all the others are tested for being sub-tracks of the best one and discarded if they are; the same is done again with the remaining tracks starting with the second-best one. The final *Assignment Unit* assigns the physical parameters mentioned at the beginning of this section to the selected best two tracks by comparing the original track segment data (meanwhile stored in a pipeline) to lookup tables. E.g. p_T is determined by using the difference in the ϕ coordinate of the two innermost track segments, a 5 bit-encoded value between 0 and 140 GeV/c is assigned on the basis of 90% efficiency: if in the Global Trigger a certain p_T threshold is applied, this assignment strategy ensures that 90% of the muons with a true transverse momentum greater or equal to the threshold are accepted.

3.3.3 CSC Local Trigger

For the trigger, the anode wires of the CSC chambers in the endcaps mainly serve the purpose of determining the bunch crossing of a muon in question (and the r coordinate), while the cathode strips are used to extract the ϕ coordinate of a muon and its angle of passage ϕ_b. *Anode Front-End Boards* [27] connected to the anode wires amplify and digitize their signals in groups of 10–15 wires (to reduce channel count) and send them to *Anode LCT boards* (LCT standing for Local Charged Track), which compares the pattern of hits in the six layers of a CSC chamber to template patterns, requiring multi-layer coincidence and compatibility with an origin at the interaction point. The bunch crossing number is assigned using a two-threshold coincidence technique: if a first coincidence threshold (say, two layers) is reached, the bunch crossing number is recorded and the circuitry waits for the maximum drift time of 50 ns. If after this time the second threshold has been reached (say, coincident hits in four layers), the recorded hit pattern is considered valid. Hits are compared to template hit patterns and assigned a quality rank, the best two result pattern numbers (with wire group and bx number) are transferred to the *Trigger Motherboard (TMB)* in a crate mounted onto the outside of the endcap disk.

The cathode signals are pre-processed in a very similar manner by the *Cathode Front End Boards*, additionally half-strip precision is attained by interpolating between the charges deposited in adjacent strips. The *Cathode LCT (CLCT)* circuits analyze the half-strip hit patterns for consistency with high p_T muon tracks. The bend angle is coarsely determined from the shape of the hit pattern.

The TMB circuits (located on the same boards as the CLCT) check for time coincidence of anode and cathode patterns. If there is an ambiguity (due to more patterns coinciding, the TMB uses information from the RPCs to resolve it. Two (or less) LCTs per chamber are sent to the *Muon Port Cards*, which collect LCTs from a whole CSC sector (30° for the innermost station, 60° for the others) and send them to the CSC Track Finder in the counting room via optical links.

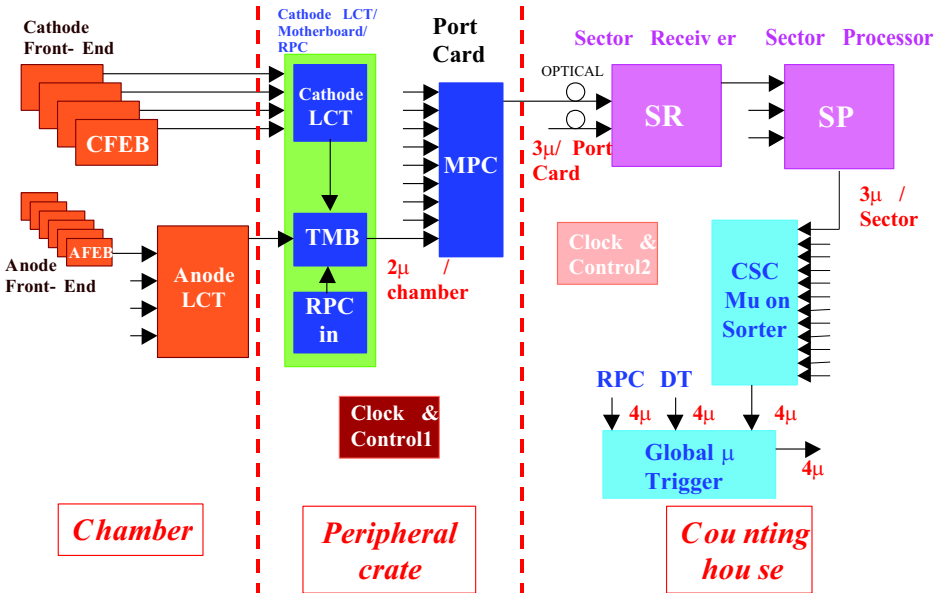


Figure 3.21: Overview of the CSC Trigger system(Source: [54])

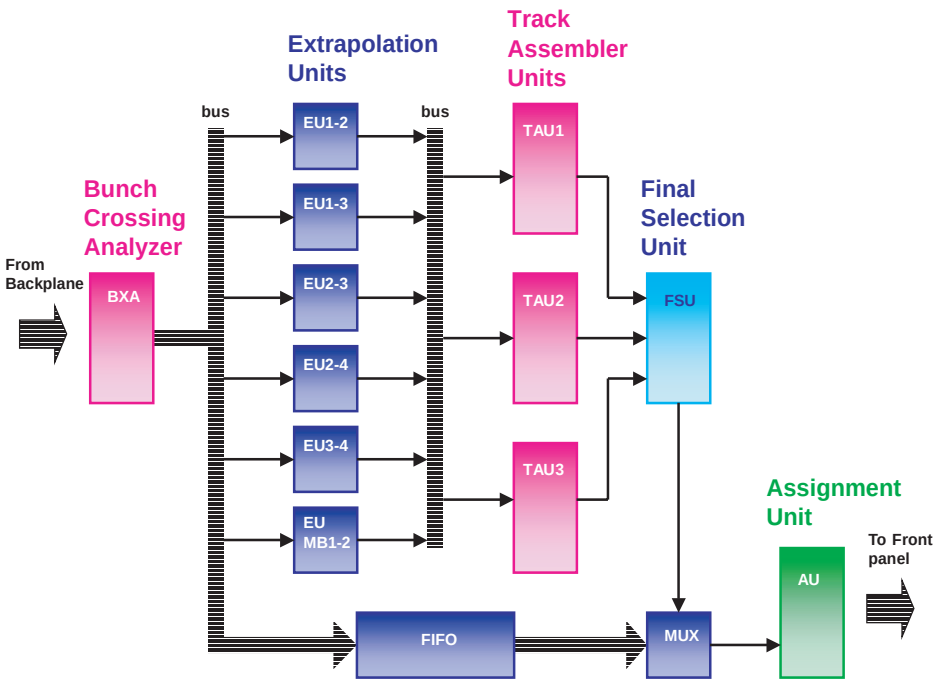


Figure 3.22: Block diagram of the Sector Processor logic (Source: [27])

3.3.4 CSC Track Finder

In the underground counting room, 12 *Sector Processor* boards (see figure 3.22) in the CSC Track Finder crate receive the LCTs for one 60° azimuthal sector each. Data is being synchronized and pattern numbers are converted to η, ϕ and ϕ_b coordinates using lookup tables; at this stage, misalignment can be corrected for by altered lookup values. Up to four track segments are received from the DT station MB2/1 Sector Collector Board. In contrast to the DT track finder, data exchange between neighboring stations in the endcaps is not necessary for the CSC track finder, since bending of muon tracks is limited in the endcaps. Converted coordinates and quality bits are forwarded to the DT track finder and the next stage on the CSC SP board, the *Bunch Crossing Analyzer*, which synchronizes data from DT and CSC. Its logic allows for inclusion of out-of-time bunch crossings into the same extrapolation step, because the DT track finder sends two track segments from one chamber (and one bunch crossing) one after the other (i.e. taking two bx) over the same link; and because bunch crossing assignment in the DT and CSC systems is not fully accurate due to drift times. If capacity suffices in the following stage, track segements/LCTs from more than one bx are included into the extrapolation and flagged in order not to deliver the same track twice.

In the next stage, the *Extrapolation Unit*, almost all pairwise combinations of LCTs (with full 3D coordinates) are compared to each other for common compatibility with a muon track from the interaction point. Combinations of stations 1 and 4 are avoided, since the very high rates in these areas would result in too many coincidental matches. The *Track Assembly Unit* determines whether any of the track segment pairs identified by the previous stage belong to the same muon. For valid tracks, a hit in either ME2 or ME3 is required ('key stations'), because the others have too many background hits and thus track segments. Output is divided into three data streams: patterns with 'key hits' in ME2, ME3 (barrel/endcap overlap region) and ME3 (endcap region). Extrapolation quality codes are assigned to the track candidates (based on how successful extrapolations in a track, which stations with a hit). The *Final Selection Unit* combines the information from the three Track Assembler streams (with up to three tracks each), canceling out duplicate tracks: if the number of common segments (identified by previously assigned segment numbers) exceeds a threshold, the lower quality track is discarded. The best three tracks are delivered to the next stage.

Finally, a p_T value is assigned to the remaining tracks using the differences between the ϕ-values at two or three stations (as available), together with a sign of charge bit based on the same measurement. The η and ϕ coordinates are converted to give their value at the ME2 station in order to allow later comparison with CSC data in the GMT. A three bit quality code is assigned, indicating the accuracy of the p_T measurement. Quality code 3 for instance means a p_T value based on three or four stations including ME1 or MB1, while code 1 describes an assignment based on two stations, including ME1 or MB1; or three stations ME1-ME3-ME4. Quality code 1 indicates a measurement quality so poor (mostly tracks without a segment in ME1) that p_T is set to the maximum value (140 GeV/c) to maintain trigger efficiency (but resulting in higher rates).

One *Muon Sorter Board* receives up to three tracks each from the twelve Sector Processors and determines the best four (based on p_T and quality) to be sent to the GMT.

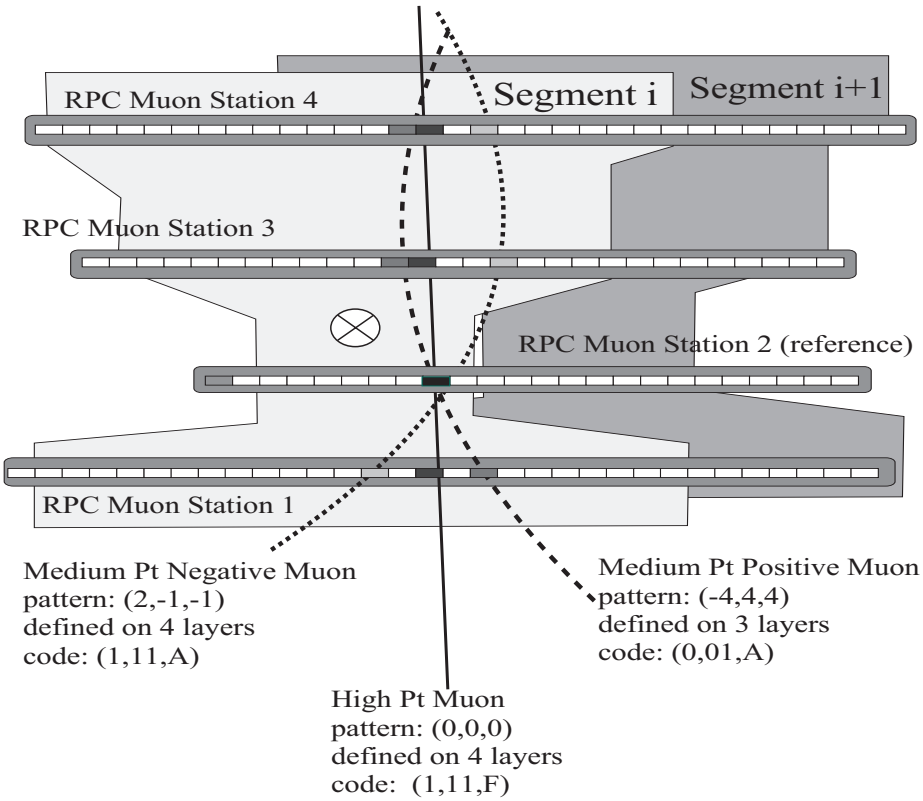


Figure 3.23: Overlapping cones of input strips to the RPC trigger PAC ASICs and principle of pattern-based track reconstruction (Source: [27])

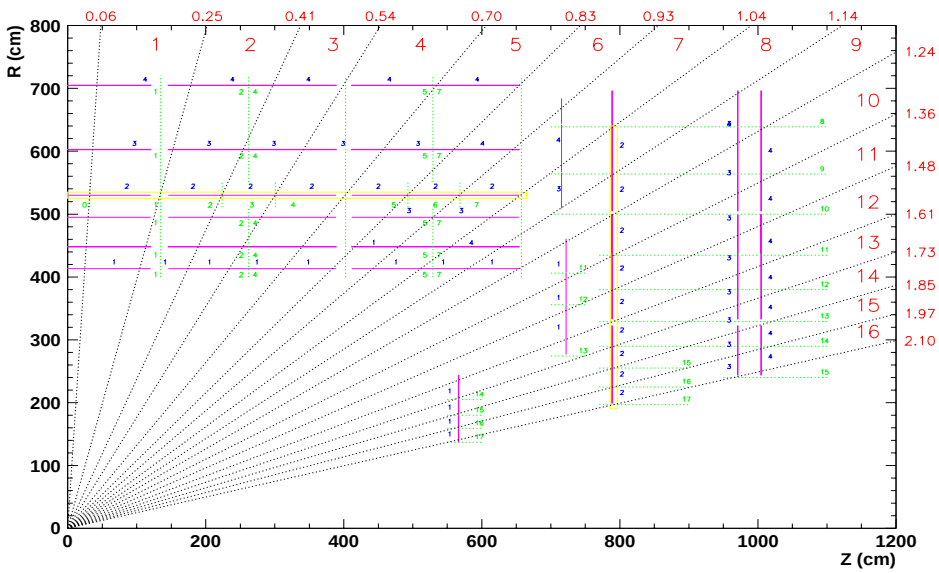


Figure 3.24: RPC trigger tower boundaries in the (r, ϕ) plane (Source: [27])

3.3.5 RPC Trigger

The CMS RPCs being a dedicated trigger detector, the RPC Trigger is responsible for both DAQ readout and triggering on high p_T muons. *Front End Boards (FEB)* process inputs from the RPCs in two eight channel ASICS (amplifier, discriminator, monostable and differential line drivers) and deliver them to the *Link Boards*, which collect data from several FEBs, synchronize them to the TTC clock and perform data compression (the whole RPC system has 200,000 channels but very low occupancy, so compression and multiplexing can save a great deal of bandwidth). Optical links transmit data to the counting room (with the number of links varying according to chamber occupancy), where the signal is received and split, one link going to the *Readout Boards* for delivery to the DAQ, one going to the *Trigger Boards (TB)*. On the TBs, Pattern Comparator ASICs (PAC) receive data from eight RPC strips (called a segment; corresponds to $\Delta\phi = 2.5°$ and $\Delta\eta = 0.1 \ldots 0.2$ in ϕ) in one 'reference station' (the upper one in MB2 in the barrel, ME2 for the endcap) and more strips in three other stations, covering an hourglass-shaped cone that overlaps with the neighboring PACs (see figure 3.23). The PACs compare hit patterns from four stations to pre-defined valid ones, delivering up to one track candidate. Since low-p_T muons (< 6 GeV/c) don't reach the outer stations MB3 and MB4, the two inner DT stations have RPCs on both sides to provide hits from four stations for both low and high-p_T muons. Since the total number of possible hit patterns would be too large, especially for low-p_T muons granularity is reduced by OR-ing strips at the PAC inputs. Thus, about 80 patterns in total remain for comparison. If several track candidates are found, the PACs internally select the best one (based on p_T and quality). Since the overlapping segments give rise to ghosts (tracks detected twice by neighboring PACs), a 'ghost busting' step is performed after the PACs. For each of the 33 trigger towers – logical segmentations in η (see figure 3.24), which don't correspond to the physical segmentations of the RPCs, but are rotated in ϕ by 5° to reduce the number of optical links necessary – up to 144 track candidates are the result of the PAC computation, of which the best four are delivered to a second ghost busting logic. A final sorting stage selects the best four muons in the endcaps and the barrel for delivery to the GMT.

3.4 L1 Global Trigger

All the trigger objects computed in the Calorimeter and Muon Triggers (with its final step, the Global Muon Trigger left to be described in the following chapter) finally are received by the Global Trigger (GT) [62, 61, 37] which synchronizes the different latencies of these trigger paths and evaluates a set of freely programmable trigger conditions. If at least one of these conditions yields 'true', the GT issues a 'Level-1-Accept' (L1A) signal via the TTC optical network to all the components in CMS that are connected to the DAQ (including the L1 trigger itself), and data readout is triggered. The GT also allows for throttling the overall L1A rate (after condition evaluation) based on occupancy/status feedback signals from the trigger & DAQ components via the TTS system. The Global Trigger Crate in the underground counting room also houses the Global Muon Trigger and the Trigger Control System (TCS), which issues control signals to the L1 systems via the TTC network.

Figure 3.25 gives an overview of I/O and internal data flow in the GT crate: The *Pipeline Synchronized Buffer (PSB)* cards receive and synchronize trigger object inputs from the GCT (4 isolated electrons/photons, 4 non-isolated e/γ, 4 central jets, 4 forward jets, 4 τ-flagged jets, total transverse energy E_T, total transverse energy in jets $\mathrm{H_T}$, missing transverse energy

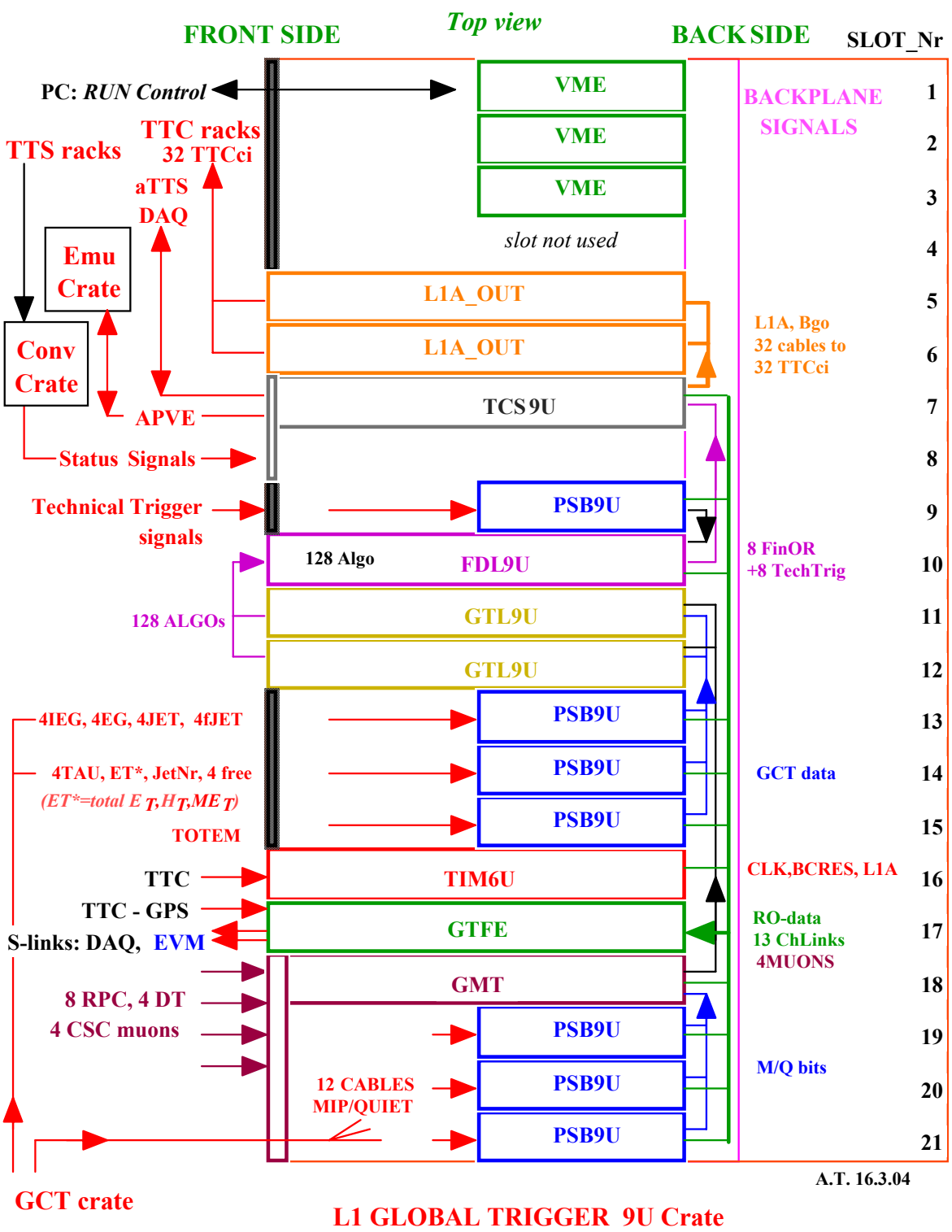


Figure 3.25: Schematic overview of the Global Trigger crates

E_T^{missing} (as a two-vector with magnitude and azimuthal direction) and 12 jet counts (jets above some externally set thresholds). The PSB cards are also used to synchronize the MIP and ISO bits sent from the GCT to the GMT (see next chapter) and technical triggers from TOTEM and CMS for calibration, tests and other purposes. Altogether, the GT crate contains seven PSBs (3 GCT → GT, 3 GCT → GMT, 1 technical triggers → TCS). PSBs can receive data on eight serial receivers (1.2 Gbps Infiniband; converted to 16 bit, 80 MHz parallel) and/or 16 parallel LVDS (64 bit, 40 MHz) channels. Data is oversampled for phase adjustment, then a programmable delay can be applied for synchronization (individually for each 16 bit channel). Ring buffers keep data for later DAQ readout on L1A and simulation memories allow for injection of test data into the system via VME, along with 'spy' memories which allow for input data readout via VME. Finally, data is sent as multiplexed GTLp signals via the GT crate custom backplane to the receiving boards (GMT, GTL or TCS).

The *Global Trigger Logic (GTL)* board is the very heart of the L1 trigger system: all the aforementioned GCT data and the best four muons (as delivered by the GMT via the backplane) serve as inputs to 128 programmable trigger condition expressions. These expressions are computed in two steps: first, the so-called 'conditions' are evaluated: these are tests for some parameters of one, two or four trigger objects from the same family (i.e. e/*gamma*, muons, ...), and subsequently boolean expressions (called 'algorithms') with the conditions as operands are composed to allow for overall trigger conditions containing a logical combination these particle families. A graphical setup program has been developed [36] that provides an intuitive way of entering and viewing setups of such conditions and algorithms; it reads and writes XML files, which are further translated into VHDL firmware source code implementing the logic specified [47]. VHDL code is subsequently synthesized and downloaded into the GTL's two algorithm FPGAs from the crate controller PC via a PCI-VME bus adapter. It is possible to evaluate the following types of conditions (and combinations thereof, see figure 3.26):

Simple particle condition: thresholds on E_T (for calorimeter objects) or p_T (muons), masks on η and ϕ. for muons only: sign of charge, reconstruction quality (muons); for calo objects only: isolation, MIP (minimum ionizing particle)

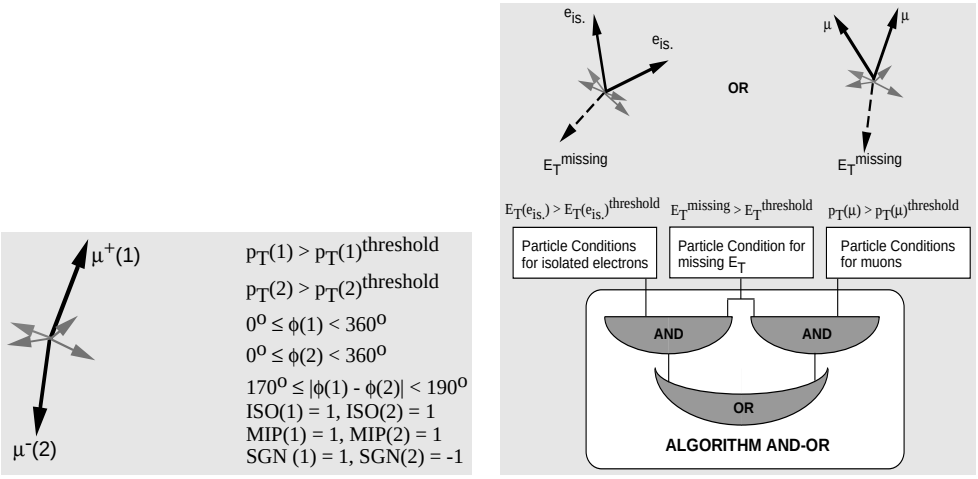
Spatial correlation conditions: masks on differences between the η or ϕ coordinates of two particles, thus allowing for instance to require two muons back-to-back, i.e. with $170° \leq |\phi_{\mu_1} - \phi_{\mu_2}| \leq 190°$

Energy conditions: thresholds on total E_T, H_T, E_T^{missing}, direction of E_T^{missing}

Jet count conditions: numbers of jets in externally defined energy ranges

The (up to) 128 algorithm result bits are sent to the neighboring *Final Decision Logic (FDL)* board, which computes a final OR of these bits and the 64 technical trigger bits (which can also be disabled using veto masks). Trigger partitioning is supported by the FDL: it is possible to assign groups of algorithms to up to eight partitions, which receive different final OR signals. FDL also applies programmable prescaling factors (to enforce a lower trigger rate by randomly dropping L1A decisions) and maintains rate counters for each algorithm. Like other cards, FDL provides readout data to the DAQ and enables test pattern injection and spy readout using SIM/SPY memories.

The eight final OR bits are propagated to the *Trigger Control System (TCS)* board, which controls the L1A rate and applies throttling if necessary. It also issues control signals to all



(a) Two muons back-to-back (b) Boolean combination of conditions

Figure 3.26: Illustrations of trigger conditions (Source: [62])

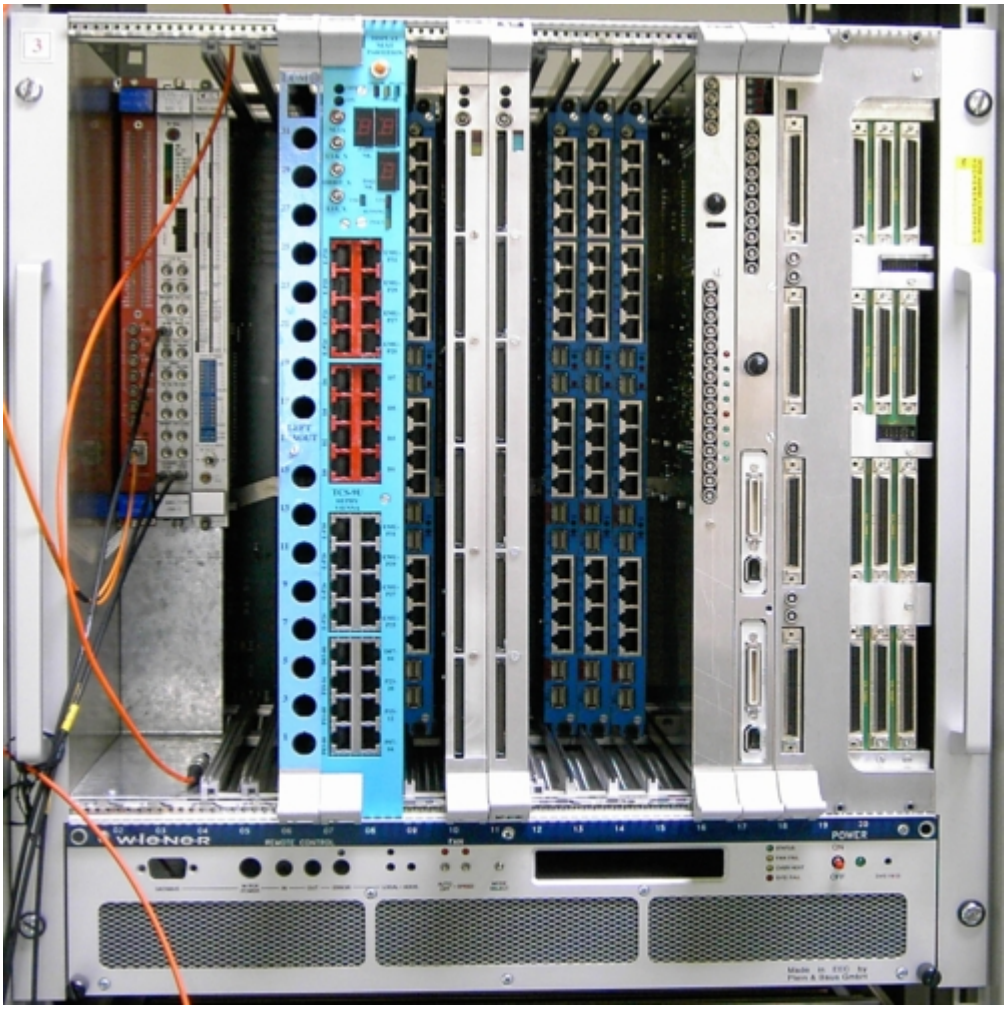


Figure 3.27: The Global Trigger Crate with all prototype modules. From left to right: PCI-VME64x bridge and auxiliary modules (6U, left top). L1A Out – TCS – PSB – FDL – GTL – 3 × PSB – TIM – GTFE – GMT – 3 × PSB hidden behind GMT front panel

the readout and trigger crates. Via its front panel, the TCS receives status information from the Trigger Throttling System (TTS), which was mentioned already earlier in these chapter (synchronous signals from the front end electronics, asynchronous ones from the DAQ indicating buffer occupancy, sync errors, busy/ready, ...). In the case of the inner detectors however, this feedback method is not fast enough due to the great data rates (and therefore smaller buffer sizes in terms of bunch crossings) and too long signal propagation delays. Therefore buffer occupancy of these systems is emulated in local state machines in a neighboring crate, which deliver fast feedback signals to the TCS. Thus, the TCS is able to assure maximum average and instantaneous L1A rates and react in a well defined way to the conditions of the trigger & DAQ subsystem. Apart from the L1A signal, the TCS also issues various control signals to the subsystems, including the bunch counter reset (BCRES), re-sync and reset commands (called 'BGo' commands). Functionality is included to manually issue triggers and control signals to the various partitions during testing and commissioning. TCS sends its output via the crate backplane to the *L1A Out* boards that fan-out the signals to 32 TTCci cards in a neighboring rack, which encode them onto the TTC optical network and distribute them to the 32 detector partitions.

The GMT, FDL, TCS and PSB cards in the GT crate receive the L1A signal themselves, causing them to send readout data (event records) to the DAQ via the *Global Trigger Front End (GTFE)*. GTFE converts GTLp signals on the backplane to S-Link signals as used by the DAQ and Event Manager and formats the readout data prior to transmission.

The *Timing Card (TIM)* receives TTC signals for the GT crate (LHC 40,08 MHz clock, L1A, BCRES, BGo, some of which are generated by the TCS) and distributes them via the backplane after applying programmable delays for each board. TIM can also generate a standalone clock for testing and programmable BGo commands (definable for arbitrary bunch crossings).

Chapter 4

Global Muon Trigger

The Global Muon Trigger board in the Global Trigger crate receives up to 16 muon candidates per bunch crossing from the Regional Muon Triggers and combines them in order to find the best four candidates in the entire detector. It additionally receives 'minimum-ionizing particle' (for additional confirmation) and 'quiet' bits (for determining isolation) from the Global Calorimeter Trigger and assigns them to the muon candidates. The GMT receives some 1100 bits per bunch crossing and processes them without dead-time in 10 FPGA chips. Since it is the scope of this thesis to integrate the GMT into its surrounding software and hardware environments, it is described here in some detail. The following is based on earlier design, implementation, simulation and performance studies [54, 52, 55, 56, 48] carried out by members of the Institute for High Energy Physics of the ÖAW, Vienna.

4.1 Conceptual Design and Functions

4.1.1 Matching and Pairing

Since the CMS muon system is (at least) twofold redundant in both the barrel and the endcaps, the DT and RPC (barrel) or the CSC and RPC (endcap) systems respectively may deliver the same muon candidates, and such a match is of course a good confirmation of the measurement. Therefore it is one of the GMT's tasks to match and compare candidates from the respective subdetector trigger paths. Note that the treatment of the DT-CSC overlap region is described elsewhere (section 4.1.2). In a fist step the spatial proximity for each pair of muon candidates in two corresponding subdetectors is computed: while the ϕ scale is linear and allows for a simple substractor (modulo 144, to account for wrap-around) to calculate $\Delta\phi$, the η scale is not, and a lookup table (LUT) is required. From the resulting $\Delta\phi$ and $\Delta\eta$ values, a weighted distance $\Delta r = \sqrt{w_\eta(\Delta\eta)^2 + w_\phi(\Delta\phi)^2}$ (with the weight parameters w_η and w_ϕ) is computed, and a match quality value MQ is assigned:

$$MQ = \begin{cases} MQ_{max}\,(1 - \frac{\Delta r}{\Delta r_{max}}) & : \Delta r \leq \Delta r_{max} \\ 0 & : \Delta r > \Delta r_{max}, \end{cases}$$

with Δr_{max} denoting the size of the matching window. Due to the non-linearity of the η scales in the DT and RPC triggers, the resolution in η is dependent on η itself. This is taken into account in the $\Delta\eta$ LUT by correcting that difference: instead of the nominal difference between the bin centers, the difference between the centers of simulated muon distributions

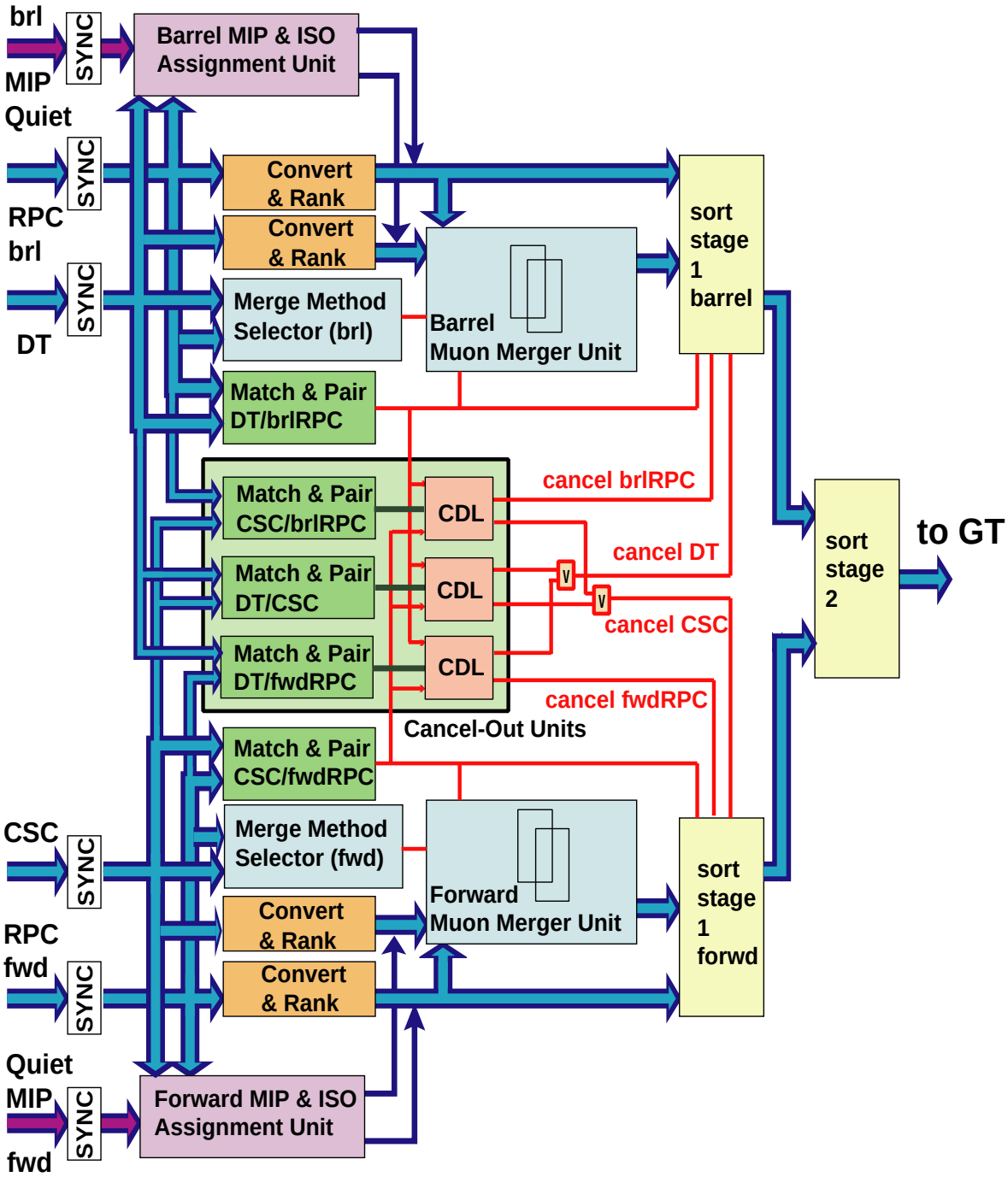


Figure 4.1: Block diagram of the GMT (Source: [54])

assigned to the two bins is used, improving match efficiency while maintaining small match windows; a larger window would give rise to fake matches.

Each element in the resulting 4×4 match quality matrices (one for barrel and endcap each) is then tested for being the highest match quality in its row and column, and the corresponding muons are paired. The paired muons are excluded (corresponding rows and columns disabled), and the step is repeated to find more matches. The result is a pairing matrix with up to four `true`-values in different rows and columns. See figure 4.2(a) for a schematic view of the matching logic.

4.1.2 Ghost-busting in the Overlap Region

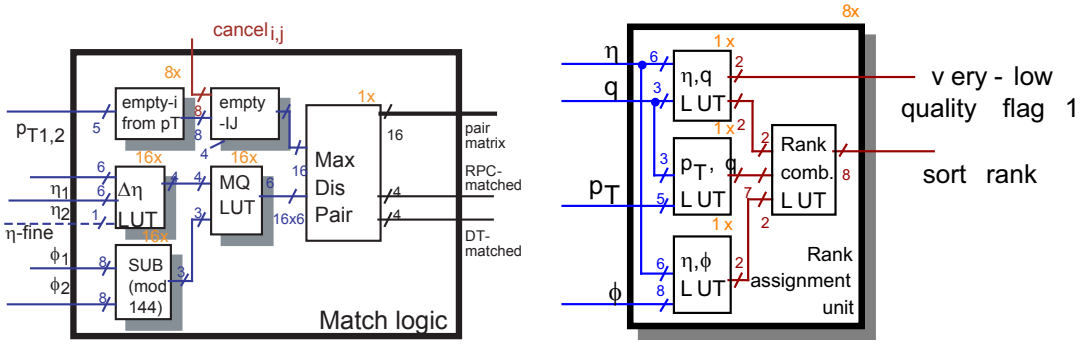
In the barrel-endcap overlap region, ghosts may be created if a muon is reported by both the DT and CSC system; or by one of them and the RPC trigger, but assigned to the other RPC detector region (i.e. CSC and barrel-RPC or DT and forward-RPC); the GMT treats these two cases in separate Cancel-Out Units (COU). Since the barrel and endcap regions generally are treated in two distinct processing paths (i.e. FPGAs, see section 4.2), the η and ϕ values are exchanged between these paths, along with the matching decisions taken within the paths. In the Cancel-Out Units (see figure 4.3), matching is performed in a way analogous to the procedure described above, but in the combinations CSC-DT, fwdRPC-DT and brlRPC-CSC. Based on the pairing values obtained within the Cancel-Out Unit and those computed within the fwd/brl trigger paths themselves, a cancel-out decision is taken, by default according to the following truth tables:

Inputs			Output		
DT match w/ CSC	CSC match w/ RPC	DT match w/ RPC	cancel brlRPC (optional)	cancel DT	Comment
0	x	x	0	0	no DT/CSC match
1	0	0	0	0	cancel CSC
1	0	1	0	0	cancel CSC
1	1	0	0	1	cancel DT
1	1	1	0	0	no cancellation

Table 4.1: Default programming of the Cancel Decision Logic for the barrel part of the GMT (Source: [54])

Inputs			Output		
CSC match w/ DT	DT match w/ RPC	CSC match w/ RPC	cancel fwdRPC (optional)	cancel CSC	Comment
0	x	x	0	0	no DT/CSC match
1	0	0	0	1	cancel CSC
1	0	1	0	0	cancel DT
1	1	0	0	1	cancel CSC
1	1	1	0	0	no cancellation

Table 4.2: Default programming of the Cancel Decision Logic for the endcap part of the GMT (Source: [54])



(a) Match logic (b) Ranking logic

Figure 4.2: Match and Ranking logic units (Source: [54])

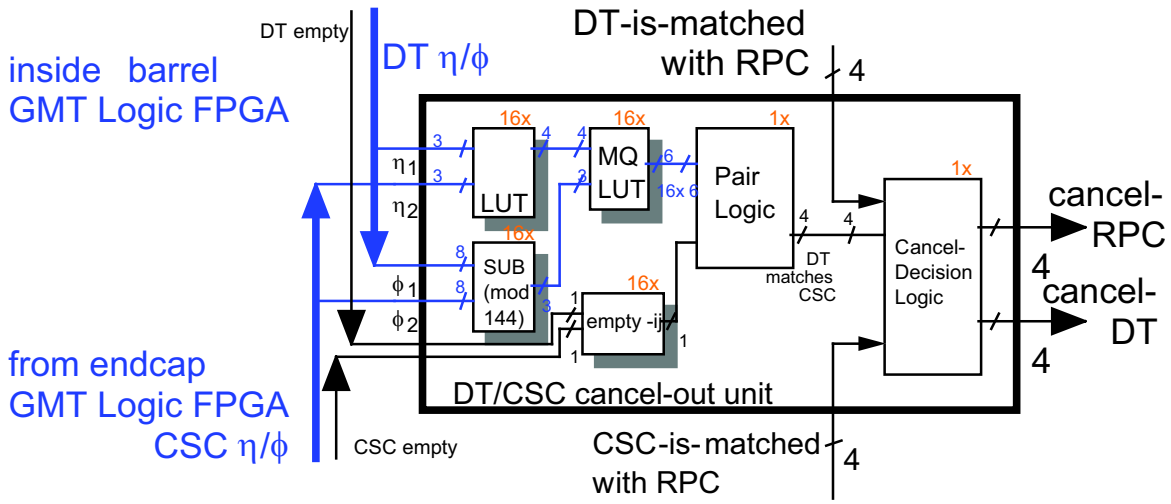


Figure 4.3: Overlap Region DT/CSC Cancel-Out Unit, inside the GMT Barrel Logic FPGA (Source: [54])

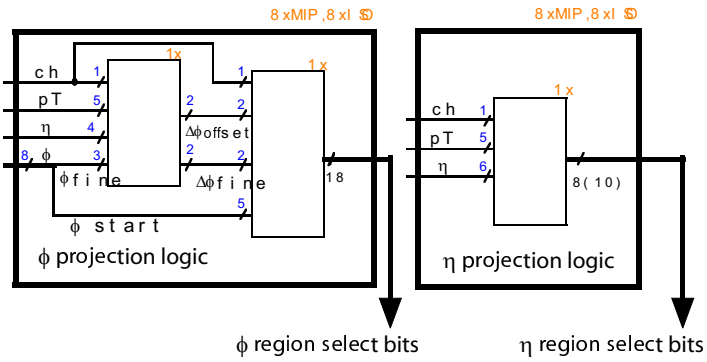


Figure 4.4: Parameter projection logic (Source: [54])

Inputs			Output		
CSC match w/ brlRPC	CSC match w/ fwdRPC	brlRPC match w/ DT	cancel brlRPC	cancel CSC	Comment
0	x	x	0	0	no CSC/brlRPC match
1	0	0	0	1	cancel CSC
1	0	1	0	0	leave cancellation to DT/CSC COU
1	1	x	0	0	no cancellation

Table 4.3: Default programming of the CSC/barrelRPC Cancel Decision Logic (in the barrel processing chain of the GMT) (Source: [54])

Inputs			Output		
DT match w/ fwdRPC	DT match w/ brlRPC	fwdRPC match w/ CSC	cancel fwdRPC	cancel DT	Comment
0	x	x	0	0	no DT/fwdRPC match
1	0	0	1	0	cancel fwdRPC
1	0	1	0	0	leave cancellation to DT/CSC COU
1	1	x	0	0	no cancellation

Table 4.4: Default programming of the DT/forwardRPC Cancel Decision Logic (in the forward processing chain of the GMT) (Source: [54])

While it would be a simple solution for the first category of ghosts – DT-CSC duplicates – to always require a match with RPC this would reduce overall efficiency to RPC efficiency. Therefore, as can be seen from the truth tables, if a candidate is found by both the DT and CSC systems but only once in the RPCs, the non-matching DT/CSC candidate is dropped and the corresponding candidate in the other processing chain is kept. If a DT and a CSC candidate match, and both are confirmed by the RPC, then both are kept in order not to suppress di-muon triggers. The result of the CSC/DT-RPC COUs is OR-ed with the result of the DT/CSC COUs.

4.1.3 MIP/Iso Assignment

As mentioned already, the GCT delivers a MIP and a Quiet bit for every one of the 252 calorimeter regions ($\Delta\eta \times \Delta\phi = 0.35 \times 0.35$rad) to the GMT. The Mip bit denotes an energy deposit compatible with the passage of a minimum-ionizing particle such as a muon and thus can help confirm a muon candidate and its p_T assignment. The Quiet bit is set on a region if an energy below some (low) threshold was deposited in the HCAL. If all the Quiet bits in a rectangular region around a muon crossing the HCAL are set, the Isolation (Iso) bit is set on the muon, indicating that it does not originate from a jet but from the vertex itself, as is the case in some important discovery channels for the CMS physics programm. Iso bit assignment therefore is expected to help reduce background. Since the η and ϕ parameters available to the GMT are valid for the second muon stations, the trajectories have to be projected back to the inner surface of the HCAL to obtain the HCAL regions necessary for MIP assignment. As jets are not (or neglibibly) bent in the B field, the direction of the muon

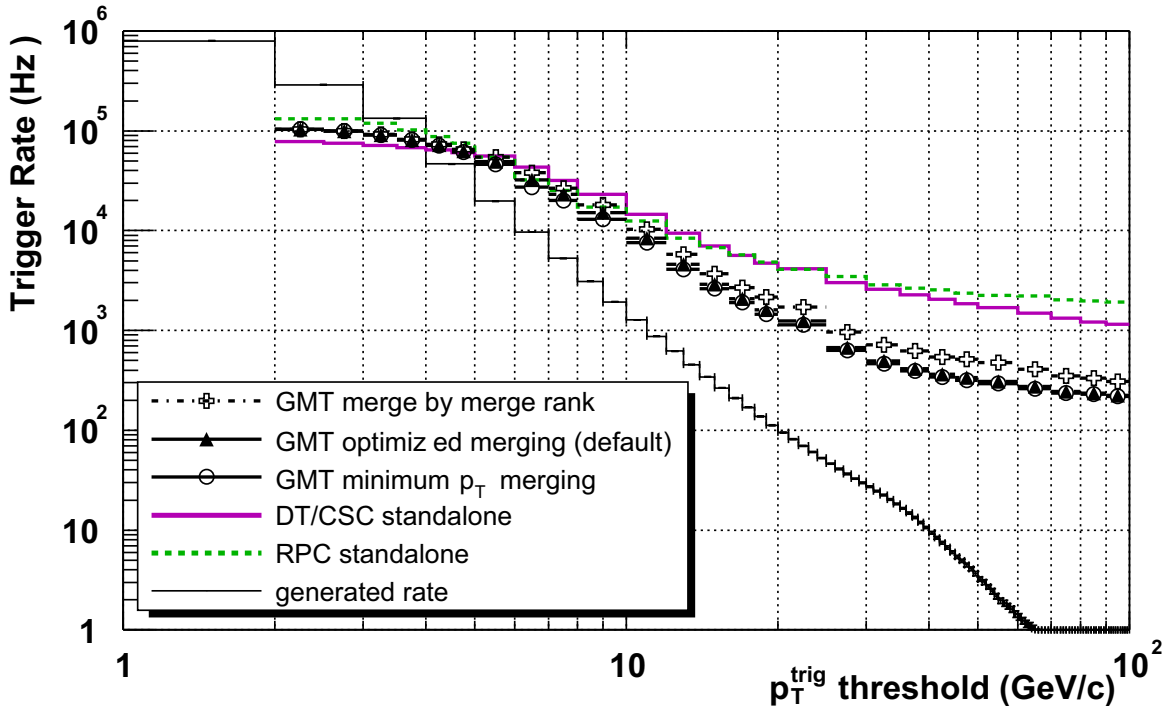


Figure 4.5: Trigger rates at low luminosity ($\mathcal{L} = 2 \times 10^{33}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$) as a function of trigger threshold for three types of GMT merging (see text). Obtained using a minimum bias dataset. Updated ORCA 6.2.3 simulation (Source: [54])

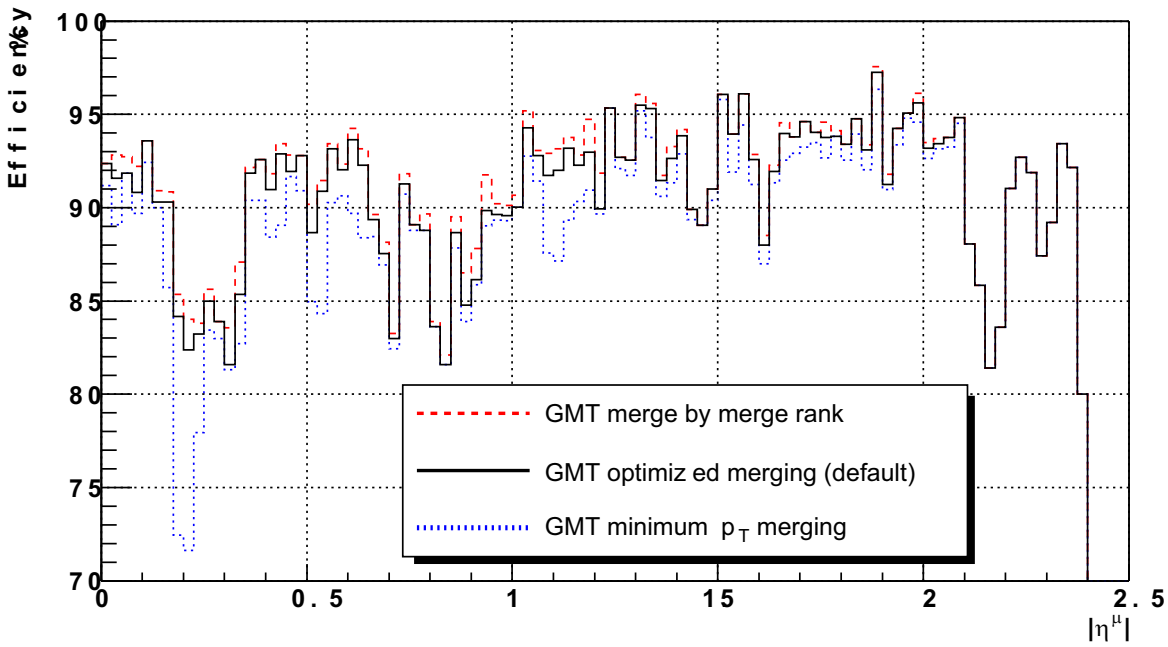


Figure 4.6: Efficiency for triggering on muons from W $\rightarrow \mu\nu$ decays as a function of pseudo-rapidity with a p_T threshold of 14 GeV/c for three types of GMT merging (see text). Updated ORCA 6.2.3 simulation. (Source: [54])

at the vertex has to be known to look for the corresponding Quiet bits. These projections could have been implemented using large LUTs with p_T, sign of charge, η and ϕ as input values, and the calorimeter region select bits as output, but this would have required too much memory. In order to fit all the GMT logic onto one 9U motherboard, projection of η and ϕ was separated, and characteristics of charged track bending in CMS are exploited: the η projection is independent of ϕ, thus a smaller LUT suffices. ϕ projection is done in two steps: a coarse step which simply omits the three lower bits of the ϕ value to get the index of the calorimeter region, and a fine step that uses p_T, sign of charge, the lower ϕ bits and η to obtain an offset from that start region (see figure 4.4).

4.1.4 Conversion of Parameters

Since the GMT input and output scales are programmable and may be different, the η coordinate is converted using LUTs. The ϕ variable at the GMT input is defined at the second muon stations while the GT may require its value at the vertex. Therefore ϕ can optionally be projected back to the vertex using pre-computed LUTs, with p_T, η and ϕ as input values.

4.1.5 Sort Rank Assignment

In order to determine the best four muons in the detector, a rank is assigned to every candidate based on the following considerations:

Transverse momentum: Higher-p_T muons are the basis of most trigger conditions, so a high efficiency has to be reached when selecting them. p_T therefore is valued quite strongly in rank assignment.

Too high rates: If certain regions in η (and ϕ) give higher rates than expected, it has to be possible to reduce their ranks. If a region contains hot channels (hardware failures that always or extremely often lead to candidates), it has to be disabled all together.

Triggers from noise: In some pseudorapidity regions, triggers might be generated from noise in the chambers. These candidates mostly have very low quality, but can have any p_T. Thus it is necessary to be able to rank candidates lower based on their pseudorapidity and quality.

Low quality candidates in certain regions: Simulations have shown that certain regions in η can deliver high rates of low quality candidates, partly also due to design problems in the Regional Muon Triggers. In some regions p_T resolution can be very poor, resulting in higher rates. Therefore it is necessary to be able to exclude (or rank lower) these candidates if they are unconfirmed by the complementary muon trigger path.

Exclusion from trigger conditions: Some of the candidates mentioned above may be too likely to be ghosts or fake triggers or correspond to a very poor p_T measurement, therefore they should be excluded from single- and/or di-muon triggers (but maybe included in quad-muon ones). Therefore a quality code is assigned to every muon and delivered to the GT, indicating these conditions.

The rank (and final quality) therefore is dependent on p_T, η, ϕ and regional trigger quality. Again, a straightforward LUT in external ROM memory with all these parameters as input values would be too large (address space about 4 Mbit) to fit onto a single-board GMT design,

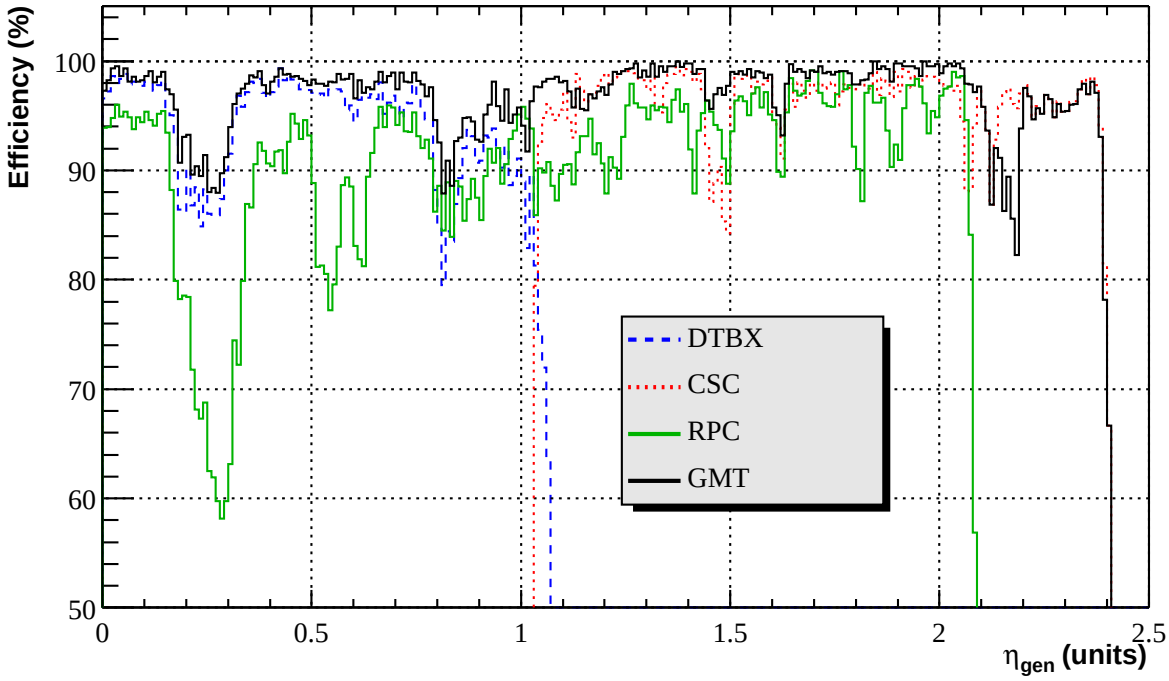


Figure 4.7: Efficiency of the GMT and regional muon triggers as a function of pseudorapidity η for a sample of single muons with flat p_T distribution. No p_T threshold applied. GMT candidates that are suppressed in the single-muon trigger are excluded. Updated ORCA 6.2.3 simulation. (Source: [54])

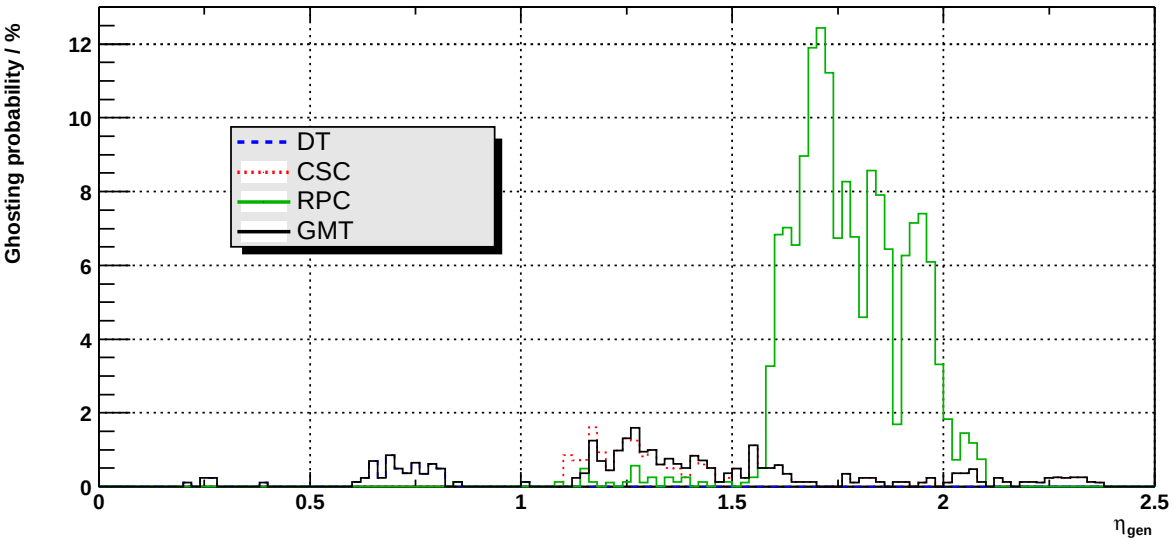


Figure 4.8: Ghosting probability as a function of η in the GMT and regional muon triggers. No p_T threshold is applied. Candidates that are suppressed in a di-muon trigger are excluded. Updated ORCA 6.2.3 simulation (Source: [54])

which is favorable in terms of latency reduction. Therefore the design has been optimized (see figure 4.2(b)), sacrificing some (practically unimportant) flexibility to more modest hardware requirements.

4.1.6 Merging

If the matching and pairing logic described earlier decides to merge candidates, the parameters of the merged candidate have to be determined in an optimized way. The GMT implements a number of different merging methods for the various parameters, which can be chosen using register settings. Extensive simulation studies have been carried out in order to optimize these methods (see figures 4.5 and 4.6 for the efficiency and rate reduction obtainable by different GMT merging methods):

Always DT/CSC/RPC: the simplest method is of course to always give preference to the value delivered by one subdetector trigger system.

Min p_T: since some regions tend to yield low quality p_T assignments, it is preferable to select the lower p_T value available in order to reduce the trigger rate while maintaining efficiency.

Merge rank: Quite similarly to the sort rank assignment logic, it is possible to provide a three bit merge rank for every candidate in a LUT dependent on η, ϕ, p_T and quality. The predefined values are based on the accuracy of p_T measurement in some regions of the detector (and trigger).

Combined p_T and rank: This method is the default for most parameters. On the basis of the values of η and quality, a logic block decides whether to use the min-p_T or merge rank method. Generally, the min-p_T is preferred, apart from cases where efficiency loss would be too large. This way, low rate and high efficiency can be combined to the best possible compromise.

Smart η: this method takes the DT Track Finder's coarse/fine bit into account and prefers the RPC p_T value over the DT one in case of a p_T measurement denoted as coarse.

Smart MIP/Iso: instead of assigning the MIP/Iso bit according to the selection of some other merging method, the GMT can AND or OR these bits to produce the merged value. Defaults are OR for MIP and AND for Iso.

4.1.7 Sorting

In a final step, the GMT sorts all the candidates based on their sort ranks. Sorting is performed in two steps: first for the barrel and forward regions separately, canceling out at this stage the candidates selected by the Cancel Out Units. The best four barrel and forward candidates (respectively) are then sorted again in the second stage, yielding the GMT output muons to the GT.

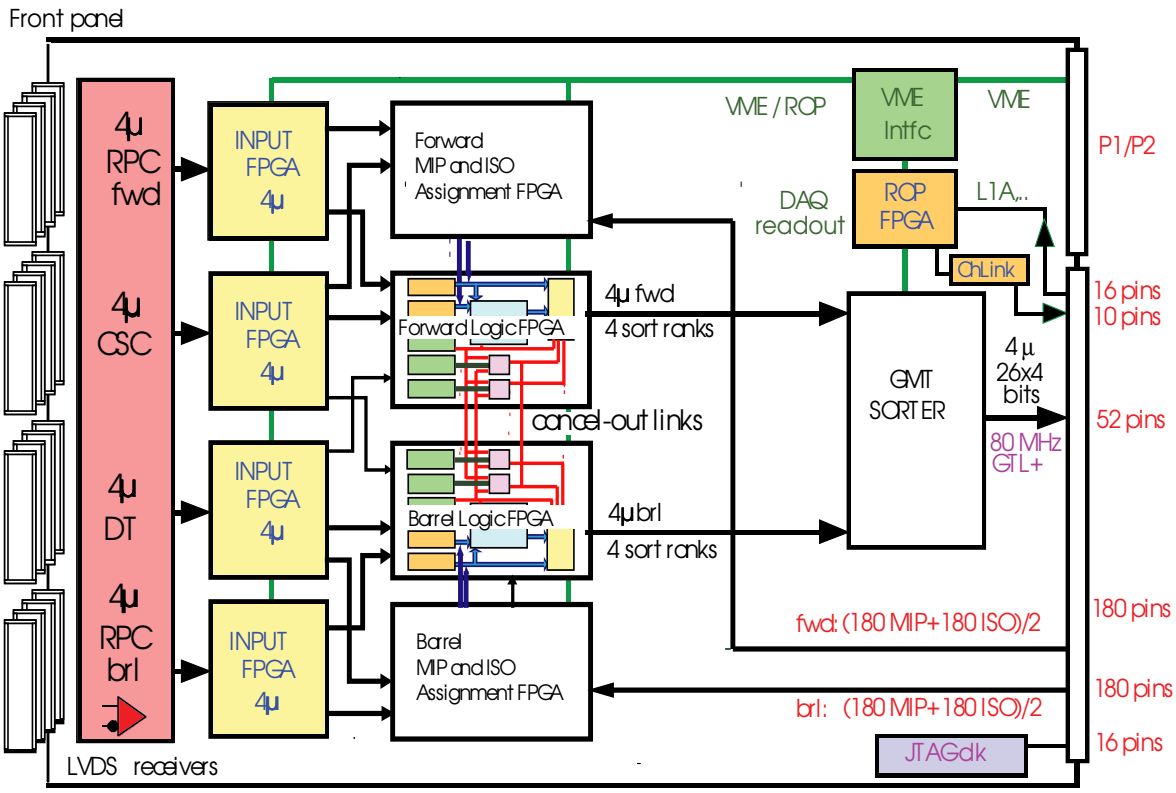


Figure 4.9: Schematic view of the GMT board (Source: [54])

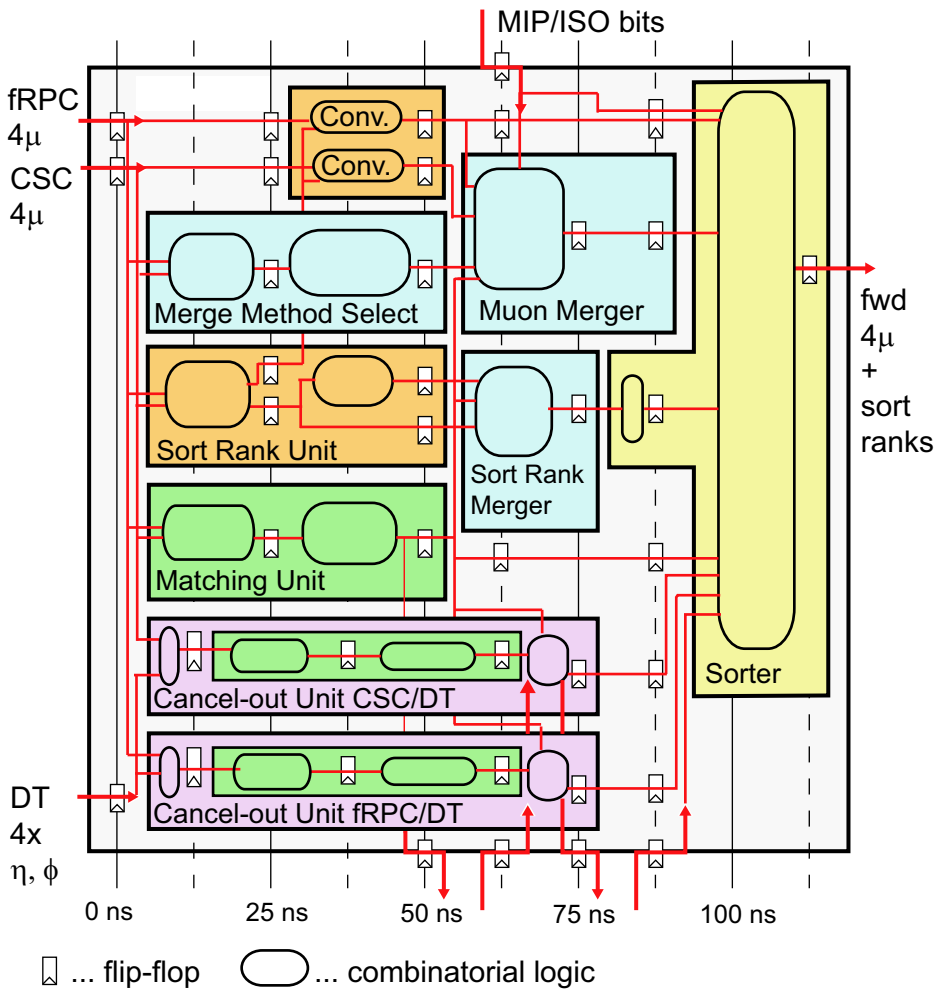


Figure 4.10: Simplified schematic view of the Forward Logic FPGA (LFF) (Source: [54])

4.2 Implementation

4.2.1 Hardware Overview

After some optimization steps mentioned earlier in this chapter and improvements in available FPGA technology, it has become possible for the GMT to be implemented on a single 9U motherboard which is housed in the GT crate. This single-board design reduces the necessary latency and allows for barrel-endcap cancel out logic, since in a two-board design with one board for the barrel, one for the endcap not enough space would have been available on the crate backplane to route the necessary connections.

The muon candidates from the Regional Muon Triggers are transmitted on 40 MHz LVDS parallel connections using Shielded Twisted Pair cables and SCSI connectors on the custom, four slot wide front panel. Each muon is transferred on its own cable, thus 4×4 connectors have been foreseen. After being converted from LVDS to low voltage TTL standard, the signals are received by four Input FPGAs (named INB for barrel RPC, INC for CSC, IND for DT, INF for forward RPC), which over-sample it at 160 MHz to provide for phase adjustment. Next, programmable delays of up to 16 LHC bunch crossings can be employed in order to synchronize the three different muon trigger paths. Each Input FPGA provides four $1k \times 16$ bit VME-addressable RAMs, which can be used for simulation data injection and readout of the received data. Next, the input muons are dispatched using point-to-point links to the two Logic FPGAs, one for the barrel and one for the endcap part (named LFF and LFB for forward and barrel), as well as the two MIP-and-Iso Assignment FPGAs (AUF and AUB for forward and barrel.

AUF and AUB also receive the 2×252 MIP and Quiet bits from the crate backplane connectors, with each chip only seeing those calorimeter regions that it needs for barrel and endcap muons respectively (which overlap). The MIP and Quiet bits are at first received via ChannelLink serial 1.6 Gb/s connections from the GCT by three adjacent PSB cards that are covered by the extra-wide GMT front panel (see figure 3.27). The PSBs phase-adjust and synchronize these bits and send them on 80 MHz time-multiplexed point-to-point links via the backplane (MIP and Quiet alternating). The AUF and AUB chips contain all the logic and (large) LUTs needed for MIP and Iso assignment as described in section 4.1.3 and deliver one MIP and one Iso bit per candidate to the respective Logic FPGAs. Using the default firmware, simulated result bits can be injected before the AUx output stage using small simulation memories.

These LFF and LFB chips contain most of the logic described in the previous section. Input candidates for the respective detector parts are received from the INx chips, and MIP and Iso bits from the AUF/B chips. Additionally, part of the DT data (ϕ and reduced η) is sent to LFF, and conversely part of the CSC data to LFB , which is needed for the cancel-out logic in the overlap region. The two Logic chips exchange the resulting cancel-out decisions as described in section 4.1.2. Each Logic FPGA contains a Matching Unit, eight Rank Assignment Units, eight Parameter Conversion Units, two Overlap Region Cancel-Out Units, four Muon Merger Units and one Sorter. They deliver the best four candidates for their respective detector part to the Final Sorter FPGA (SRT). SRT then performs the final sorting. It also contains large simulation/spy memories for data injection and extraction and multiplexes the four resulting muons for transfer to the GTL board via 80 MHz point-to-point GTLp backplane links.

One last large FPGA, the Readout Processor (ROP) also receives these output muons

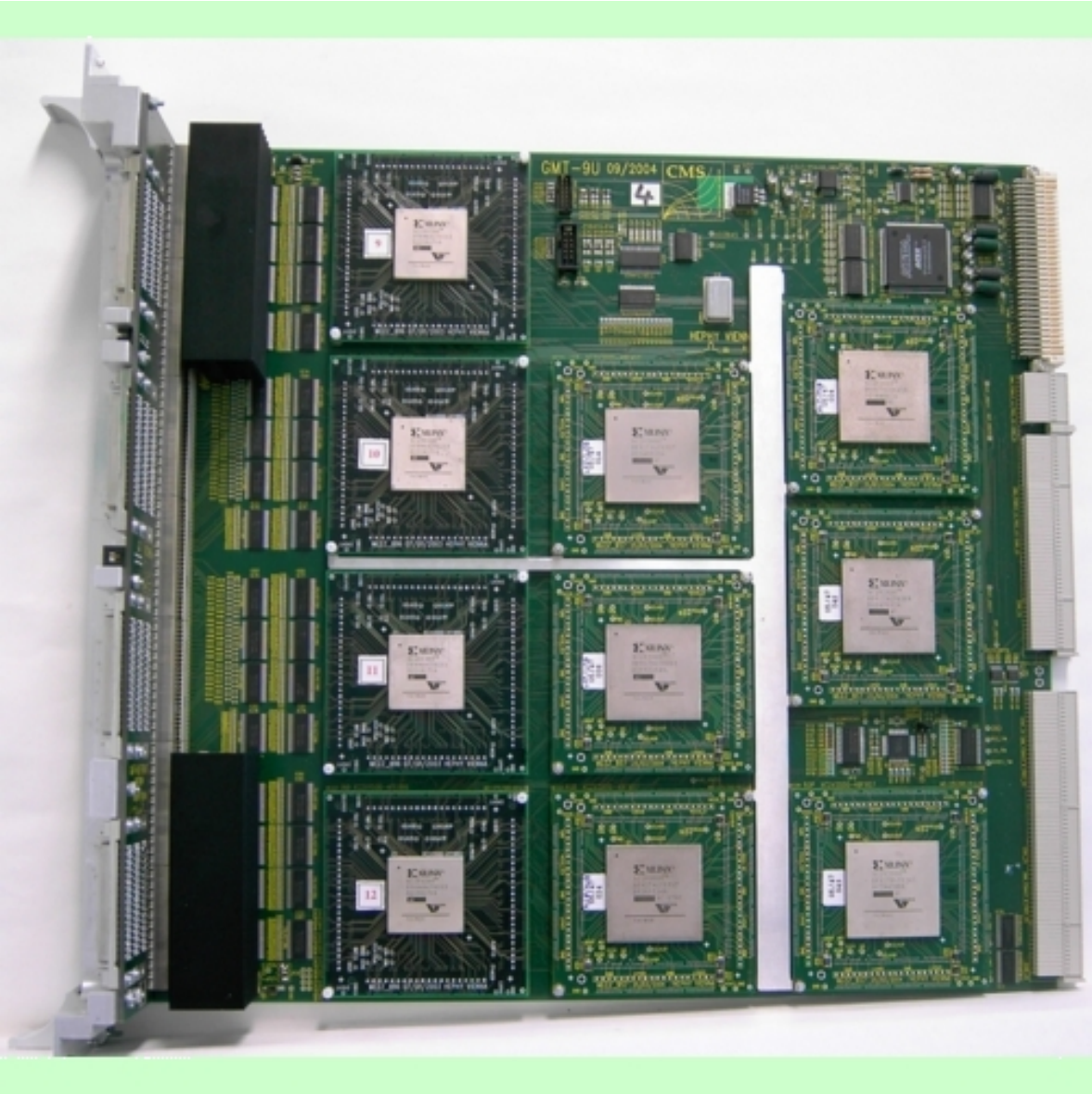


Figure 4.11: The Global Muon Trigger

and stores them in four 1k deep ringbuffers for delivery to the GTFE board via backplane ChannelLink serial connections on L1A. ROP contains the logic necessary for this readout, for VME64x read/write access to all the registers, LUTs and memories on the board, various control switches (reset, ...) and the possibility to program the other FPGAs with new firmware over VME. Firmware is normally stored permanently in two to four EEPROMs per FPGA and loaded by built-in logic at power-up; however the FPGAs can also be temporarily reprogrammed via the ROP chip. The EEPROMs are programmed using the JTAG (Joint Test Action Group) standard, a test protocol and serial daisy-chain bus. JTAG commands are inserted either using a PC (USB or parallel)-JTAG cable to a connector on the GMT front panel or via a VME-JTAG bridge implemented by the ROP chip and the JAL (JTAG access library) C++ library [53]. JAL sequences input files in the SVF format to registers in the ROP chip, which issues the corresponding JTAG commands. The individual EEPROMs and FPGAs can be in- and excluded from the JTAG chain using soldered jumpers. To save space and allow for easy replacement, the FPGAs are located on custom, removable mezzanine daughter boards and the EEPROMS right underneath the mezzanines. The GMT uses two different models of FPGAs of the Virtex-II series from manufacturer Xilinx: the smaller XC2V2000 model for the Input FPGAs, and XC2V3000 for all the others. XC2V3000 provides a total of 484 I/O pins in the packaging variant in use, and 96 18kbit SelectRAM blocks. These resources are mostly exploited by the GMT firmware.

4.2.2 Firmware Development

Prior to firmware development, the functionality of the GMT was implemented in C++ within the CMS ORCA [5] framework (Object oriented Reconstruction and Analysis) and later ported to the CMSSW (CMS Software) [1] framework that replaced ORCA. Extensive simulation studies were carried out to proof the concept and optimize efficiency and rates by improved algorithms. Event samples produced using Monte Carlo algorithms were used, simulating detector response, digitization and algorithmic treatment in the entire chain from detector front-ends, the muon and calorimeter triggers to the GMT. Two simulation result diagrams are shown in figures 4.7 and 4.8, showing the effects of the GMT on muon trigger efficiency and ghosting probability.

Firmware was modeled using the VHDL hardware description language and simulated for functionality verification using the NCSIM VHDL simulator by Cadence. The VHDL code is synthesized using the Synplify program by Synplicity and implemented with the ISE tools package provided by FPGA manufacturer Xilinx. Timing analysis at the gate level is again carried out using NCSIM. The LUTs in the GMT are generated using the same C++ code used for simulation, using scripts to convert the C++ output to VHDL code and generate the necessary Intellectual Property Cores to implement the LUTs in block or distributed RAM – block RAM denoting dedicated memory blocks in the FPGAs, while distributed RAM is facilitated using the generically programmable gates that can be used for any logic.

Chapter 5

GMT Hardware Tests and Integration

In the course of this thesis, extensive self- and interconnection tests of the GMT board and its neighbors in the trigger chain were conducted in Vienna and at the Building 904 Trigger Integration Center at CERN, Geneva. The tasks carried out involved hardware debugging, low-level online software improvement and VHDL firmware development. Several hardware bugs were identified and resolved, synchronization and data transfers with the TIM, PSB, DTTF Barrel Sorter and CSC Muon Sorter boards were verified.

5.1 Software Libraries and PC–Trigger Board Communication

5.1.1 Overview

In order to control, test, monitor and configure the GMT board, a C++ online software library is available that eventually (like online software for other trigger boards as well) uses the Hardware Access Library (HAL) [58] to carry out VME read/write operations on the GMT registers, LUTs and memories. HAL in turn interfaces the device drivers for the PCI-VME64x bus adapters under the 'Scientific Linux CERN' GNU/Linux operating system. This software is installed on a networked control PC in the rack housing the GT crate. The CAEN V2718 and (now obsolete) SBS 620 PCI-VME bus adapter models were used during the tests, which consist of a PCI card in the rack PC, an optical link (or copper cables in case of the SBS) and a 6U module in the crate that drives the VME bus on the backplane. The GMT board contains a VME interface chip that implements VME logic and forwards the lower bits of the VME address lines to the FPGAs, which thus do not need to be aware of the GMT base address in the crate but only their offsets, at the same time saving address lines. Each FPGA contains an own VME decoder logic block and responds to the read/write accesses received.

5.1.2 GMT Online Software

Figure 5.1 illustrates the class structure of the GMT online software library: the low-level classes `GMTChip` and `GMTBoard` map human-readable names to accessible areas (registers, LUTs, memories) in the GMT VME address space and implement generic r/w operations as well as specialized ones for the read-only `fw_id`, `fw_date` and `chip_id` registers present

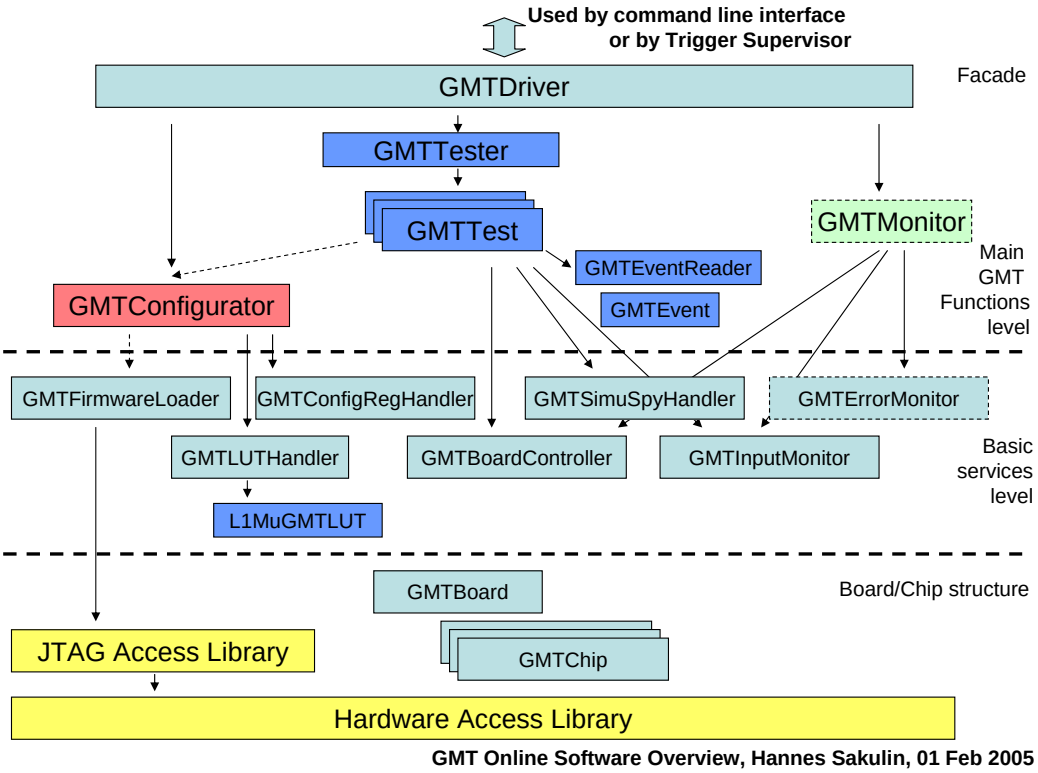


Figure 5.1: Overview of the GMT online software (courtesy H. Sakulin)

in all FPGAs. At the 'basic services level', more complex operations are implemented, such as the parsing and writing/verifying of LUT and register configuration text files generated by the ORCA/CMSSW emulator package for the GMT (classes `GMTLUTHandler` and `GMTConfigRegHandler`. The `GMTFirmwareLoader` uses the JTAG Access Library JAL to parse and sequence Serial Vector Format (SVF) files containing the firmware into EEPROMs on the GMT board; verifying already present firmware is possible as well. These configuration functions are collected in the higher-level classes `GMTConfigurator` (which reads an entire file structure of register and LUT configuration files and downloads/verifies them). The `GMTInputMonitor` class provides convenient access to delay settings of the Input FPGAs and allows to tune the phase adjustment circuits by examining the number of hi-lo transitions between clock cycles of the 160 MHz oversampling logic and selecting the cycle to use as the input sample. Simulation/spy memories can be read out and filled using the `GMTSimuSpyHandler` class. `GMTBoardController` allows to send command pulses to reset the volatile memories in the chips, reload the firmware from the EEPROMs (`NRPOG`) and reset the Digital Clock Manager (DCM) blocks of the FPGAs to attempt re-locking on the clock signal. Other auxiliary classes allow for the conversion of binary values from the hardware to human-readable form (e.g. muon candidates) or parsing of simulation data files generated by the emulator software. The `GMTTest` abstract class defines an interface for test routines, and several implementations exist (e.g. `GMTRegisterTest` for register r/w). Other implementations are described in the following sections in the context of their applications in test scenarios.

There is a simple command-line interface (CLI) program available to access the functions of the online software library. The GMT cell of the Trigger Supervisor framework also acts as a client to the library and is described in section 6.2.

The GMT online software source code can be found in the CERN CVS version control system, a web interface [4] is available. A class reference manual can be accessed at [3]. Several extensions and improvements were made to this library which are described in the following section.

5.2 Software/Firmware Improvements and Extensions

5.2.1 MIP/Quiet Assignment FPGA Test Firmware

Using the operational firmware of the MIP/Iso Assignment Unit chips (AUB, AUF) it is impossible to read out all the 2×252 MIP and Quiet bits arriving from the PSB cards, since there is not enough space available on the FPGA for a spy memory large enough to capture them. The default firmware merely allows for a readout of the 2×8 output bits of the AUx chips, which is not enough to verify all the connections. Therefore a testing firmware was developed that is stripped of most of the trigger logic in the chips and instead implements a large spy memory (wide and deep enough to capture the MIP/Q bits of 1k events) that can be read out via VME. The firmware also can be used to inject the AUx output bits into the data flow, just like the default programming. The testing firmware was successfully synthesized and used in several transfer tests (see next sections).

5.2.2 MIP/Quiet Bit Tests

Several methods for testing MIP/Quiet bit transfer from the PSB cards and processing of the bits in the GMT MIP/Iso Assignment FPGAs (AUB, AUF) were implemented in the course

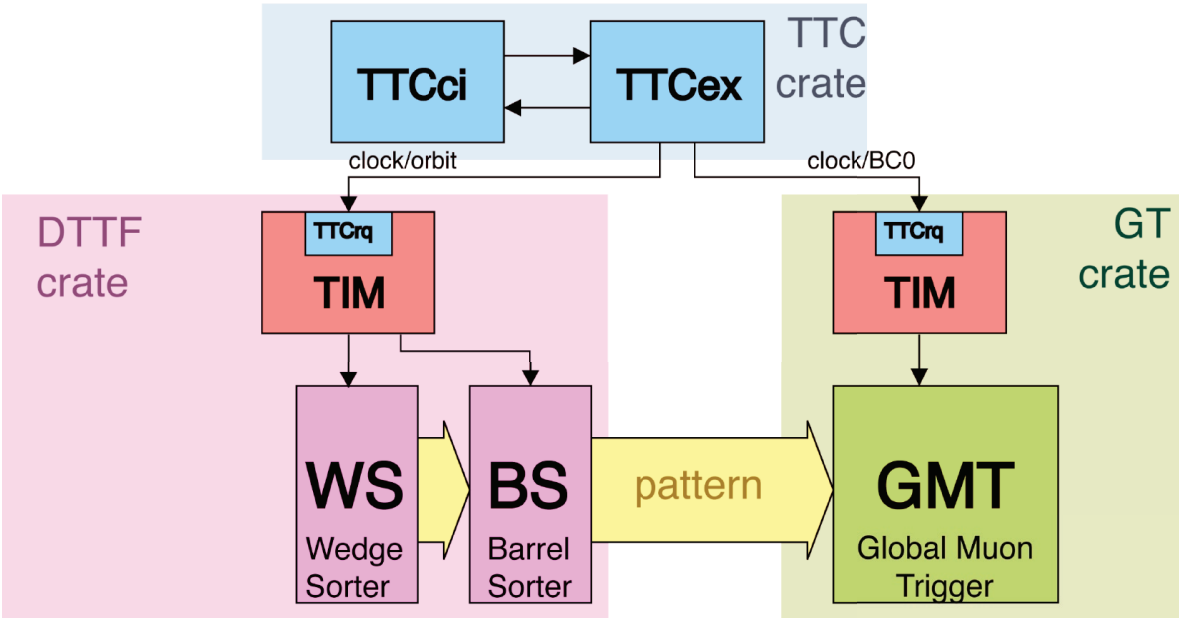


Figure 5.2: Schematic setup for the GMT-DTTF interconnection test (courtesy I. Mikulec)

of this thesis. Auto-generated class documentation for the online software can be found at [4]. The C++ classes developed for this purpose are described below:

GmtPsbOrcaMuonSelfTest

This class parses a test file format generated by the ORCA/CMSSW GMT emulator package which contains both the GMT input and output data as well as intermediate values in human-readable form. The muon candidates to be used in the test are downloaded into the Input Chip simulation memories on the GMT, and the MIP/Iso bits are downloaded into the PSB simulation memories after being converted to the necessary format using the helper class `GmtMipQHandler` (see below). Both GMT and the PSBs present in the crate are configured for the testing procedure (delays, other register settings), and the test data transfer from the PSBs is initiated in endless loop mode. Transfer from the simulation memories in the INx chips is started, and the result is read out from the SRT chip spy memory as soon as processing of all the events is completed. The results from the hardware are compared with the simulated results from the input test file, and errors are logged. Note that the cards involved of course have to be synchronized to a common clock in order for the transfer to work.

GmtPsbBiterrorTest, GmtMIAUReadoutTest

In order to spot dead/hot/short-circuited bits and other error sources between the PSBs and the GMT AUx chips, a custom testing firmware was developed for the AUx FPGAs (see 5.2.1). The `GmtPsbBiterrorTest` class is used to generate bit patterns (like running 0/1, periodic switching with unique periods for each bit, etc.) and download them into the PSB memories. Readout is triggered and performed by the `GmtMIAUReadoutTest` class, which simply prints the contents of the simulation RAMs on the screen.

GmtPsbInterconnectionTest

This class was developed for use by the GMT Cell in the Trigger Supervisor (see 6.2). In contrast to the classes mentioned before it is not intended for manual debugging but rather for automatic and routine tests of hardware functionality. It generates a walking bit pattern, downloads it into the PSB simulation memories, reads out the transfer results from the MIAU testing firmware and compares it to the expected result. The result of this evaluation is presented in a nice form to the client program.

Helper Classes

Since the mapping of MIP/Quiet bit locations in the PSB memories (which are dictated by GCT trigger tower divisions) to the locations in the AUx memories is non-obvious (as well as the association with η/ϕ regions), helper classes were implemented to facilitate this mapping and correct downloading. The `struct MipQInfo` is designed to represent the full information on one MIP/Q bit. Class `GmtMipQHandler` initializes an array of this `struct` with hard-coded values and provides a number of methods that convert the different representations into each other and return mapping information. The class also allows writing and reading of the various bitsets to/from the hardware.

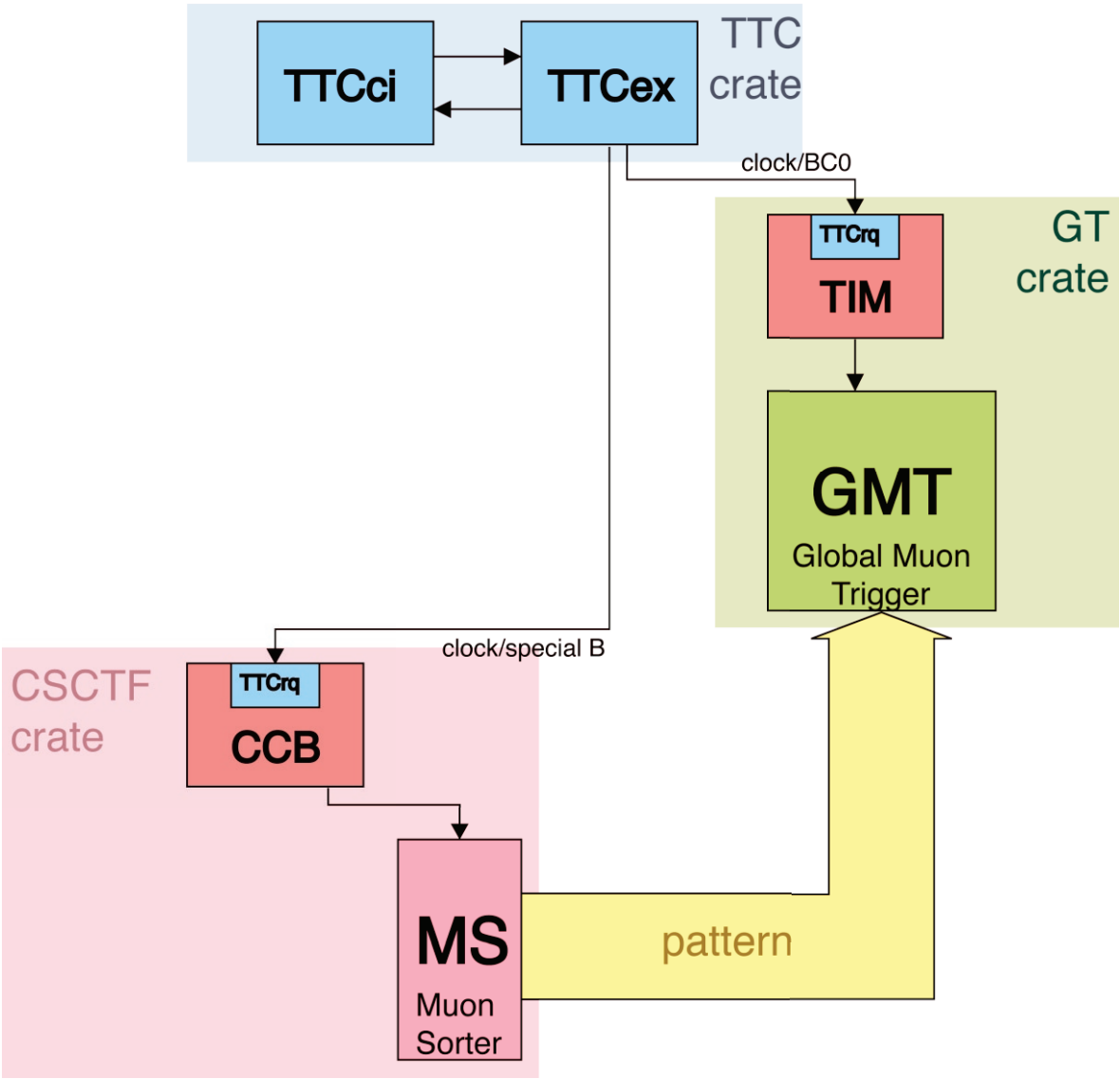


Figure 5.3: Schematic setup for the GMT-CSC interconnection test (courtesy I. Mikulec)

5.2.3 GMT Interface Bit Error Test Class

In order to detect transmission errors appearing very rarely, the GMT hardware has a built-in long term test logic: data is sent in an infinite loop from one of the Regional Muon Triggers; the first loop is captured into the Input Chip spy memories, and for all subsequent loops, that captured data is compared to the incoming data, incrementing counters on comparison errors. The `GmtInterfaceBitErrorTest` class sets up the GMT for this test procedure and allows to read out the error counters on user input. It also allows to print the contents of the simulation memories.

5.2.4 Block Transfer Test Class

VME block transfer allows for faster reading/writing of continuous blocks of data. Instead of alternating address/data cycles, a block transfer consists of only one address cycle followed by up to 256 data cycles. The `GmtBlockTransferTest` class writes 16kB of test data into one of the Input chip memories and gives an estimate of the time needed.

5.2.5 Standalone Emulator Tool

As mentioned already, GMT functionality is also implemented in C++ within the ORCA (CMSSW) frameworks (ORCA in the meantime being obsolete). This emulator software is intended for full-scale simulations of trigger performance starting from raw event data or digitized detector output data at the front-end stage. However, a full simulation is very time consuming; therefore the GMT emulation code was extracted out of the framework and adapted to run in stand-alone mode. This allows for very fast verification of the output generated by the GMT from some input received, since emulation can be performed starting from GMT input data and not detector raw data. Furthermore the standalone variant is capable of computing the GMT LUTs and configuration register values based on emulator configuration parameters on the fly during re-configuration operations.

5.3 Tests Performed

5.3.1 GMT Internal Tests

Extensive tests were carried out using the online software classes described above to verify internal functioning of the GMT. In particular, ORCA/CMSSW generated event samples were loaded into the input chips and MIP/Iso Assignment chips, passed through the GMT and read out from the spy memories of the final SRT chip. By analyzing the differences between expected and obtained result, some bit errors were identified that originated from minor hardware production errors. Two could be resolved with firmware changes, re-routing the signals affected via spare lines. For others, pins had to be re-soldered or mezzanine boards exchanged and cleaned.

5.3.2 Interconnection Tests with PSBs

Two kinds of tests were performed with the PSB cards receiving the MIP/Quiet bits from the GCT and the GMT: in the first setup, MIP/Q bits were loaded into the PSB simulation memories and simulated muon candidates into the GMT Input chip memories. Transfer was

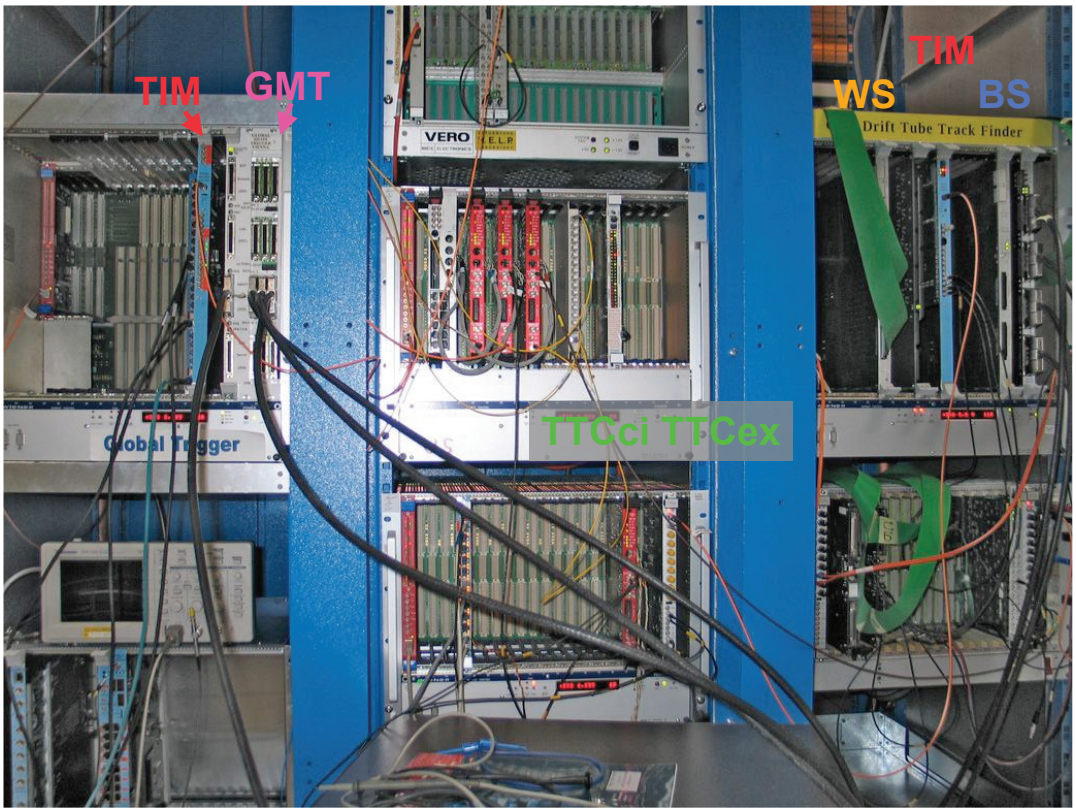


Figure 5.4: The CSC-GMT test setup in the Building 904 Trigger Integration Center (courtesy I. Mikulec)

started synchronously by issuing a BCRES (bunch counter reset) command from the TIM card via the backplane. The cards involved were all synchronized to a common clock generated by a PLL (Phase Lock Loop) and quartz oscillator integrated for test purposes onto the TIM card. Processed data was read out from the SRT chip spy memories and compared to the results simulated using the C++ emulator software. These tests showed some errors, however it was impossible to identify the precise causes because the MIP/Q data cannot be observed directly in the default firmware setup, but merely the results computed by the AUx chips.

Therefore in a second setup, the AUx testing firmware developed especially for this purpose was loaded into the AUx chips, and MIP/Q bits (test patterns with running bits and similar) were loaded into the PSB RAMs. Again, transfer was started by issuing a BCRES signal to both boards via the backplane, and the received data was read out from the AUx spy memories implemented in the testing firmware. The PSB patterns were translated into the corresponding AUx patterns using the C++ routines described earlier in this chapter and compared to the read out data. Using this technique, all the 252 signal lines were checked and a number of errors were identified and resolved by re-soldering and other minor hardware manipulations.

5.3.3 Interconnection Tests with the Regional Muon Triggers

Figure 5.2 shows the setup used to test interconnectivity with the DTTF Muon Sorter Board: a common 40.08 MHz (i.e. LHC bunch crossing frequency) was generated by a TTCci VME-TTC bridge card using the built in test QPLL circuit. The clock was transmitted via optical fiber to a TTCex optical fan-out module, and from there onto the TIM cards in the GT and DTTF crates. These cards convert the clock signal to electrical signals, lock on it using their own PLLs and distribute it via the backplane to all cards in the respective crates. Test patterns and emulator-generated muon data was injected into the simulation memories of the DTTF Wedge Sorter Board, which sent its output on to the Barrel Sorter and via four SCSI parallel cables to four muon inputs on the GMT front panel. The transfer was initiated by issuing a BCRES signal from TTCci card. The data was captured in the GMT Input chip spy memories and compared to the expected results. This way, all the 16 muon input channels of the GMT were tested (input format is very similar among the three Regional Triggers that deliver data to the GMT) and four errors were found which could be resolved. Experience was gained for synchronizing the two systems and the correct delay times were determined. A similar test was carried out CSC Muon Sorter board on the sending side (see figure 5.3).

Chapter 6

GMT Software Integration

Parallel to the interconnection tests between the GMT board and its neighbor cards in the L1 trigger, in the course of this thesis the GMT online software libraries were integrated into the Trigger Supervisor (TS) framework, which is in charge of controlling, monitoring and testing the L1 trigger and will act as a subsystem controller to the Run Control and Monitoring System (RCMS) of CMS. Several TS-controlled interconnection tests have been carried out, and classes have been implemented to facilitate the automatic configuration of the GMT from the trigger configuration database. A structure for persistent storage of configuration data in the database and a distributed file system were designed and implemented. Extensive tests were carried out on the TS framework, and possible technologies were explored.

6.1 Software Frameworks

6.1.1 XDAQ

The XDAQ Cross-Platform DAQ Framework [8] is the basis for both the Trigger Supervisor and many other applications of the CMS Trigger/DAQ, including the Event Builder, Builder Units, Event Manager and Trigger Throttling Systems. Implemented in C++, XDAQ provides infrastructure for distributed systems with inter-node communication and service discovery based on W3C Web Services.

A XDAQ process running on a node and identified by a host name and port (URI) is called an *executive* or a *context*. It can contain several XDAQ *applications* which implement some specific functionality and usually expose a Web Services API that allows for the functionality to be invoked and used by a client; many XDAQ applications also provide an HTML graphical user interface to human clients. Applications are implemented as shared libraries that can be loaded into the XDAQ executive at run time. The context acts as a container and service provider to the applications, dispatching the incoming messages and transmitting the outgoing ones, loading the applications and of course providing a library for various tasks, the most important of which are described below. An executive is configured using a configuration file parsed at startup. The following code listing shows an example configuration that loads the GMT cell and an access control application and notifies the executive of a 'TStore' application (see below) running in a different context, which allows the XDAQ library to provide more comfortable communication access:

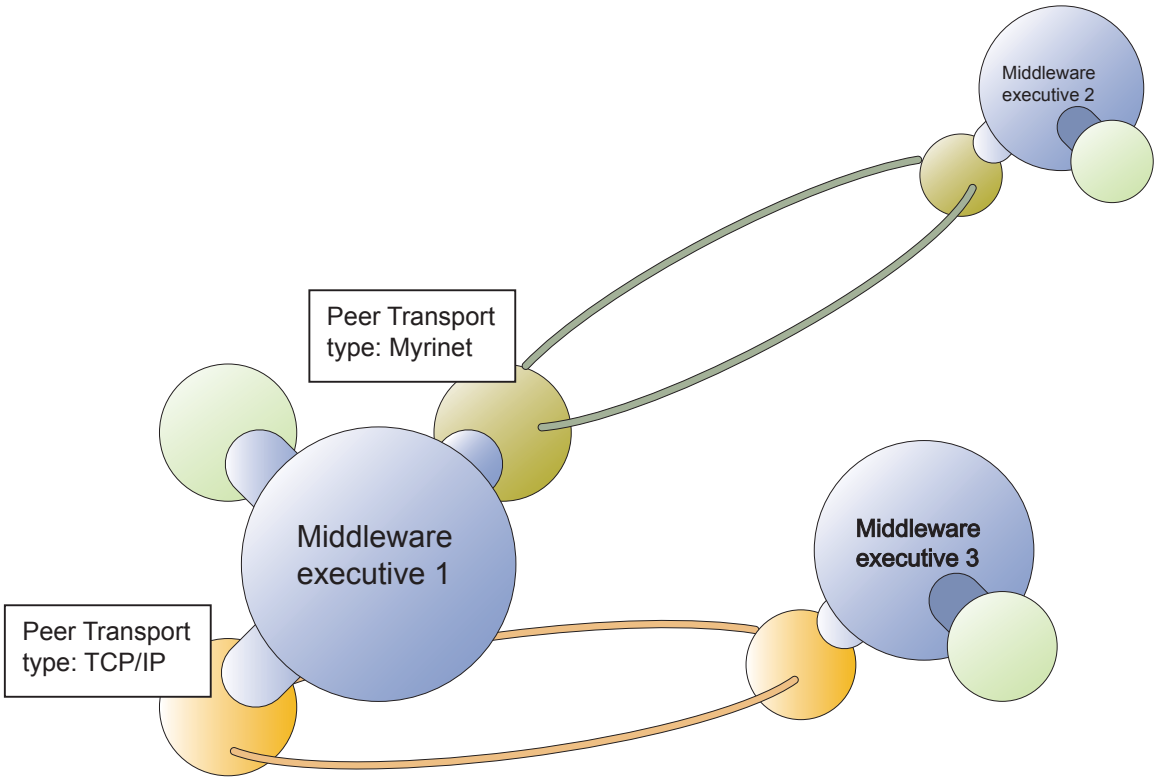


Figure 6.1: Communication over multiple networks through peer-transports (Source: [27])

```
<?xml version='1.0'?>
<!-- The following line contains a reference to an XSL file       -->
<!-- that allows the rendering of this XML file into an HTML file -->
<!-- <?xml-stylesheet type="text/xsl" href="xdaqConfig.xsl"?> -->

<xc:Partition xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
              xmlns:xc="http://xdaq.web.cern.ch/xdaq/xsd/2004/XMLConfiguration-30">

  <xc:Context url="http://vmepcS2G17-02:4328">
    <xc:Application class="Cell" id="13" instance="6" network="local">
      <properties xmlns="urn:xdaq-application:Cell" xsi:type="soapenc:Struct">
         <name xsi:type="xsd:string">GMT</name>
          <xhannelListUrl xsi:type="xsd:string">file://${XDAQ_ROOT}/trigger/gmt/ts/cell/xml/xhannelList/cell_xhannel_list_ts.xml
          </xhannelListUrl>
          <mirrorBasePath xsi:type="xsd:string">/opt/l1muon_afs_mirror/</mirrorBasePath>
      </properties>
    </xc:Application>

    <!-- Access control for standalone -->
    <xc:Application class="CellAccess" id="15" instance="0" network="local">
    </xc:Application>

    <xc:Module>file://${XDAQ_ROOT}/trigger/gmt/ts/cell/lib/linux/x86/libCell.so</xc:Module>
    <xc:Module>${XDAQ_ROOT}/trigger/ts/access/cell/lib/linux/x86/libCellAccess.so</xc:Module>
  </xc:Context>

  <xc:Context url="http://vmepcS2G17-02:3667">
    <xc:Application class="TStore" id="120" instance="0" network="local">
      <properties xmlns="urn:xdaq-application:TStore" xsi:type="soapenc:Struct">
        <views xsi:type="soapenc:Array" soapenc:arrayType="xsd:ur-type[1]">
          <item xsi:type="xsd:string" soapenc:position="[0]">${XDAQ_ROOT}/trigger/gmt/db/tstoreviews/gmtview.xml</item>
        </views>
      </properties>
    </xc:Application>

    <xc:Module>${XDAQ_ROOT}/daq/tstore/lib/linux/x86/libtstore.so</xc:Module>
    <xc:Module>${XDAQ_ROOT}/daq/tstore/tstore/lib/linux/x86/libTStore.so</xc:Module>
  </xc:Context>

</xc:Partition>
```

Web applications declared in such a way for command and control purposes communicate (i.e. invoke callbacks on each other) using the SOAP (Simple Object Access Protocol) standard published by the W3C, an XML-based message format and protocol suitable for calling methods on remote objects and retrieving the results. The applications may describe the services they export using WSDL (Web Services Description Language), and various service discovery protocols are implemented in XDAQ. Applications register the callbacks they export to the executive and respond according to the SOAP protocol.

The XDAQ suite furthermore includes the following important libraries:

xoap: This package provides helper methods for SOAP message creation, altering, sending and retrieving/parsing – the transport layer is decoupled however. Creation of a basic SOAP request message works as follow:

```
xoap::MessageReference msg = xoap::createMessage();
xoap::SOAPPart soap = msg->getSOAPPart();
xoap::SOAPEnvelope envelope = soap.getEnvelope();
xoap::SOAPBody body = envelope.getBody();
xoap::SOAPName command =
  envelope.createName("myMessage","xdaq", "urn:xdaq-soap:3.0");
body.addBodyElement(command);
```

If serialized, the resulting message looks like this and contains a simple dummy message:

```
<SOAP-ENV:Envelope
    xmlns:SOAP-ENV=http://schemas.xmlsoap.org/soap/envelope/
    SOAP-ENV:encodingStyle=http://schemas.xmlsoap.org/soap/encoding/>
  <SOAP-ENV:Body>
```

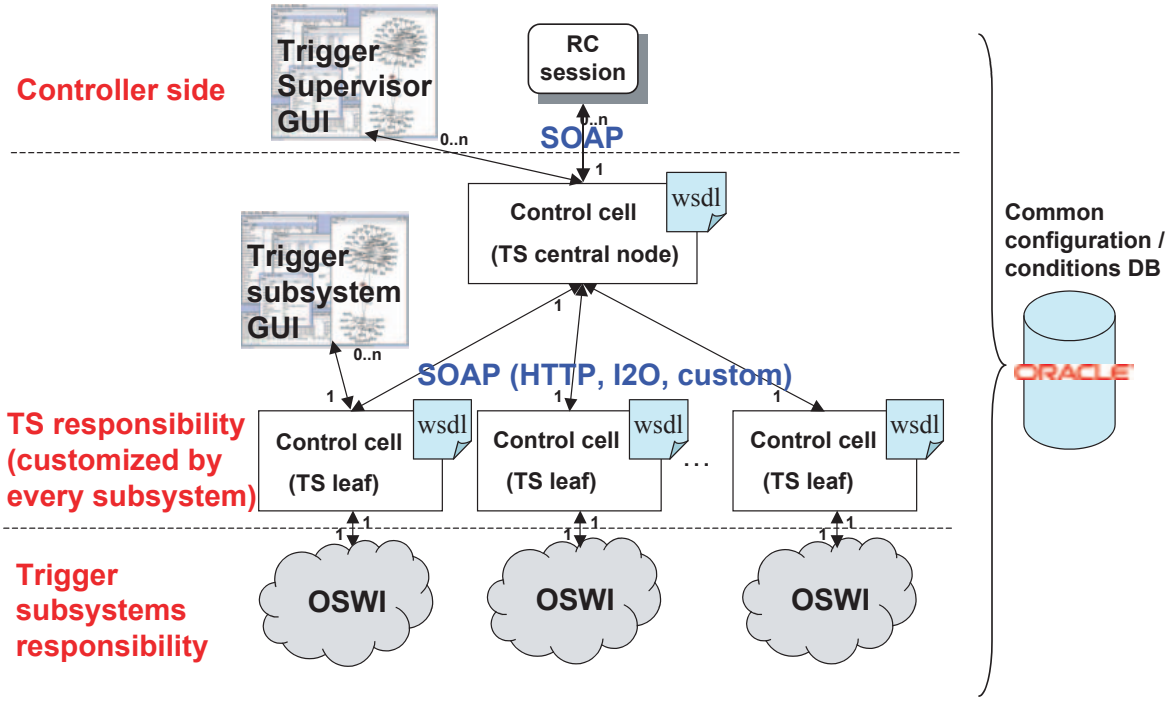


Figure 6.2: Architecture of the Trigger Supervisor (Source: [45])

```
    <xdaq::myMessage xmlns::xdaq="urn:xdaq-soap:3.0"/>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

xdata: allows for serializing and deserializing data types and structures. This ensures platform independence and allows for these data types to be safely transmitted via the various transport layers

Peer transport (pt): Transport layer services. Protocol implementations include `pthttp` (for SOAP messaging and web page display via HTTP), `pttcp` (for binary messaging using the TCP protocol), `ptfifo` (for inter-process communication using filesystem FIFOs)

I2O: Implementation of the I2O binary messaging standard that is far more efficient than text-based communication (as with SOAP) for large amounts of data as in the DAQ system.

xcept: various exception classes

xmas: a distributed monitoring and alarming suite which alows for information providers to announce the items they provide and for subscribers to collect them and react to status changes.

TStore: a Web Service (SOAP) frontend to a relational database. Allows for queries, inserts, updates and other SQL statements to be invoked using SOAP messaging.

toolbox: a great variety of auxiliary classes, including finite state machines, math functions, threading, smart and garbage collected memory management, event dispatching and others

6.1.2 Trigger Supervisor

The Trigger Supervisor (TS) [45, 43, 44] is a distributed system of Web Service applications built on top of the XDAQ framework. Its purpose is to control, configure, test and monitor all the hardware systems in the L1 trigger. It will act as a subsystem to the RCMS Run Control during automated operations ('runs'), but will also expose local graphical user interfaces for trigger subsystems. The TS is organized as a hierarchical (tree) structure of so-called 'Cells': each Cell is a XDAQ application running on a crate controller PC with a VME interface to the hardware. The top (root) Cell ('Central Cell') interfaces RCMS and issues SOAP commands to other Cells, which can also operate in standalone mode, i.e. without a connection to the Central Cell. The main concepts and components of the TS framework are described in the following:

Cell

Figure 6.3 shows an UML diagram of the main classes and objects involved in a TS Cell: each Cell implements the abstract base class `CellAbstract`, which in turn inherits from `xdaq::Application` and thus acts as a XDAQ application that can export a SOAP and HTML/CSS/Javasript interfaces to clients. In the constructor of a Cell implementation, all the Commands and Operations as well as SOAP and HTML callback functions are registered,

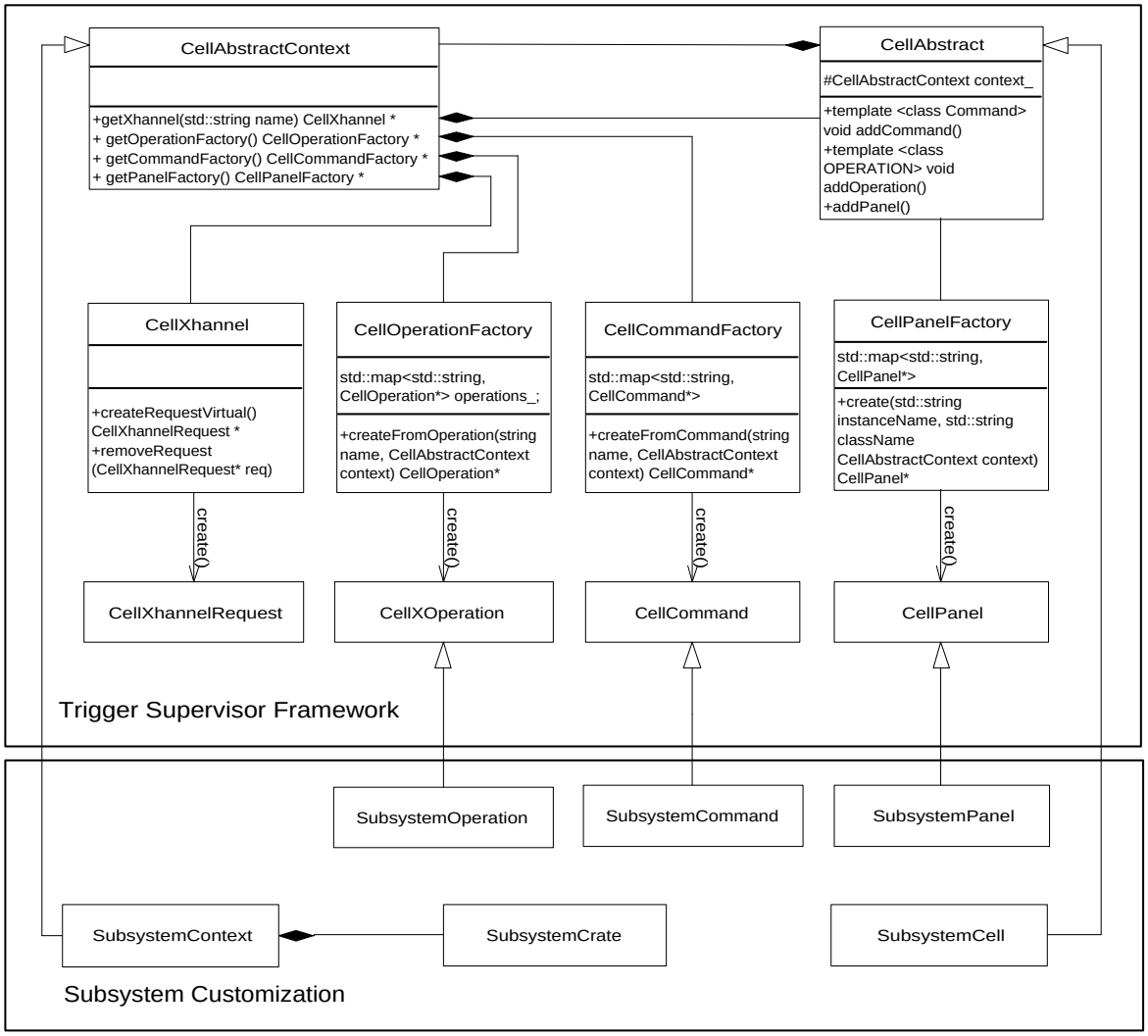


Figure 6.3: UML diagram of the components of a TS Cell (Source: [37])

and a `CellContext` (for which `CellAbstractContext` provides most functionality already) object is instantiated and assigned to a member variable. `CellContext` is a container for shared objects in the Cell: it holds pointers to all the utility objects, e.g. `CellAccess` to manage user authentication, authorization and login sessions and various factory classes, which encapsule and manage the creation and deletion of objects such as `CellCommand`s, `CellOperation`s and `CellPanel`s. In subsystem Cell implementations, the CellContext might e.g. also hold a pointer to an object of the underlying hardware driver library. The cell automatically provides a generic Web user interface to the commands and operations defined (see below), and specialized panels can be provided that offer a more customized user interface for complex control tasks.

Commands

Command objects (which extend the CellCommand class) can be thought of as encapsulation objects for a function: they can take parameters, perform one or more actions (e.g. read from hardware, write to database) and return a return value or error message. For each Command class known to a Cell, a generic Web user interface is provided that allows for creation, parameter input, execution and deletion of Command objects; at the same time, the commands can be invoked by sending appropriate SOAP messages to the Cell via http, basically specifying the Command class name and parameter values. The Cell will respond with a SOAP reply containing the return value or an error message.

Operations

Operation objects (which extend `CellOperation`) are finite state machines (FSM); during the transitions from one state to another, some action is performed (e.g. Command objects are invoked, or some arbitrary code is executed). Figure 6.5 shows an example FSM for a hardware configuration operation. An operation can accept parameters as well, and just like for commands, a generic Web and SOAP client interface is provided by the Cell, including an image visualization of the FSM in the Web interface.

'Xhannels'

Since the creation and (a)synchronous submission and reception of SOAP messages is rather tedious, complicated and possibly subject to change, the subclasses of `CellXhannel` help hide these complications from the user and act as an abstraction layer should the SOAP interface of the target service change. The Xhannel objects are declared in an XML configuration file and automatically instantiated at Cell load time; they have knowledge of the target URI of some service (such as the TStore SOAP frontend for relational databases, or some other Cell) and the specifics of SOAP messaging with that service.

AjaXell Web GUI Framework

In order to provide a rich user interface to human users, a client- and server-side library was developed that uses the AJAX (Asynchronous Javascript and XML) set of technologies to produce responsive web pages that do not just follow the GET/POST principle of traditional pages. Based on the Dojo Javascript library, a number of widgets have been made available that are common to usual desktop applications (trees, tabs, tables, etc.). The widgets

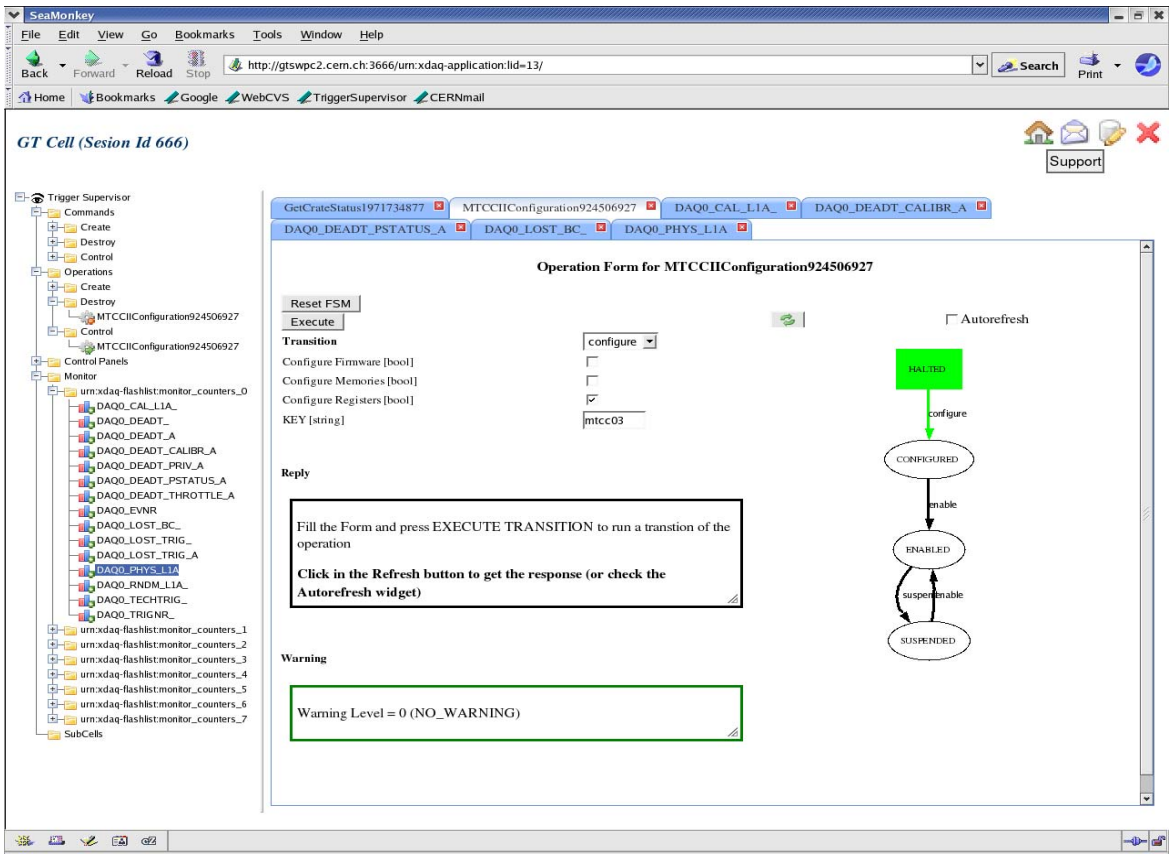


Figure 6.4: Screenshot of the AJAX-based TS graphical user interface (Source: [44])

running in the user's browser communicate with the Cell in a XDAQ executive by sending XML messages, specifying that some event has occurred (e.g. tree node expand icon clicked, command execution button clicked), and receiving the answer asynchronously (i.e. without blocking the web page or reloading it entirely).

Central Cell

As mentioned already, the Central Cell is the root node of the TS tree of cells. It issues SOAP commands to all the subsystem cells, e.g. during a configuration operation. During runs, the Central Cell is under control of a RCMS subsystem controller.

Database-driven Configuration Strategy

For automatic configuration of the entire trigger system, every subsystem and the central cell will implement a configuration operation with the same finite state machine. The operation takes one parameter: the id of a row in the configuration database, which contains the configuration values, or pointers (foreign keys) to them. As illustrated by figure 6.6, the RCMS invokes the Configure(Key) transition of the central cell configuration operation, which looks up the subsystem config ids in a trigger config table and invokes the same transition in all the subsystems, passing them the respective keys. The subsystems are responsible for developing a database structure that enables them to retrieve all the configuration data associated to a configuration id.

6.1.3 Contributions to the TS Framework

User Interface Technology Evaluation

In the course of this thesis, a technology evaluation was carried out to determine the technology best suitable for rich Web based user interfaces of the TS Cells. Apart from the functionality requirements, the main factors taken into account were development speed, available expertise and long-term maintainability. In particular, the two alternatives AJAX and Java applets were investigated. Since the conclusion was that Java applets would be most suitable for this application scenario (primarily due to the expected speed of development and already available Java GUIs), a proof of concept applet was implemented, that generated a user interface for SOAP communication with the underlying Cell from an XML file specifying the user interface components. An XML data binding library allowed for quick translation into a Java data model. An XML Schema was defined for the configuration file, and the proof of concept applet was deployed onto a test server. While the result was satisfying on an Apache web server, using XDAQ as a server for the applet turned out to be impractical: the XDAQ pthttp library only implements a subset of the HTTP protocol, which lead to too long load times due to the lack of partial GET and conditional GET implementations and non-standard reactions to HTTP 404 (file not found) events. The concept therefore was dropped, and later the AJAX variant was implemented as described above.

Debugging and Tests

During the early phases of TS development, great effort was put into testing and debugging the code. In particular, several integration tests with multiple Cells involved were carried

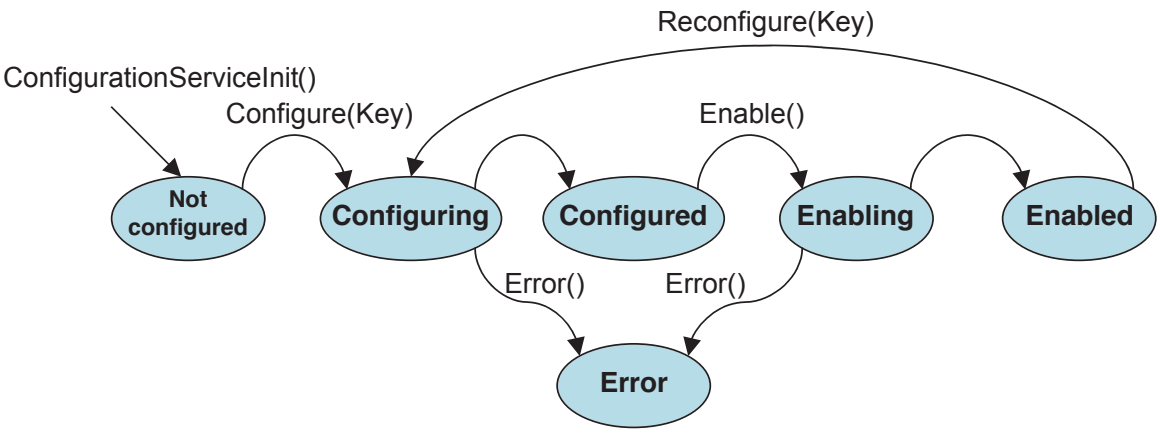


Figure 6.5: Finite state machine used for trigger configuration (Source: [45])

out that are described in sectionintegration. A first configuration demonstrator setup was presented to the CMS Trigger integration project management involving three cells.

TS Webpage and Logo design

A homepage [7] based on the Web content management system Joomla was set up and adapted for the Trigger Supervisor project, allowing for multi-user dynamic content publishing. In order to provide a recognizable graphical identity, a Trigger Supervisor logo was designed that is loosely based on the CMS logo and thus provides a link to the experiment visual identity. Both logos are shown in figures 6.7 and 6.8.

6.2 GMT TS Cell

In the course of this thesis a TS cell for the GMT subsystem was implemented in basic form. Since this thesis was carried out during a very early development phase of the TS framework, much effort had to be invested into porting the cell to new releases of the TS API or the underlying XDAQ framework, and it was used extensively as a test case for the TS framework. Therefore the GMT cell cannot be considered complete as of the end of this thesis, but it should be possible to implement the necessary functionality quite quickly as soon as the surrounding code bases have been consolidated. In the following, the classes involved in the GMT cell are described:

6.2.1 Main Classes

Cell

This class is derived from `CellAbstract` consequently from xdaq::Application. It is the XDAQ application that provides the GMT cell.

Fields:

`private xdata::String mirrorBasePath_` holds the base path on the local file system where the GMT AFS configuration file repository is being mirrored to improve performance (see section 6.3. This parameter can be set in the `profile.xml` file that loads the GMT cell (see listing in section 6.1.1):
`<mirrorBasePath xsi:type="xsd:string">/opt/l1muon_afs_mirror/</mirrorBasePath>`

Methods:

`public Cell(xdaq::ApplicationStub *s)` instantiates the (singleton) `CellContext` object and assigns it to the cellContext_ field inherited from CellAbstract. It also registers the `mirrorBasePath` variable to the `ApplicationInfoSpace` object provided by XDAQ and binds it to the `mirrorBasePath_` field. This ensures that this parameter is read from the configuration file and assigned to the field.

`public void init()` Here, all the available commands and operations in the cell are registered to the CellFactories

`public void actionPerformed(xdata::Event &e)` listens to `ItemChangedEvents` on the mirrorBasePath_ parameter and passes on any changes to the `CellContext` object

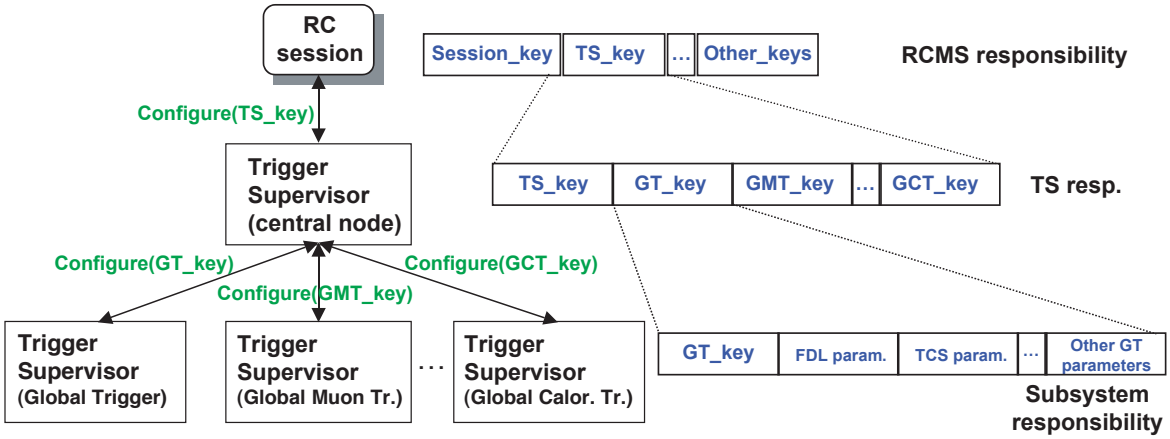


Figure 6.6: Database-driven configuration strategy for the TS (Source: [45])

CellContext

This class is derived from `AbstractContext` and additionally holds some objects necessary for GMT operation

Fields:

`private GTCrate *gtCrate_` holds a pointer to the `GTCrate` object of the GMT online software library which provides access to the hardware drivers needed in the GMT cell, such as the PSB and GMT ones.

`private string mirrorBasePath_` see `Cell::mirrorBasePath_`; this is a duplication of the corresponding `Cell` field, since that object is not available to commands and operations

Methods:

`public void setGTCrate(GTCrate* gtCrate)` sets the `GTCrate` used by all the other classes in the GMT cell. This method takes ownership of the object pointed to by the `gtCrate` argument, i.e. it is deleted in the DTOR, or if this method is called again with a different argument.

`public GTCrate* const& getGTCrate()` returns a reference to the pointer to the `GTCrate` object set by `setGTCrate()` (that pointer can be 0 if the field hasn't been initialized).

`public void setMirrorBasePath(string path), const string& getMirrorBasePath()` 'setter' and 'getter' methods for the `mirrorBasePath_` field

6.2.2 Commands

Since all commands implement the skeleton defined by `CellCommand`, not the fields and methods are described here but what parameters can be passed to the commands and the functionality implemented in the `code()` method.

GmtInitializeCrateObject

Parameters:

`xdata::String busAdapter` a string identifying the VME bus adapter driver to use. Possible values: 'CAEN', 'SBS' or 'DUMMY'

This command creates a new `GTCrate` object with the specified bus adapter driver type and passes it to `CellContext::setGTCrate()`. It also enables the other commands, which are disabled by default since they depend on an instance of `GTCrate`.

GmtReadRegister

Parameters:

`xdata::String RegisterName` a valid name of a GMT register or memory base

Reads the value of the register specified and returns it.

Figure 6.7: Trigger Supervisor logo

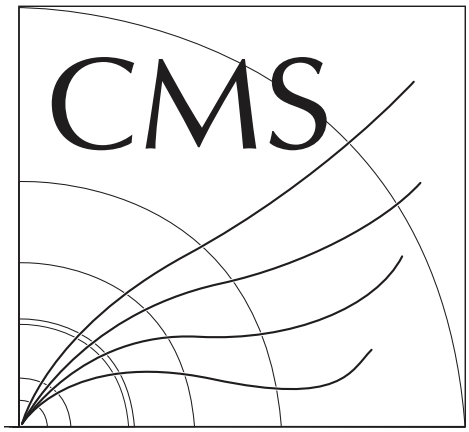


Figure 6.8: CMS logo

GmtConfigureFromDB

Parameters:

`xdata::String GMTKey` the row ID in the configuration database that identifies the GMT configuration to load

`xdata::Boolean configureRegisters` whether or not to write configuration data to the GMT registers. This takes less than one second to complete.

`xdata::Boolean configureLUTs` whether or not to re-write the GMT LUTs according to the configuration in the database. This may take up to one minute to complete.

`xdata::Boolean configureFirmware` whether or not to re-write the GMT firmware according to the configuration in the database. This may take several minutes to complete.

This rather complex operation fetches the GMT configuration identified by `GMTKey` from the configuration database (in fact this action is performed by the auxiliary class `GmtDBRecord` which encapsulates the specifics of a database schema and the data transfer; see below) and uses it to configure the registers, LUTs and firmware as requested. While register configuration is rather straightforward since the register values are available directly from the DB (see 6.3 below), in case of the LUTs and firmware the DB only holds URL to the files to write to the hardware. Therefore this command uses auxiliary class `GtHttpDownloader` to get these files either from the file repository or the local cache. Then, the GMT online software library is used to write these files to the hardware.

GmtInsertRegisterConfigFromHardwareState

Parameters:

`xdata::String "Primary key for new DB row"` the new primary key to identify the current configuration with in the DB. If this primary key is already in use, an error is returned

This commands reads the values of all the configuration registers from the hardware and creates a new row in the register configuration table in the DB containing these values. This command has to be considered 'experimental' code, since the required TS `Xhannel` object for communication with the DB was not stable when this thesis was finished.

6.2.3 Operations

MTCCIIConfiguration.cc

Parameters:

`xdata::String KEY` the row ID in the configuration database that identifies the GMT configuration to load

`xdata::String busAdapter` a string identifying the VME bus adapter driver to use. Possible values: 'CAEN' or 'DUMMY'

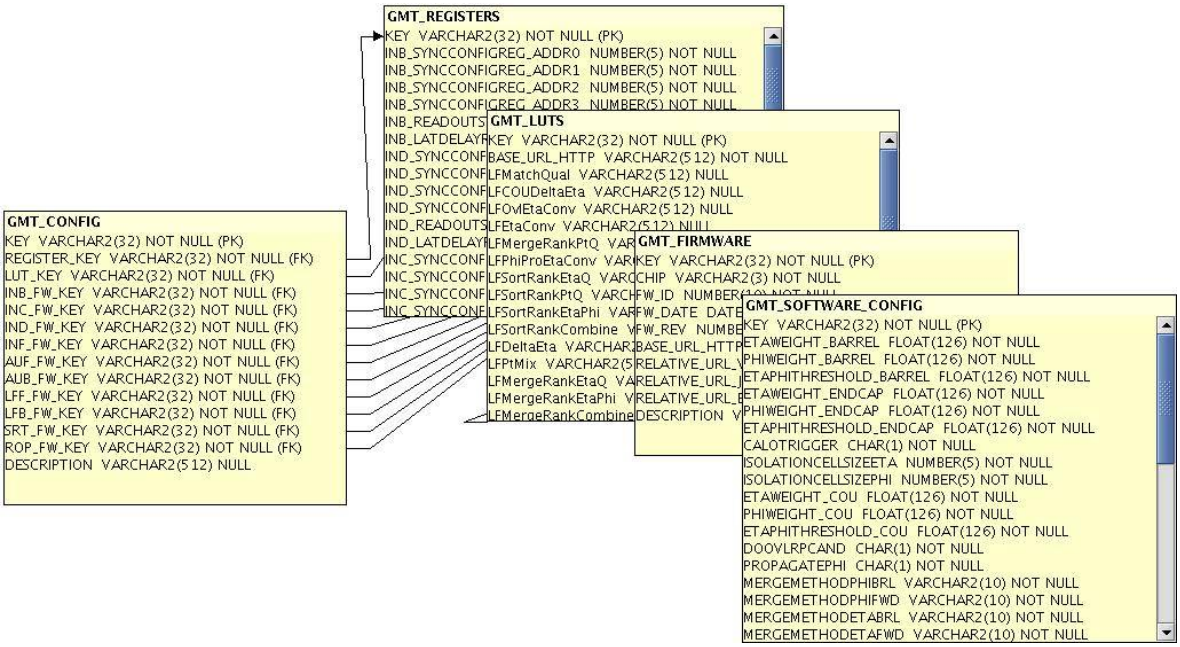


Figure 6.9: GMT configuration database layout

In the `configure` transition of its FSM, this operation invokes the `GmtInitializeCrateObject` command to (if necessary) re-instantiate the GTCrate object using the bus adapter specified. It then invokes the `GmtConfigureFromDB` command to configure the GMT with data from the DB row identified by the `KEY` parameter. This operation simply encapsulates the functionality provided by the two commands in order to become compatible with the configuration operations as required by the TS central cell. It was used as for testing the configuration operations during the CMS magnet test and 'Cosmic Challenge II' test run in spring 2007.

CellItDttfGmt

No parameters.

This operation was used in one of the integration tests described below; it can be used as an example of an interconnection test between the PSBs and the GMT: the GMT online software library class `GmtPsbInterconnectionTest` is used to download a bit pattern into the PSB simulation memories, transfer it to the GMT and read out from the testing firmware spy RAMs in the GMT AUx chips. The expected result and the actual one are compared and errors reported.

Deprecated Operations

The TS operation classes `CellConfiguration`, `CellSelfTest`, `CellInterconnectionTest`, `CellGmtConfiguration` and `CellGmtSelfTest` are all deprecated, they were used for testing and debugging purposes or in deprecated releases of the TS framework.

6.2.4 Auxiliary Classes

GmtDBRecord

This class fetches a GMT configuration record from the DB and stores it in member fields, providing access methods. It thereby hides the details of the database schema (which might be subject to change) as well as the data transfer method. Currently, two transfer methods are available: one using the deprecated DStore library and the `CellXhannelDB` xhannel, the other one the current TStore library and the `CellXhannelDBTStore` xhannel. The corresponding methods for fetching the data are: `public void GmtDBRecord::fillFromDB(const string &key)` and `public void GmtDBRecord::fillFromDBTStore(const string &key)`. As soon as the data has been retrieved, the `public bool isInitialized()` method returns true, and references to the corresponding hashmaps with name–value associations can be retrieved using getter methods. Two `struct`s and several `typedef`s are defined in the class header to store and access LUT and firmware info in a handy way (using typedef'ed iterators etc.).

GtHttpDownloader

By using the CURL C library [2], this class (which is compiled into the GTBase library that provides common functions for GT and GMT) can be used simple for http-downloads, with the result delivered in form of a string (`const string& download2string(const string &url)`), filestream (`void download2filestream(const string &url,FILE *filestream)`) or file (`download2file(const string &url, const string &saveAsFilename)`).

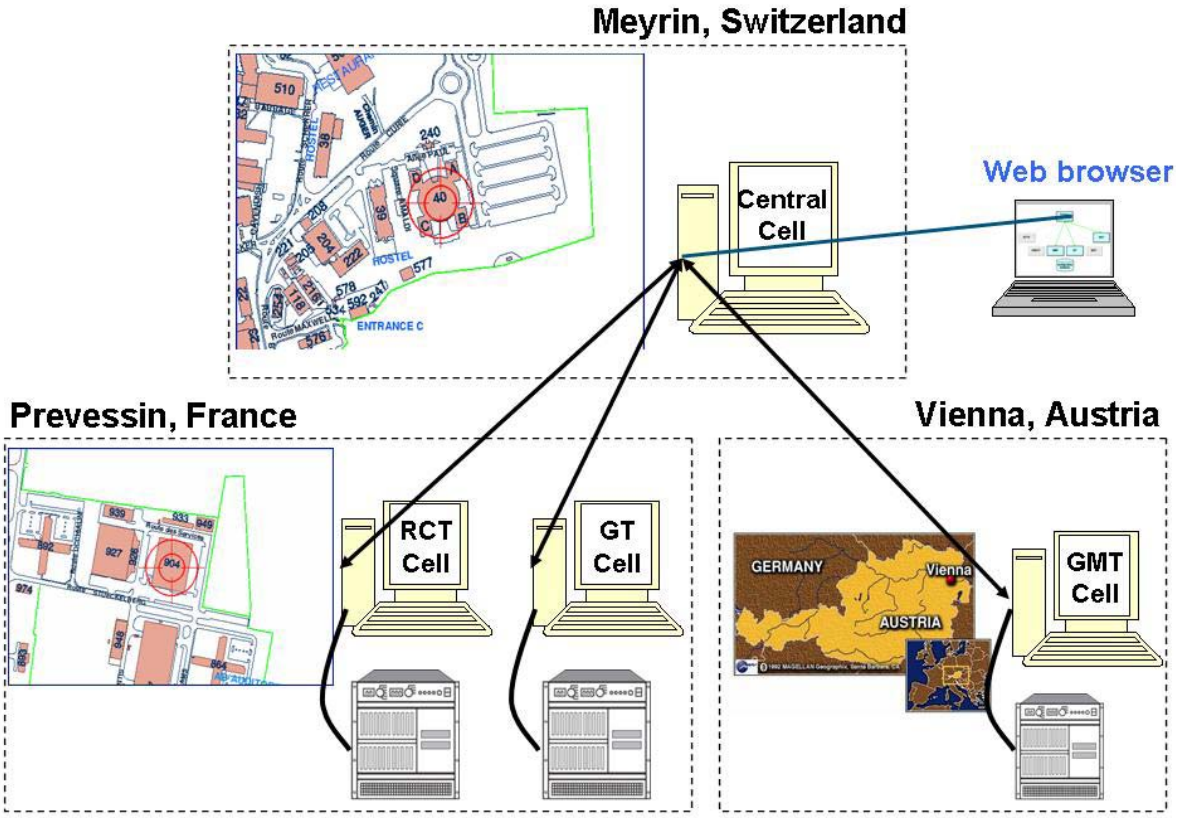


Figure 6.10: Setup for the first TS-driven interconnection test (Source: [43])

Plus, the class offers cached downloads (`download2fileMirrored(const string &url, const string &mirroredFilename, const string &saveAsFilename="")`): in this case a local mirror path is searched for the filename and relative path requested; if found, the local copy is used, if not, an http download is initiated.

6.3 Configuration Database and File Repository

In order to hold all the values needed for configuring the GMT, a database layout has been designed (see figure 6.9): the `GMT_REGISTERS` table simply associates a primary key with values for all the configuration-relevant registers. The `GMT_LUTS` table per row has a `base_url_http` column, which points to the HTTP base URL of the individual LUT files, the relative URLs of which are given in the other columns, one per LUT type. The `GMT_FIRMWARE` table uses the same separation to point to the various formats of firmware files and additionally contains columns giving the firmware ID, date and revision corresponding to a firmware file. The `GMT_SOFTWARE_CONFIG` table contains the CMSSW GMT emulator parameters used to generate the LUTs and configuration register values in a configuration. In the future, these emulator parameters should be used to generate the LUTs etc. on the fly. The `GMT_CONFIG` main table finally associates an overall GMT config ID with foreign keys pointing to rows in the aforementioned tables and representing one consistent GMT setup.

Since the Configuration Database infrastructure is unable to hold large BLOBs (binary large objects), the LUT and firmware files cannot be stored directly in the DB. Instead, a structure for a file repository has been designed that is hosted on the CERN AFS distributed file system and can be accessed via an HTTP server. In oder to save transfer time and improve availability, the AFS repository is mirrored to the local disc of the crate controller PC using a `cron` job and the `rsync` tool. The `GtHttpDownloader` class mentioned above can be used to download the files form the HTTP repository or use the local copy if available. The AFS repository also holds large files for GT configuration as well as test files and documentation.

6.4 Integration Tests

Several integration tests have been undertaken (incorporating the GMT TS cell), in which more than one TS subsystem cell were controlled by the TS central cell to perform some coordinated operation, e.g. database-driven configuration as described in section 6.1.2. In particular, interconnection tests between hardware boards have been carried out using the TS as a controller: in this scenario, common synchronization is set up for all the participating systems, the sending subsystem loads test patterns into its simulation memories, transfer is triggered and the receiving subsystem captures data, which is read out and compared to an expected result. The TS central cell during this test triggers FSM transitions in the subsystem cells using SOAP commands and collects the results. Such tests were performed involving the GMT, DTTF and GT (PSB, GTL) subsystems, and the protocol concept was shown to work well.

Bibliography

[1] CMSSW reconstruction and analysis framework. http://cmsdoc.cern.ch/cms/cpt/Software/html/General/.

[2] CURL C library. http://curl.haxx.se/.

[3] GMT online software class reference. http://wwwhephy.oeaw.ac.at/p3w/cms/trigger/globalMuonTrigger/onlinesw/html/index.html.

[4] GMT online software CVS web interface. http://isscvs.cern.ch/cgi-bin/cvsweb.cgi/TriDAS/trigger/gmt/GMT/?cvsroot=tridas.

[5] ORCA object oriented reconstruction and analysis framework. http://cmsdoc.cern.ch/orca/.

[6] SPEC standard performance evaluation corporation. http://www.spec.org/.

[7] Trigger supervisor project webpage. http://triggersupervisor.cern.ch.

[8] XDAQ cross-platform DAQ framework. http://xdaqwiki.cern.ch.

[9] S. Abdullin et al. Summary of the CMS potential for the Higgs boson discovery. *Eur Phys J C*, 39, 2004. DOI:10.1140/epjcd/s2004-02-003-9.

[10] P. Azzurri. The CMS silicon strip tracker. CMS Conference Report 2005/034, 2005. http://cms.cern.ch/iCMS/jsp/openfile.jsp?type=CR&year=2005&files=CR2005_034.pdf.

[11] Y Ban, J Cai, H Liu, S Qian, Q Wang, Y Ye, Y Ying, B Hong, Q Wang, S J Hong, K S Lee, S Park, K S Sim, A Ball, J P Chatelain, I Crotty, A Sharma, W Whitaker, W K Van Doninck, S Akimenko, L Litov, A Marinov, M I Asghar, I Ahmed, H Hoorani, A Colaleo, G Iaselli, F Loddo, and M Maggi. Beam test of the first production forward RPC. Technical Report CMS Note 2004-037, CERN, Geneva, Jan 2005.

[12] Vincenzo Barone and Enrico Predazzi. *High-Energy Particle Diffraction*. Springer, 2002.

[13] M Bellato, L Berti, V Brigljevic, G Bruno, E Cano, Sergio Cittolin, Akos Csilling, S Erhan, D Gigi, F Glege, R Gómez-Reino, M Gulmini, J Gutleber, C Jacobs, Miklos Kozlovszky, H Larsen, Ildefons Magrans de Abril, G Maron, F Meijers, E Meschi, S Murray, A Oh, L Orsini, L Pollet, A Rácz, G Rorato, D Samyn, P Scharff-Hansen, C Schwick, Paris Sphicas, N Toniolo, S Ventura, and L Zangrando. Run control and monitor system for the cms experiment. Technical Report cs.DC/0306110, CERN, Geneva, Jun 2003.

[14] Michael Benedikt, Paul Collier, V Mertens, John Poole, and Karlheinz Schindl, editors. *LHC Design Report*, volume 3: The LHC Injector Chain. CERN, Geneva, 2004.

[15] Olivier Sim Brüning, Paul Collier, P Lebrun, Stephen Myers, Ranko Ostojic, John Poole, and Paul Proudlock, editors. *LHC Design Report*, volume 1: The LHC Main Ring. CERN, Geneva, 2004.

[16] Markus Brugger, Massimiliano Fierro, and Claudia-Elisabeth Wulz. Drift tube based pseudorapidity assignment of the level-1 muon trigger for the compact muon solenoid experiment at CERN. *Nuclear Instruments and Methods in Physics Research Section A*, 482:254–270, 2002.

[17] G. Charpak, R. Bouclier, T. Bressani, J. Favier, and Č. Zupančič. The use of multiwire proportional counters to select and localize charged particles. *Nuclear Instruments and Methods*, 62:262–+, 1968.

[18] The ALICE Collaboration. *ALICE Technical Proposal*. CERN/LHCC 1995-071, 1995. http://doc.cern.ch//archive/electronic/other/generic/public/cer-000214817.pdf.

[19] The ATLAS collaboration. *ATLAS: technical proposal for a general-purpose pp experiment, at the Large Hadron Collider at CERN*. CERN/LHCC 1994-043, 1994. http://atlas.web.cern.ch/Atlas/TP/tp.html.

[20] The CMS Collaboration. *CMS Technical Proposal*. CERN/LHCC 1994-038, 1994. http://cmsdoc.cern.ch/TPref/TP.html.

[21] The CMS Collaboration. *CMS ECAL Technical Design Report*. CERN/LHCC 1997-033, 1997.

[22] The CMS Collaboration. *CMS HCAL Technical Design Report*. CERN/LHCC 1997-031, 1997.

[23] The CMS Collaboration. *CMS Muon Project: Technical Design Report*. CERN/LHCC 1997-032, 1997.

[24] The CMS Collaboration. *The Magnet Project Technical Design Report*. CERN/LHCC 1997-010, 1997. http://cmsdoc.cern.ch/ftp/TDR/MAGNET/magnet.html.

[25] The CMS Collaboration. *The Tracker Project Technical Design Report*. CERN/LHCC 1998-006, 1998.

[26] The CMS Collaboration. *The CMS tracker : addendum to the Technical Design Report*. CERN/LHCC 2000-016, 2000.

[27] The CMS Collaboration. *The Trigger and Data Acquisition project, Volume 1, The Level-1 Trigger Technical Design Report*. CERN/LHCC 2000-038, 2000.

[28] The CMS Collaboration. *The Trigger and Data Acquisition project, Volume 2, Data-acquisition and High-level Trigger Technical Design Report*. CERN/LHCC 2002-026, 2002.

[29] The CMS Collaboration. *CMS Physics Technical Design Report, Volume 1, Detector Performance and Software*. CERN/LHCC 2006-001, 2006.

[30] The CMS Collaboration. *CMS Physics Technical Design Report, Volume 2, Physics Performance*. CERN/LHCC 2006-021, 2006.

[31] The LHCb Collaboration. *LHCb Technical Proposal*. CERN/LHCC 1998-004, 1998. http://lhcb-tp.web.cern.ch/lhcb-tp/.

[32] The LHCf Collaboration. *LHCf Experiment Technical Design Report*. CERN/LHCC 2006-004, 2006.

[33] M. Della Negra, K. Eggert, M. Lanzagorta, M. Pimia, and F. Szoncso. Compact muon solenoid. In Göran Jarlskog and Dieter Rein, editors, *Proccedings of the Large Hadron Collider Workshop, Aachen, Germany*, volume 3, pages 547–563. CERN, 1990.

[34] D. Denegri. Standard model physics at the LHC. In Göran Jarlskog and Dieter Rein, editors, *Proccedings of the Large Hadron Collider Workshop, Aachen, Germany*, volume 1, pages 60–117. CERN, 1990.

[35] Janos Erö. New approach for the CMS muon trigger track finder processor. In *Proc. of the 5th Conference on Electronics for LHC Experiments LEB '99, Snowmass, CO, USA*, pages 309–313, 1999.

[36] P. Glaser, T. Nöbauer, H. Bergauer, M. Padrta, A. Taurok, and C.-E. Wulz. Design and development of a graphical setup software for the CMS global trigger. *IEEE Transactions on Nuclear Science*, 53:1282–1291, 2006.

[37] Philipp Glaser. System Integration of the Global Trigger for the CMS Experiment at CERN. Master's thesis, Vienna University of Technology, 2007.

[38] Manfred Jeitler. The performance of the NA48 detector. *Nucl. Instrum. Methods Phys. Res., A*, 478(1-2):404–410, 2002.

[39] A. Kluge and T. Wildschek. The hardware muon trigger track finder processor in CMS - architecture and algorithm. Technical Report CMS Note 1997-092, CERN, Geneva, 1997.

[40] J. Knobloch and L. Robertson, editors. *LHC Computing Grid Technical Design Report*. CERN/LHCC 2005-024, 2005.

[41] S. Lami. TOTEM: The experiment to measure the total proton-proton cross section at LHC. *arXiv:hep-ex/0612049v1*, 2006.

[42] Christian Lippmann. *Detector Physics of Resistive Plate Chambers*. PhD thesis, Johann Wolfgang Goethe-Universität, Frankfurt am Main, 2003. http://deposit.ddb.de/cgi-bin/dokserv?idn=968973019.

[43] Ildefons Magrans de Abril and Marc Magrans de Abril. The CMS trigger supervisor project. IEEE 2005 Nuclear Science Symposium Conference Record http://triggersupervisor.cern.ch/index.php?option=com_docman&task=doc_download&gid=6, 2005.

[44] Ildefons Magrans de Abril and Marc Magrans de Abril. Enhancing the user interface of the CMS level 1 trigger online software with AJAX. Proceedings of the Real Time Conference 2007 http://triggersupervisor.cern.ch/index.php?option=com_docman&task=doc_download&gid=37, 2007.

[45] Ildefons Magrans de Abril, Claudia-Elisabeth Wulz, and João Varela. Concept of the CMS trigger supervisor. *IEEE Trans. Nucl. Sci.*, 53:474–483, 2006.

[46] Ivan Mikulec. Measurement of the direct CP violation by NA48 experiment at CERN. *Nucl Phys B, Proceedings Supplement*, 99(3):47–85, 2001.

[47] Alexander Nentschev. Development of a Set-Up Software for the Global Trigger of the CMS-Experiment at the LHC at CERN. Master's Thesis, Vienna University of Technology, 2004.

[48] Norbert Neumeister, Paul Porth, and Herbert Rohringer. Simulation of the global muon trigger. CMS Internal Note 1997-023, http://wwwhephy.oeaw.ac.at/p3w/cms/trigger/globalMuonTrigger/notes/cms-note-97_023/gmt.html, 1997.

[49] W. K. H. Panofsky. The evolution of particle accelerators & colliders. *Beam Line*, 27:36–44, 1997.

[50] M Pieri, G Maron, A Brett, E Cano, Sergio Cittolin, S Erhan, D Gigi, F Glege, R Gómez-Reino Garrido, M Gulmini, J Gutleber, C Jacobs, F Meijers, E Meschi, A Oh, L Orsini, L Pollet, A Rácz, P Rosinsky, H Sakulin, C Schwick, J Varela, J Branson, S Murray, I Suzuki, R Arcidiacono, K Sumorok, and V Brigljevic. CMS DAQ event builder based on gigabit ethernet. Technical Report CMS-CR-2006-033. CERN-CMS-CR-2006-033, CERN, Geneva, Jun 2006.

[51] Carlo Rubbia. A few considerations of strategy on the future of CERN (opening talk). In Göran Jarlskog and Dieter Rein, editors, *Proccedings of the Large Hadron Collider Workshop, Aachen, Germany*, volume 1, pages 9–39. CERN, 1990.

[52] Anton Sakulin, Hannes und Taurok. The Level-1 Global Muon Trigger for the CMS Experiment. Presented at: 9th LECC 2003 Workshop, Amsterdam, Holland, September 29 to October 3, 2003. CMS Conference Record 2003-040, http://cmsdoc.cern.ch/documents/03/cr03_040.pdf, 2003.

[53] Hannes Sakulin. JTAG access library. http://cmsdoc.cern.ch/~hsakulin/jal/.

[54] Hannes Sakulin. *Design and Simulation of the First Level Global Muon Trigger for the CMS Experiment at CERN*. PhD thesis, Vienna University of Technology, 2002. http://wwwhephy.oeaw.ac.at/p3w/cms/trigger/globalMuonTrigger/notes/phd_hs.pdf.

[55] Hannes Sakulin. A robust solution to the ghosting problems of the CMS level-1 muon trigger in the barrel/endcap overlap region. CMS Internal Note 2002-024, http://wwwhephy.oeaw.ac.at/p3w/cms/trigger/globalMuonTrigger/notes/in02_024.pdf, 2002.

[56] Hannes Sakulin and Massimiliano Fierro. Studies of the global muon trigger performance. CMS Note 2001-003, http://wwwhephy.oeaw.ac.at/p3w/cms/trigger/globalMuonTrigger/notes/note01_003.pdf, 2001.

[57] Frank Peter Schilling. Prospects for measuring $b_s^0 \to \mu^+\mu^-$ with the CMS detector. Technical Report CMS-CR-2005-022. CERN-CMS-CR-2005-022, CERN, Geneva, Sep 2005.

[58] C. Schwick. HAL hardware access library. http://cmsdoc.cern.ch/~cschwick/software/documentation/HAL/.

[59] Wesley Smith and Joao Varela. CMS-LHC signal exchange. Presentation to Ad-Hoc Working Group on LHC Machine Parameters and Signal Exchange, June 19, 2002 http://cmsdoc.cern.ch/~wsmith/LHCWG_0602.pdf.

[60] Paris Sphicas. Trigger and data acquisition at the LHC. Lecture for 2006 CERN Summer Students Program, http://agenda.cern.ch/askArchive.php?base=agenda&categ=a062849&id=a062849%2Ftransparencies%2FCERN_Summer_Students_2006_Sphicas.pdf, 2006.

[61] A. Taurok, H. Bergauer, and M. Padrta. Implementation and synchronisation of the first level global trigger for the CMS experiment at LHC. *Nucl. Instrum. Meth.*, A473:243–259, 2001.

[62] C. E. Wulz. Concept of the first level global trigger for the CMS experiment at LHC. *Nucl. Instrum. Meth.*, A473:231–242, 2001.

[63] Alexandre Zabi. Test beam results on the performance of the CMS electromagnetic calorimeter. CMS Conference Report 2006/067, 2006. http://cms.cern.ch/iCMS/jsp/openfile.jsp?type=CR&year=2006&files=CR2006_067.pdf.

List of Figures

1.1 Livingston plot showing colliders only (Source: [49]) 10
1.2 Schematic layout of the LHC (Source: [15]) 12
1.3 LHC injector chain (Source: [14]) 12
1.4 Beam structure of the LHC (Source: [59]) 12
1.5 Schematic layout of the LHC arc cells. MBA and MBB denote the type A and B main dipole assemblies (Source: [15]) 12
1.6 Cross sections and event rates as a function of center-of-mass energy (Source: [34]) 14
1.7 Charged particles (hadron and muon) fluxes in the CMS detector and experimental area. Data from Fluka simulation for $\mathcal{L} = 10^{34}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$(Source: [23]) . 16
1.8 Absorbed dose (in Gy) in the CMS detector. The values correspond to an integrated luminosity of $5 \times 10^5\,\mathrm{pb}^{-1}$, as expected to be accumulated during the first ten years of LHC operation. (Source: [23]) 16
1.9 Feynman diagrams for main Higgs production channels at the LHC (Source: [54]) 18
1.10 Calculated (leading order) cross sections for the SM Higgs boson as a function of m_H (Source: [9]) 18
1.11 Calculated branching ratios for the SM Higgs boson as a function of m_H (Source: [9]) 18
1.12 Reconstructed di-photon invariant mass distribution of the $\mathrm{H} \rightarrow \gamma\gamma$ signal (dark) and the background (light) with $\mathrm{m_H} = 130\mathrm{GeV/c^2}$ for $100\mathrm{fb}^{-1}$ (Source: [9]) . . 18
1.13 Expected statistical significance with 30 fb^{-1} for the SM Higgs boson as a function of $\mathrm{m_H}$ (Source: [9]) 20
1.14 The 5σ-discovery potential for the lighter scalar MSSM Higgs boson as a function of $\mathrm{m_A}$ and $\tan\beta$ for $30\mathrm{fb}^{-1}$ with $\mathrm{A_t} = 2450\mathrm{GeV/c^2}$ (Source: [9]) 20

2.1 Schematic elevation view of CMS 24
2.2 Schematic view of the CMS magnet inside the cryostat assembly 24
2.3 Schematic elevation view of the CMS tracking system 26
2.4 Projected view of one quarter of the CMS tracking system (without vertex pixel detector). Red lines: single-sided modules, blue lines: double-sided modules. Angle coordinate is pseudorapidity; x and y in mm 26

2.5 Performances of the CMS tracker reconstruction of charged particles. Left: track reconstruction efficiencies as a function of pseudo-rapidity $|\eta|$ for muons and pions of different tranverse momentum p_T. Center: track transverse momentum resolutions for muons as a function of $|\eta|$ and for different p_T values. Right: track impact parameter resolutions for muons as a function of $|\eta|$ and for different p_T values. (Source: [10]) . . . 26
2.6 Layout of the CMS ECAL . . . 28
2.7 Left: Distribution of energy reconstructed in a 3×3 crystal matrix for electrons at 120 GeV. Right: Resolution as a function of the energy (Source: [63]) . . . 28
2.8 Cross section in transverse plane of one 20° wedge of the CMS HCAL . . . 28
2.9 A 150 GeV Higgs event decaying into four muons in the CMS detector (Source: [23]) . . . 30
2.10 Side view of one quarter of CMS, showing positions of the muon systems (Source: [54]) . . . 30
2.11 Cross section of a drift cell with electric field lines, without magnetic field (Source: [23]) . . . 32
2.12 Cross section in the $r - \phi$ plane of a drift tube chamber inside the iron yoke (Source: [23]) . . . 32
2.13 Drift tube superlayer during assembly. Anode wires can be seen between the aluminium cathodes (Source: CERN CDS) . . . 32
2.14 Principles of MWPC and CSC . . . 34
2.15 Schematic drawings of the CMS Cathode Strip Chambers design (Source: [23]) 34
2.16 CMS endcap wheel YE2 fully equipped with CSCs (Source: CERN CDS) . . . 34
2.17 Principles of RPCs. Quoting [42]: "a) Some gas atoms are ionized by the passage of a charged particle. An avalanche is started. b) The avalanche size is sufficiently large to influence the electric field in the gas gap. c) The electrons reach the anode. The ions drift much slower. d) The ions reach the cathode. The charges in the resistive layers influence the field in a small area around the position where the avalanche developed" (Source: [42]) . . . 36
2.18 A cloud chamber photograph of electron avalanches in a parallel plate counter (Source: [42]) . . . 36
2.19 RPC efficiency as measured in a test beam (Source: [11]) . . . 36
2.20 Fisheye view of the CMS experimental cavern during the lowering of the central barrel wheel YB0 on Feb. 28, 2006. YB+1 and the HB+ HCAL supermodule can be seen on the left [Source: CERN CDS] . . . 38

3.1 CMS trigger and data acquisition architecture (Source: [28]) . . . 42
3.2 CMS Level 1 Trigger architecture and sub-units (Source: [27]) . . . 42
3.3 Latency loop detector front-end – L1 Trigger – detector . . . 42
3.4 Cross section of the CMS experimental cavern UXC55 and the neighboring counting room USC55. Trigger data is transmitted through two direct but inclined tunnels (to absorb radiation) between the caverns to minimize cable length (Source: [27]) . . . 44
3.5 The CMS DAQ and event building system . . . 44
3.6 'Three-dimensional' view of the full Event Builder system (Source: [28]) . . . 44
3.7 Various on-detector buffer and readout architectures (Source: [60]) . . . 46
3.8 sTTS and aTTS integration with major data flow components (Source: [28] . . 46

3.9 Layout of the LHC Computing Grid Tier-0 Network (Source: [40]) 48
3.10 Sample of LCG node activity and data transfers (Source: screenshot of monALISA monitoring application http://monalisa.caltech.edu/ 48
3.11 Calorimeter trigger tower layout in the ECAL endcap (Source: [27]) 50
3.12 RCT electron/photon trigger algorithm (Source: [27]) 50
3.13 A full RCT crate without covers, front and rear view 50
3.14 Muon trigger data flow (Source: [27]) 52
3.15 DT local trigger BTI wire groups (Source: [16]) 52
3.16 Overview of the DT Local Trigger: functions of BTIs, TRACOs and TS (Source: [27]) 52
3.17 Logical segmentation and block diagram of the DT Track-Finder system (Source: [27]) 54
3.18 DT Track Finder Processor block diagram (Source: [27]) 54
3.19 Principle of the DT Track Finder algorithm (Source: [27]) 54
3.20 The data mapping from CSC chamber-level LCT information into 60° sector information for the CSC Track Finder (Source: [27]) 56
3.21 Overview of the CSC Trigger system(Source: [54]) 58
3.22 Block diagram of the Sector Processor logic (Source: [27]) 58
3.23 Overlapping cones of input strips to the RPC trigger PAC ASICs and principle of pattern-based track reconstruction (Source: [27]) 60
3.24 RPC trigger tower boundaries in the (r, ϕ) plane (Source: [27]) 60
3.25 Schematic overview of the Global Trigger crates 62
3.26 Illustrations of trigger conditions (Source: [62]) 64
3.27 The Global Trigger Crate with all prototype modules. From left to right: PCI-VME64x bridge and auxiliary modules (6U, left top). L1A Out – TCS – PSB – FDL – GTL – 3 × PSB – TIM – GTFE – GMT – 3 × PSB hidden behind GMT front panel 64

4.1 Block diagram of the GMT (Source: [54]) 68
4.2 Match and Ranking logic units (Source: [54]) 70
4.3 Overlap Region DT/CSC Cancel-Out Unit, inside the GMT Barrel Logic FPGA (Source: [54]) 70
4.4 Parameter projection logic (Source: [54]) 70
4.5 Trigger rates at low luminosity ($\mathcal{L} = 2 \times 10^{33}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$) as a function of trigger threshold for three types of GMT merging (see text). Obtained using a minimum bias dataset. Updated ORCA 6.2.3 simulation (Source: [54]) 72
4.6 Efficiency for triggering on muons from $\mathrm{W} \rightarrow \mu\nu$ decays as a function of pseudorapidity with a p_T threshold of 14 GeV/c for three types of GMT merging (see text). Updated ORCA 6.2.3 simulation. (Source: [54]) 72
4.7 Efficiency of the GMT and regional muon triggers as a function of pseudorapidity η for a sample of single muons with flat p_T distribution. No p_T threshold applied. GMT candidates that are suppressed in the single-muon trigger are excluded. Updated ORCA 6.2.3 simulation. (Source: [54]) 74
4.8 Ghosting probability as a function of η in the GMT and regional muon triggers. No p_T threshold is applied. Candidates that are suppressed in a di-muon trigger are excluded. Updated ORCA 6.2.3 simulation (Source: [54]) 74
4.9 Schematic view of the GMT board (Source: [54]) 76

4.10 Simplified schematic view of the Forward Logic FPGA (LFF) (Source: [54]) . . 76
4.11 The Global Muon Trigger 78

5.1 Overview of the GMT online software (courtesy H. Sakulin) 82
5.2 Schematic setup for the GMT-DTTF interconnection test (courtesy I. Mikulec) 84
5.3 Schematic setup for the GMT-CSC interconnection test (courtesy I. Mikulec) . 86
5.4 The CSC-GMT test setup in the Building 904 Trigger Integration Center (courtesy I. Mikulec) 88

6.1 Communication over multiple networks through peer-transports (Source: [27]) . 92
6.2 Architecture of the Trigger Supervisor (Source: [45]) 94
6.3 UML diagram of the components of a TS Cell (Source: [37]) 96
6.4 Screenshot of the AJAX-based TS graphical user interface (Source: [44]) 98
6.5 Finite state machine used for trigger configuration (Source: [45]) 100
6.6 Database-driven configuration strategy for the TS (Source: [45]) 102
6.7 Trigger Supervisor logo 104
6.8 CMS logo 104
6.9 GMT configuration database layout 106
6.10 Setup for the first TS-driven interconnection test (Source: [43]) 108

List of Tables

1.1 Overview of LHC nominal parameters (Source: LHC Design Report [15]) . . . 13
1.2 Average particle multiplicities (as used for radiation environment simulation in [20]) . . . 17
1.3 Approximate event rates of some physics processes at the LHC for a luminosity of $\mathcal{L} = 2 \times 10^{33}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$. One year is equivalent to an integrated luminosity of 20 fb^{-1}. (Source: [30]) . . . 19

2.1 Parameters of the CMS superconducting solenoid. (Source: [29]) . . . 25
2.2 Masses of the major parts of CMS . . . 27
2.3 Comparison of properties of various crystals (Source: [21]) . . . 31
2.4 CMS muon chamber properties and statistics (Source: [23]) . . . 33

4.1 Default programming of the Cancel Decision Logic for the barrel part of the GMT (Source: [54]) . . . 69
4.2 Default programming of the Cancel Decision Logic for the endcap part of the GMT (Source: [54]) . . . 69
4.3 Default programming of the CSC/barrelRPC Cancel Decision Logic (in the barrel processing chain of the GMT) (Source: [54]) . . . 71
4.4 Default programming of the DT/forwardRPC Cancel Decision Logic (in the forward processing chain of the GMT) (Source: [54]) . . . 71

Acknowledgements

I would like to thank Univ.-Doz. Dr. Claudia-Elisabeth Wulz for supervising this thesis in such an open and encouraging way. It was her lecture on "Perspectives in High Energy Physics" that re-evoked my interest in particle physics – which in fact dates back to my primary school days, when I learned about the LEP detectors and the Quark model in issue 79 of the famous "Was ist Was" series of children's books. Dr. Wulz provided me with the unique opportunity to witness the construction of a new generation of accelerator and detectors during the very phase when they take form.

I greatly owe to the staff of the Institute for High Energy Physics (both in Vienna and at CERN) for sharing their great professional and human experience: Toni Taurok gave me a deep insight into the design of large electronics systems; with Ivan Mikulec I spent many concentrated weeks at the electronics integration center in Bat. 904, learning both about the physical and sociological aspects of CERN. Ildefons Magrans de Abril introduced me to distributed systems development, and Hannes Sakulin's professional C++ and VHDL code and PhD thesis made my integration task an especially instructive one. I would also like to thank Manfred Jeitler for many inspiring and witty discussions.

I will never forget the great time I spent in our shared Geneva flat with my dear friend Philipp Glaser. Without him I probably wouldn't have taken the step to go there. His incredibly life-loving, open and honest personality can be found just as rarely as a golden Higgs boson event!

Finally I would like to thank my friends, my sister Carolina and my parents for being there.

Vienna, October 8th, 2007