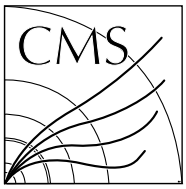
Available on CMS information server

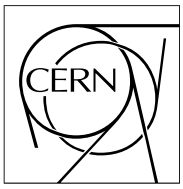
**CMS NOTE 2002/033**



**The Compact Muon Solenoid Experiment**

**CMS Note**

Mailing address: CMS CERN, CH-1211 GENEVA 23, Switzerland



# CMS L1 Trigger Control System

CMS Trigger/DAQ Group

Editor: J. Varela

Abstract

The L1 Trigger Control System (TCS) is responsible to control the delivery of L1 Trigger Accepts depending on the status of the sub-detectors readout systems and the data acquisition system. This status is derived from information provided through the Trigger Throttling System (TTS) and from the status of front-end Emulators. TCS is also responsible for generating synchronization and fast reset commands, as well as to control the delivery of test and calibration triggers. TCS uses the TTC network to distribute information to the subsystems. TCS partitioning permits groups of subdetectors main components to operate independently during setting-up, test or calibration phases. Local trigger control is foreseen for the subdetector operation in standalone mode (test beam mode)

This document provides an overall description of the TCS requirements and architecture, and a detailed description of the TCS components. The main TCS components are the Central Trigger Controller (TCS), the Local Trigger Controller (LTC), the TTCci module (CMS version of TTCvi) and the sTTS Fast Merging Module (FMM).

# Contents

# 1. Requirements

The main task of the L1 Trigger Control System (TCS) is to control the delivery of L1 Trigger Accepts (L1A) generated by the Global Trigger, depending on the status of the readout electronics and the data acquisition. This status is derived from local state machines that emulate the front-end buffers occupation, as well as from direct information transmitted back by the CMS subsystems through the Trigger Throttling System (TTS). In addition, TCS will require that the sequence of L1A complies with precise trigger rules designed to reduce the probability of buffer overflows. TCS is also responsible for generating synchronization and fast reset commands, as well as to control the delivery of test and calibration triggers. TCS uses the TTC network to distribute information to the subsystems.

TCS partitioning permits groups of subdetectors or subdetector main components to operate independently during setting-up, test or calibration phases.

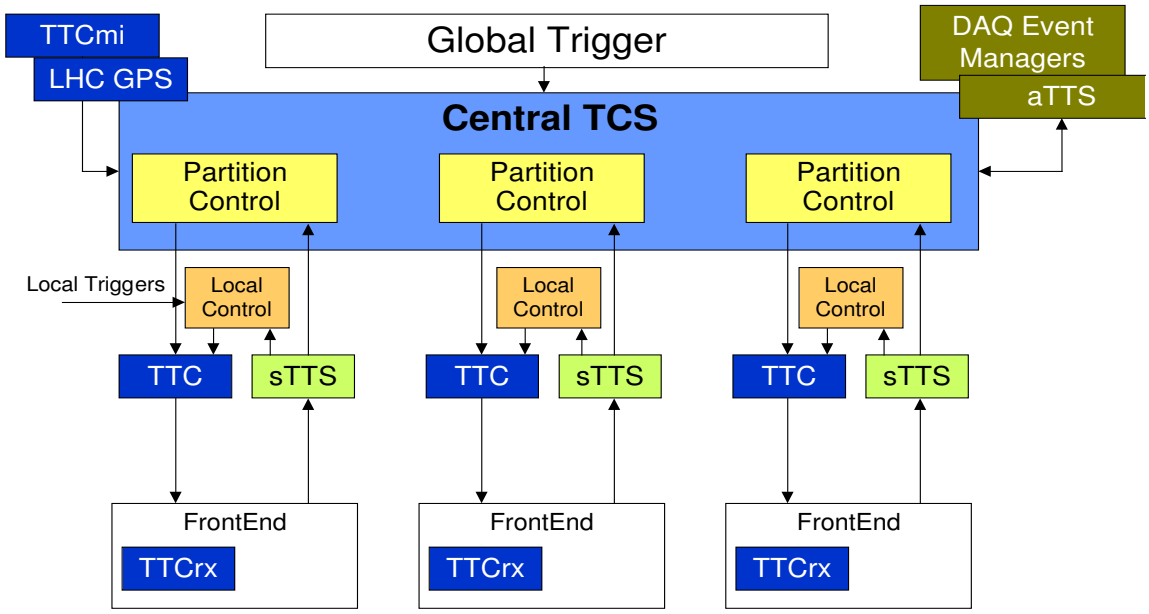



**Fig. 1:** Overview of the Trigger Control System.

## 1.1 Requirements on Partitioning

TCS is organized in partitions (Figure 1), each one corresponding to a major component of a subdetector, to which it distributes L1A and other TTC commands and from which it collects TTS signals. Software configuration allows the partitions to be grouped in partition-groups, each one matching a DAQ partition.

Partitioning requires that the TTC and TTS trees are organized in branches (partitions), with intelligence at the top allowing independent operation of partition-groups. TCS provides this intelligence being able to control the delivery of L1A and other fast commands to independent partitions-groups, and to handle the TTS fast feedback received from independent partitions-groups.

Partitions from different subdetectors may be combined in one partition-group. All partitions are integrated in a single group when in normal experiment operation.

At the central level, the configuration and subsequent operation of partition-groups is done by the central Trigger Controller located in the Global Trigger crate. Each partition-group may operate with an independent physics trigger. The partition-group is able to define its own trigger conditions, receiving L1As that correspond to that conditions. L1A is distributed to just one partition-group at a time.

At the subdetector level, control and clocking may be taken optionally by a local trigger controller. In this case local triggers and clocks are used in place of the global triggers.

## 1.2 Requirements on Front-End Event Identifier

The subdetector event fragments delivered to the Data Acquisition are identified with a standard Front-End Event Identifier. This identifier contains the Event Number, the Bunch Number and the Orbit Number.

The Event Number counts the number of L1A received since the last Event Counter Reset (TTC command). The Event Number has 24 bits and completes a new cycle every 168 seconds (at the maximum 100 kHz L1A rate).

The Bunch Number counts the number of 40 MHz clocks within one LHC orbit. The Bunch Number has 12 bits allowing for the 3564 clocks within one orbit. The Bunch Number is reset by the Bunch Crossing Zero (BC0) TTC command distributed once per orbit. By definition, the first bunch after the main LHC gap is bunch number 1.

The Orbit Number has 32 bits and counts the orbit number in the run since the last Orbit Counter Reset distributed by the TTC system. Normally, the OCR command is distributed at the start of a new run.

The GPS absolute time of the first bunch in the orbit is obtained from the interface to LHC Beam Synchronous Timing system (BST) and logged by TCS.

## 1.3 Requirements on L1A Control

TCS is responsible to guarantee that the subsystems are ready to receive every L1A delivered. This functionality is essential to prevent buffers overflows and/or trigger signals missed when the subsystems are not ready to receive them.

### FrontEnd Emulators

To prevent overflow of the front-end de-randomizer buffers, TCS houses Emulators of the most critical subdetector de-randomizers (tracker and preshower). The de-randomizers store a fixed number of words per L1A and for some subdetectors all behave identically. For these cases, the buffers occupancy depends only on the L1A rate and on the readout throughput. A state machine receiving the L1A signals can emulate the de-randomizer behavior and determine its occupancy at each new L1A. If a new L1A is estimated to cause a de-randomizer overflow, this L1A is not delivered. In case a L1A is not delivered it is nevertheless counted in the Deadtime Monitor (see below). The Emulators are provided by the subdetector groups in the form of 6U VME modules to be installed in the central trigger rack.

### Trigger Rules

The delivery of L1A signals will comply with a set of Trigger Rules. These rules take the general form ’no more than n L1A signals in a given time interval’. Suitable rules, inducing a dead time less than 1%, minimize the buffer overflow probability.

The first trigger rule is imposed by the tracker and the preshower that there must be two untriggered crossings between consecutive L1As, yielding a minimum separation of 75 nsec.

### Synchronous TTS (sTTS)

TCS receives warning signals from the CMS subsystems through sTTS indicating that some of the buffers are almost full. These signals are used to prevent overflows in the readout buffers of the data acquisition chain.

The occupancy of these buffers depends on the event fragment sizes after data suppression (zero suppression or selective readout) and in consequence can not be emulated in a central place. The strategy to avoid overflows is based on sTTS according to the following scheme:

- locally, the occupation of buffers is monitored, detecting when filled above a programmable warning level;
- at this point a sTTS warning signal is sent to TCS, which inhibits or reduces the L1A temporarily;

– before the sTTS feedback loop is closed L1A may still arrive, in which case 'empty events' are stored (a small data block containing the event identification and a buffer overflow error flag);

– the storage of complete events is resumed when the buffer occupation gets below a programmable storage level (the storage level is set below the warning level to avoid unstable behavior of the sTTS warning signal).

The readout buffers size must be such that the probability of reaching the warning level times the number of buffers is negligible and doesn't contribute significantly to the data acquisition inefficiency (less than 1%).

### Deadtime Monitor

TCS contains a Deadtime Monitor that tracks the disposition of each active beam crossing, i.e. whether accepted, rejected, and whether lost due to downtime of trigger and/or DAQ. Active beam crossings in the LHC orbit are defined as the crossings potentially with particles (i.e. all crossing except the gaps in the orbit)

## 1.4 Requirements on Synchronization Control

In addition to the L1A signal, TCS must send to the subsystems a number of other fast signals for synchronization purposes.

### Bunch Crossing Zero

TCS sends to the subsystems a TTC command (called BC0) synchronous with the end LHC orbit. The BC0-command is sent every orbit to all TTCci boards and the TTCci's in turn send a BC0 over the TTC B-channel to the trigger and readout electronics.

There is only one BC0 Generator, located centrally in the TCS board, to force the same time relation between all subsystems. The TCS sends the BC0-command at the same time to all TTCci's who have to delay it to assign the bunch crossing numbers correctly in the respective partitions.

The central TCS sends BC0 at BC=0, defined as the clock cycle synchronous with L1A signal from bunch zero in TCS (see Section 3.12).

During a Reset/Resync procedure the TCS and the TTCci's do not send BC0-commands (see Section 3.6).

The Bunch Crossing Zero command (BC0) is used by the trigger subsystems to flag the bunch zero data flowing in the trigger system and is used by the readout subsystems to reset the bunch counters.

The trigger systems should not stop sending trigger data unless in some specific resynchronization, configuration or diagnostic mode. Otherwise, the trigger dataflow should not stop.

### Event and Orbit Counters Reset

TCS is able to send TTC commands that reset the event and orbit counters.

### Start/Stop

TCS should be able to send Start/Stop commands to the sub-systems. These commands are recognized by the trigger subsystems which start/stop sending trigger data synchronously at the next LHC orbit, on reception of BC0.

## 1.5 Requirements on sTTS

The subdetectors send to TCS using the sTTS feedback tree, the following feedback status: Ready, Busy, Overflow Warning, Out of Sync, Error and Disconnected. TCS should be able to receive these status information per subdetector partition. The TTS states are mutually exclusive.

When active the status signals have the following meaning:

a) *Ready:* the partition is ready to receive triggers.The Ready status is applied continuously to know that the system is connected and working. The Ready status has to be different from the signal received when cables are unconnected or the electronics switched-off.
b) *Busy:* the partition is temporarly busy preparing itself to take data and can't yet receive triggers. The subsystem cannot receive triggers but will eventually be able to receive them without an external intervention (i.e signal).
c) *Overflow Warning:* the partition buffers are close to overflow.
d) *Out of Sync:* event fragments collected in the partition don't correspond to the same front-end pipeline position or have different Event IDs or/and synchronous trigger data is out of sync. The system will not be able to recover synchronization without an external intervention (i.e resync signal).
e) *Error:* the system is in error state and needs a reset. The system will not be able to receive triggers without an external intervention (i.e reset signal).
f) *Disconnected:* setup not done, cables removed, or power-off.

The sTTS has programmable logic to determine when to issue the signals (e.g. more than n modules in error state implies that the Error signal is send to TCS). The subdetectors send the sTTS signals as fast as possible. The collection of sTTS signals and the decision to send a signals to TCS is done by hardware.

## 1.6 Requirements on Calibration and Test Triggers

Calibration and test triggers in CMS can be delivered in several different contexts:

1. Subdetectors in standalone mode: subdetectors generate test and calibration sequences locally and capture the data with the subdetector local DAQ.
2. Subdetectors in DAQ partition mode: TCS generates test and calibration triggers and the data is collected by one of 8 central DAQ partitions (see Section 2.2).
3. Periodic test or calibration triggers issued by TCS during a Physics Run: the triggers are issued centrally and all subsystems deliver an event data block in order to keep the event synchronization (the event block can be empty if the subdetector doesn't require test data).
4. Local test or calibration triggers issued by the subsystems during a Physics Run: the subsystems perform test, calibration or monitoring activities during Private Gaps or Private Orbits defined by TCS.

Calibration triggers issued centrally by TCS follow a well defined protocol. With a fixed delay before the calibration trigger, TCS issues a TTC command that is used by the subsystems to prepare the calibration signal (e.g. laser pulse generation). It is the responsibility of the subsystems to synchronize the test/calibration pulses with the test trigger. Test triggers are defined as simple triggers, without any corresponding TTC command.

## 1.7 Requirements on Subsystems Reset

A number of circumstances during data taking, in particular synchronization errors due to single event upsets in the frontend pipelines, due to buffer overflows or due to errors in signals transmission, will require a reset in order to recover normal functioning. The standard recover procedure involving a new run start may translate into an unacceptably high data acquisition inefficiency.

For this reason TCS has the possibility to distribute reset TTC-commands to the subsystems. The reset command defines a time interval without triggers that can be used by the subsystems to reset event counters and readout buffers or to partially reset their electronics. These commands can be issued on a periodic basis, independently of the status of the readout electronics, or in response to a loss of synchronization identified in some subsystem.

Two fast reset commands are foreseen. The Resync command should be interpreted as a re-synchronization of all subsystems to the same event identifier and may involve the reset of readout buffers. The Hard Reset command is intended to recover from electronics malfunctioning, allowing in particular fast reconfiguration of frontend FPGAs. None of these resets should imply reloading of parameters under software control.

Local resets applied locally are possible in the following conditions:

1. Special resets of the front-end electronics can be initiated by the sub-systems during the main gap provided no events are lost and the sub-system preserves the ability to respond to central TTC commands.

2. The front-end electronics reset should not affect the TTCrx chips, and in particular the local Event Number and Bunch Number.

# 2. Architecture

## 2.1 Overview

An overview of the TCS architecture is given in Figure 2. The main TCS components are:

- Central Trigger Controller
- Local Trigger Controller
- TTCci Module (CMS version of TTCvi)
- sTTS Fast Merging Module

Control is taken by the central Trigger Controller when running in DAQ-Partition mode. In Standalone mode, control is taken by the local trigger control module. The switching between the two modes is performed at the TTCci input, which receives signals from both controllers. An overview of the functions performed by each component is given in this chapter. Individual chapters are dedicated to the detailed description of each component. The TCS interfaces to external systems are described below.

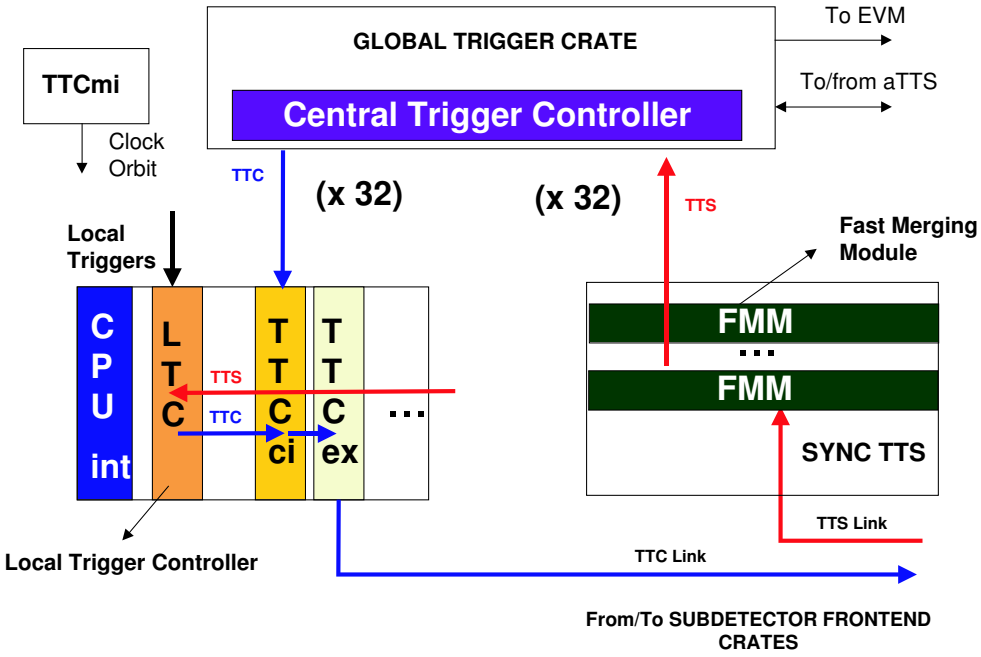



**Fig. 2:** Trigger Control Architecture.

### Interface to Global Trigger

The Global Trigger has the ability to define 128 programmable Trigger Conditions that are based on the trigger objects (electrons/photons, jets, muons, $E_T$ sums) computed by the trigger system sub-components. It performs 8 final ORs of selected sub-sets of the trigger conditions.

The central Trigger Controller receives from the Global Trigger the results of the 8 final L1As to be distributed to the various partitions-groups. It also receives the 128 trigger conditions bits to be sent to the DAQ Event Manager.

The central Trigger Controller is located in the Global Trigger Crate and backplane dedicated lines are used for this interface.

## LHC Clock Interface

The LHC Clock and Orbit signals are distributed from the Prevessin Control Room to the LHC experiments through singlemode optical fibers. At the experiment Counting Room, the Clock and Orbit signals are recovered by circuitry (LHCrx module) in the TTC Machine Interface (TTCmi) crate. At this level the clock jitter is expected to be of the order of 10 ps rms [7]. Fanout modules (TTCcf) in the TTCmi crate are used to distribute the Clock and Orbit signals to nearby TCS modules:

- Central Trigger Controller, TIM-module in GT crate;
- Local Trigger Controllers and TTCci/TTCex modules.

## Interface to BST

The LHC Beam Synchronous Timing System (BST) for LHC beam instrumentation serves to synchronize acquisitions in areas that are very distant geographically, and is also used to convey signals, parameters and commands simultaneously to all instruments around the machine. All necessary real-time information is regrouped and transmitted in a so-called BST message.

A real-time process collects all the data and commands to be transmitted via a dedicated TTC system on a given turn (not to be confused with the experiment TTC system). These data and commands are then assembled into a message. At each orbit turn clock period, the BST Master transmits the message for the current turn over channel B of the dedicated TTC system.

Each message consists of 32 bytes, of which 8 bytes codify the GPS Absolute Time. It contains also the beam type and energy, the mean current per bunch and the number of injected bunches [see ref. 9 for detailed format].

A TTCrx on the central and local trigger controllers, connected to the BST system, allows access to this information, in particular to GPS Absolute Time.

## Interface to DAQ Event Manager

The central Trigger Controller sends to the Event Manager (EVM) all 128 Algorithm bits and a word indicating the partitions-group that has received the L1A signal as well as the type of trigger (Table 1, Table 2 and Table 3).

The Event Manager will be located in the surface Control Room, whereas the TCS modules will be placed in the underground Counting Room. The communication between TCS and the EVMs will use a standard frontend readout link (FRL). The L1A-record (Table 1) complies with the FED Common Data Format [8].

**Table 1:** L1A-record payload from TCS to EVM, organized in 32-bit words. The Trigger Number and the Bunch Crossing Number are part of the header and comply to the FED Common Data Format. When necessary, leading zeros fill the MSB bits. Two additional trailer 32-bit words (not shown), defined in the FED Common Data Format, are included. A CRC code is included in the trailer.

| Content | Comment | Number of bits |
|---|---|---|
| Trigger number | Counts all L1A | 24 |
| Bunch crossing number | BC counter: max. number =3564 | 12 |
| Orbit number | Orbit counter for 12 h | 32 |
| Event number | Counts all L1A for this DAQ partition | 24 |
| DAQ#, Trigger Type | 3 bit DAQ partition nr+ 4 bit trigger type | 7 |
| Algo bits_3 | bits 127...96 | 32 |
| Algo bits_2 | bits 95...64 | 32 |

**Table 1:** L1A-record payload from TCS to EVM, organized in 32-bit words. The Trigger Number and the Bunch Crossing Number are part of the header and comply to the FED Common Data Format. When necessary, leading zeros fill the MSB bits. Two additional trailer 32-bit words (not shown), defined in the FED Common Data Format, are included. A CRC code is included in the trailer.

| Content | Comment | Number of bits |
|---|---|---|
| Algo bits_1 | bits 63...32 | 32 |
| Algo bits_0 | bits 31...0 | 32 |
| Technical Trigger bits | bits 31...0 | 32 |
| GPS Absolute Time | Bytes 1 to 4 of BST GPS time | 32 |
| GPS Absolute Time | Bytes 5 to 8 of BST GPS time | 32 |

**Table 2:** Trigger Types

| Type | Name | Comment |
|---|---|---|
| 0001 | Physics trigger | Final OR (physics algorithms) |
| 0010 | Calibration trigger | Calibration cycle |
| 0011 | Test trigger | Test cycle |
| 0100 | Technical trigger | Technical trigger (external trigger) |
| 0101 | Simulated events | Reserved for DAQ usage |
| 0110 | Traced events | Reserved for DAQ usage |
| 1111 | Error | |
| others | undefined | |

**Table 3:** Format of 'DAQ#, Trigger Type' word

| 31, 30 .....8, 7 | 6, 5, 4 | 3, 2, 1, 0 |
|---|---|---|
| 0.....0 | DAQ partition number | Trigger Type |

## 2.2 Partitioning

### Subdetector TTC/TTS partitions

The number of TTC/TTS partitions is limited to 32. Table 4 gives a list of the subdetector TTC/TTS partitions. These partitions are hardwired in TTC/TTS trees.

To each TTC partition corresponds a top TTCci module (see Section 5.) interfacing to central TCS and local LTC, from where it receives commands. The TTCci connects to a TTCex module, which distributes encoded commands through a passive optical splitting tree to the receiving TTCrx chips in the front-end electronics.

To each TTS partition corresponds a top FMM module (see Section 6.) interfacing to central TCS and local LTC, to where it sends sTTS feedback signals. The top FMM module receive sTTS signals from other merging FMMs in the sTTS tree or directly from front-end modules.

TTC and TTS partitions match each other.

**Table 4:** List of subdetector TTC/TTS partitions

| Subdetector | Number of Partitions | Partitions |
|---|---|---|
| Pixels | 2 | Barrel, Forward |
| Si-Tracker | 4 | Disk+, Disk-, Inner barrel, Outer barrel |
| ECAL | 6 | EB+, EB-, EE+, EE-, SE+, SE- |
| HCAL | 6 | HB+, HB-, HE+, HE-, HF, HO |
| RPC | 4 | Endcap+, Endcap-, Barrel+, Barrel- |
| DT | 2 | Barrel+, Barrel- |
| CSC | 2 | Endcap+, Endcap- |
| Calorimeter Trigger | 1 | Calorimeter trigger |
| Muon Trigger | 2 | CSC and DT trigger |
| Global Trigger | 1 | Global trigger + Global Muon trigger |

## Partition modes

Partitioning into subsystems may be accomplished in various ways. The principal method (DAQ Partition Mode) is based on the central Trigger Controller, which is configured to handle the required partition-groups. L1A signals provided by the Global Trigger are available to the partition-groups. TCS partition-groups match the central DAQ-partitions, if central data acquisition is required. Optionally, partitions may use Local Readout. In this case, TCS sends the physics and calibration triggers to the partition, but not to the DAQ Event Manager.

A second method (Standalone Mode) is used only at subdetector level, and is based on the Local Trigger Controller (LTC) module. This method has a restricted use since it is only employed with local triggers and local DAQ readout. Test beam setups are based on this configuration.

In summary, TCS allows the following possibilities in DAQ Partition mode:

i) TCS provides all functions necessary to run partitions or group of partitions in parallel for calibration, tests and commissioning with beam.

ii) The partitions-groups (DAQ partition) are configured by the Run Control both in the TCS and in the DAQ Event Managers. The TCS count the Event Number per partition-group. The DAQ system combines data from the subsystems of a partition-group to build events.

iii) Local readout of events triggered by central L1As is possible: a partition may want L1A to read events by the local DAQ and does not require central DAQ resources. The TCS sends L1A to the partition but not to the EVM.

Due to L1 latency limitations, the number of trigger condition OR' s in the Final Decision Logic is limited to 8. That means that up to 8 partition groups can run physics triggers simultaneously.

Each TTC/TTS partition can be included in one partition-group only.

## DAQ partitioning

The event fragment assignment to a DAQ slice by the Frontend Readout Links (FRL) can be programmed in two modes corresponding to the two main ways of implementing DAQ partitions in CMS.

Mode 1) FED-RU builder partition. The FRL logic sends the event fragments to a DAQ slice according to a common switching rule using for example the trigger number. All active DAQ slices result partitioned in the same way.

Mode 2) FED-DAQ slice partition. Each set of FED-FRLs associated to a given partition is programmed to send the event fragments always to the same DAQ slice.

In both cases, the receiving EVM is programmed to broadcast read commands only to the RUs subset corresponding to the relevant partition number.

### L1 Trigger partitioning

The Level 1 Trigger presents a special case in partitioning. Conceived as a subsystem of its own, alongside the various front-end DAQ subsystems, it nonetheless has components within it that are closely associated with each of the DAQ subsystems. For some types of testing, the detector partitions and their associated Level 1 Trigger partitions may be grouped in the same partition-group.

The Global Trigger partition can not belong to more than one partition-group, like all the other partitions. As a consequence, only one of the partition-groups running at a given time has access to the detailed global trigger data transmitted to DAQ. However, the event data assembled by DAQ for each partition-group include the global trigger data blocklet sent to the DAQ Event Manager. This data blocklet contains the 128-bit trigger conditions, providing informations on which triggers were active for a given event.

## 2.3 TTC system

The Trigger Timing and Control (TTC) system provides for distribution of trigger and fast control signals using two time division multiplexed channels transmitted over an optical passive distribution tree. The channel A is used to transmit the L1A signal whereas channel B transmits other fast commands. Both channels have a bandwidth of 40 Mbit/s.

### TTCci and TTCex

The top elements of each subdetector TTC partition are the TTC CMS Interface (TTCci) and TTC Encoder and Transmitter (TTCex) modules. The TTCci receives L1A and commands from the central Trigger Controller or from the local Trigger Controller, switching between the two sources as a function of the operation mode (DAQ Partition mode or Standalone mode).

The TTCex module encodes both channels A and B from the TTCci and has a transmitter laser with sufficient power to drive the optical splitters. The clock from the TTCmi crate is plugged directly into the TTCex modules. At reception, optical receivers with a pin diode and an amplifier generates an electrical signal at the TTCrx chip input.

### TTCrx

The TTC signals are received by the TTC Receiver (TTCrx) chip which provides as its output the 40 MHz LHC clock, both raw and deskewed, the Level 1 Accept (L1A) trigger and command data. Deskewing is provided for the clock, the L1A and the broadcast commands. The coarse deskew is provided in 16 steps of 25 ns each and the fine deskew is provided in 240 steps of 104 ps each. The TTCrx has a hardwired ID at startup, Boundary Scan, an I2C interface with all registers read/write, and 64 possible user broadcast commands.

## 2.4 TTS system

### Synchronous TTS (sTTS)

The sTTS is a tree-like structure with point-to-point connections and hardware state machines combining the status of individual components, which provides TCS with the status of the subdetector partitions.

The central TCS receives 4 hardware LVDS signals from the 32 subdetector partitions in the sTTS merging tree, and directly from the DAQ Asynchronous TTS (aTTS). The four hardware signals are named T3 (Ready), T2 (Busy), T1 (Out of Sync) and T0 (Overflow Warning). These signals are used to encode the states Ready, Busy, Overflow Warning, Out_of_Sync, Error and Disconnected (see Section 3.6).

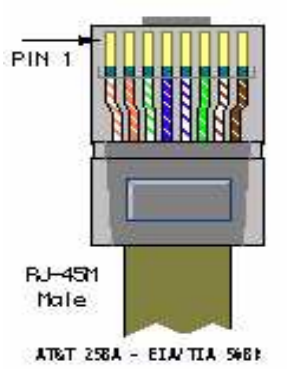
The local Trigger Controllers receives also the sTTS signals corresponding to the partitions of the respective subdetector.

The sTTS tree will transport and combine status of individual subdetector components. The partitions status is derived from the state of individual components and is updated at 40 MHz using hardware state machines. The Fast Merging Module described in Section 6. is used to build the sTTS tree. It accepts signals from 32 sources and produces a combined status at the output, according to a programmable algorithm (logical OR or majority algorithm).

The signals that originate reset procedures (Out of Sync and Error) are also collected by the software based asynchronous TTS (aTTS), allowing if necessary for more sophisticated processor based algorithms to decide when and which resets to apply.

The sTTS signals are electrical LVDS. The input/output of sTTS signals use RJ45 connectors with the following pin assignment:

- pin 1 = - T2 (BUSY)
- pin 2 = + T2 (BUSY)
- pin 3 = - T3 (READY)
- pin 4 = + T0 (OVERFLOW_WARNING)
- pin 5 = - T0 (OVERFLOW_WARNING)
- pin 6 = + T3 (READY)
- pin 7 = - T1 (OUT_OF_SYNC)
- pin 8 = + T1 (OUT_OF_SYNC)




## Interfaces TCS, sTTS and aTTS

Signals of a TTS channel: Ready, Busy, Out of Sync and Overflow Warning.

Hardware interfaces:

- sTTS to central TCS: 32 TTS channels, one per TTC partition.
- sTTS to local TCS: 6 TTS channels, one per TTC partition.
- aTTS to central TCS: 8 TTS channels, one per DAQ partition; provide the status of the DAQ partitions.
- central TCS to aTSS: 8 TTS channels, one per DAQ partition; provide the status of connected TTC partitions and TCS control logic.
- local TCS to aTTS: 1 TTS channel (used only in standalone mode).

Software interfaces:

a) Global trigger controller - RCS communications:

- Status from the GTP processor to Run Control (Info center).
- Initialize and configure the GTP, TCS and trigger partitions (partition-groups, Trigger type etc). Force TCS trigger partition into one of its states e.g. initialize and start a run.

b) sTTS - RCS controls (via DSN):

- Program the logic to combine the TTS signals of each partitions. Analyze (for debugging) the history of the (FED, FES) TTS signals.

## Asynchronous TTS (aTTS)

The aTTS controller collects the history of all messages coming from RUs, BUs, FUs, EVMs and RCS (and indirectly DCS). The messages contain the time, data error and warning tags or direct commands (e.g. RCS start stop). The history of all incoming messages is analyzed (by the DAQ Doctor) within a time window together with the TCS status. According to predefined tables the aTTS supervisor selects the action to execute such as: do nothing, send a specific TTS level to TCS (Ready, Busy, Warning, Out of Sync) or transfer the action control to RCS (in auto or manual mode). Usually an aTTS action results into a sequence of operations such as setting TCS in suitable state while sending the RCS type messages (reload, restart etc.) to the faulty DAQ component (e.g. RUs, EVM, entire DAQ unit or FRL system) and then resume the data taking run setting TTS level Ready.

In case of warning on the use of resources (rate too high), the following sequence of operation may be applied:

1. The DAQ doctor (DqD) verifies that the faulty status is persistent in a given time window, and if it is a global situation (coming from all DAQ slices) or if it comes only from few slices. In the first case the actions are:

– set TTS level Overflow Warning (for the given partition);

– if the Warning messages disappear in a given time window remove the Warning level.

2. In the second case the actions are:

– set TTS level Busy (for all partitions);

– program the FRL event distribution table with a pattern corresponding to acceptable loads for each DAQ slice;

– when all commands are acknowledged then remove the TTS level Busy.

In case of OutOfSynch or bad data messages from RUs or BUs, the DAQ doctor (DqD) verifies that the faulty status is persistent in a given time window and serious enough to intervene, then the DqD performs the following actions:

1. Suspend the trigger generation by setting the TTS level Busy (for the faulty partition or all if standard data taking).
2. If the recovery procedure consists in removing a DAQ slice for a while (e.g. complete reboot, or hardware replacement etc.), then:

– suppress the fault DAQ slice from all FRL tables;

– set TTS level Resync;

– when all commands are acknowledged then remove the TTS level Busy.

3. Else:

– issue the suitable RCS commands (e.g. load, init, restart etc.) to the DAQ components (EVM, RU,BU,FU) identified as the cause of error;

– set TTS level Resync;

– when all commands are acknowledged then remove the TTS level Busy.

The aTTS logic is replicated per partition, allow the independent operation of 8 DAQ partitions.

# 3. Central Trigger Control

## 3.1 Overview

The central trigger control is physically located in the Global Trigger rack. It consists of the central Trigger Controller module in the Global Trigger main 9U VME crate, the sTTS receiver modules and the front-end pipeline-derandomizer Emulator Modules. sTTS receivers and emulator modules are housed in 6U VME crates.

The functions of the central TCS are the following:

1. To control the delivery of the L1A based on the verification of Trigger Rules, on the emulation of the front-end buffers and on the sTTS status.
2. To generate the fast commands to be distributed to the subdetectors by the TTC network.
3. To collect the sTTS fast monitoring feedback from the subsystems and to generate the appropriate action when necessary.
4. To generate calibration and test trigger sequences.
5. To monitor the experiment deadtime

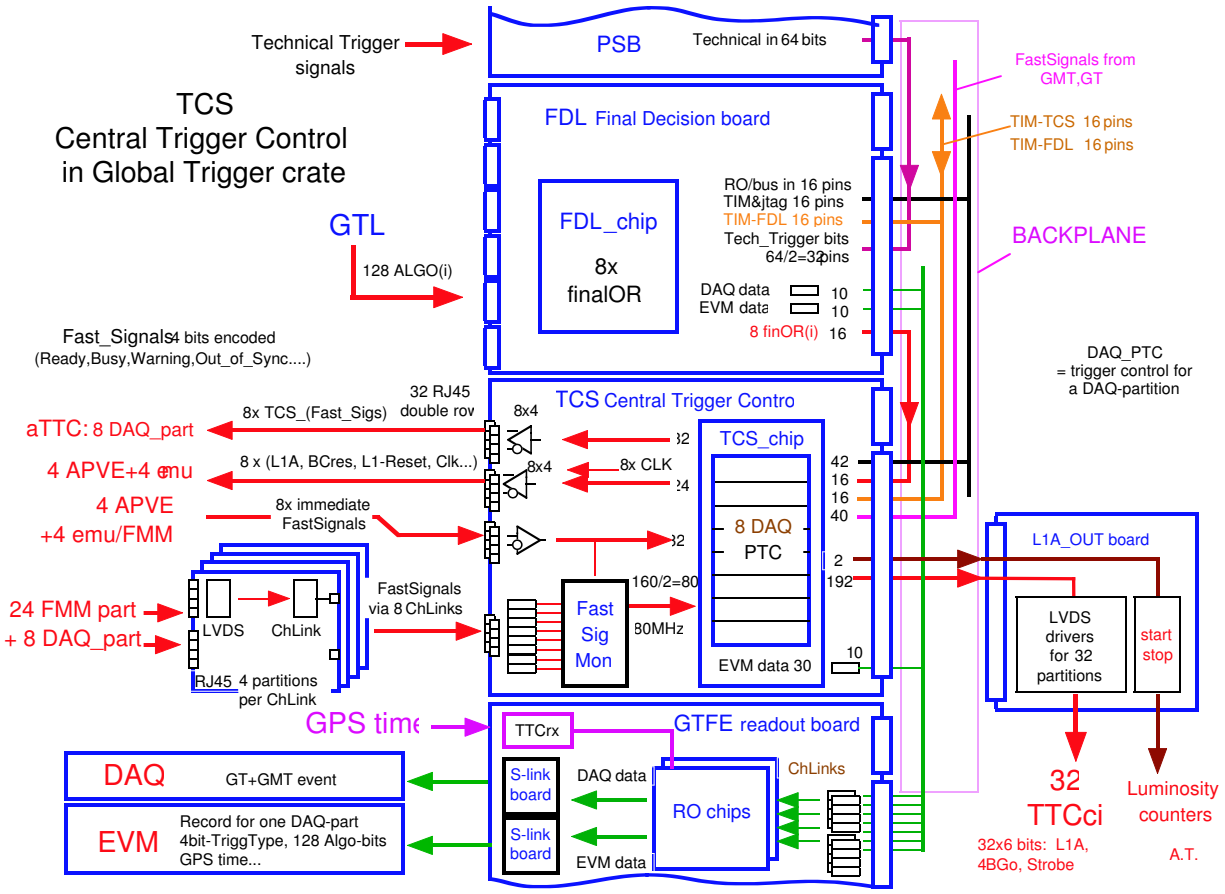



**Fig. 3:** Block diagram of TCS in Global Trigger Crate

## 3.2 Input/Output

The central TCS has the following hardware interfaces to external subsystems:

1. Interface to the Global Trigger, from where it receives the L1A information:

– 128 trigger bits and 8 final-OR transmitted in the crate backplane.

2. Interface to LHC machine Clock and Orbit signals (two cables, ECL) via TIM board of the GT crate. For test purposes, internal generation of clock and orbit may be chosen.

3. Interface to 32 TTCci modules, to distribute L1A and Fast Commands to the subdetector partitions.

– 32 cables with 8 bits LVDS: L1A, Bgo3...0, Strobe, Orbit, Clock (B-Go3...0 addresses 16 B-Go-circuits in the TTCci, to send broadcast commands via TTC B-channel to partitions)

4. Interface to sTTS to collect information on the status of the front-end electronics:

– 24 cables from partitions and 8 cables from emulators (4 for Tracker APV-Emulators; 2 for Preshower emulators; 2 free)

– for each connection, 4 bits parallel LVDS, RJ45 and Ethernet cable UTP5 (or shielded STP6);

5. Interface to Emulators:

– 8 cables to emulators, each sending Clock, L1A, BC0, L1 Reset, Event Counter Reset, Orbit Counter Reset.

– for each connection, 4bit, LVDS, RJ45 link.

6. Interface to the DAQ Event Manager, to transmit L1A blocklet (see Section 2.1). The overall trigger rate is limited to 100 kHz even in multi-partition mode.

7. Interface to aTTS, 8 TTS channels (1 per DAQ partition) input and 8 TTS channels (1 per DAQ partition) output.

8. Interface to BST system, one TTC input link to a dedicated TTCrx in the GTFE board.

9. Interface to Run Control/Detector Control Systems (VME interface).

## 3.3 Interface to TTC Partitions

The TCS sends signals (B-Go signals) to each TTCci that will trigger the transmission of the broadcast commands, stored in the TTCci FIFOs, through the TTC channel B. The B-Go signals will be used for BC0, event counters reset, fast resets, test enable and trigger start/stop commands. Four B-Go bits (3,2,1,0) address 16 B-Go circuits in the TTCci. The Strobe starts the addressed B-Go circuit. The list of B-Go commands is given in Table 5.

**Table 5:** B-Go Commands

| B-Go channel | B-Go bits | Command | |
|---|---|---|---|
| 1 | 0001 | BC0 | Reset Bunch crossing counters to begin a new LHC orbit |
| 2 | 0010 | Test Enable | to start calibration procedure |
| 3 | 0011 | Private Gap | |
| 4 | 0100 | Private Orbit | |
| 5 | 0101 | ReSync | (=L1Reset) clears buffers and pipelines...etc. |
| 6 | 0110 | HardReset | |
| 7 | 0111 | Reset Event Counter | |
| 8 | 1000 | Reset Orbit Counter | normally sent at begin of a new run |

**Table 5:** B-Go Commands

| B-Go channel | B-Go bits | Command | |
|---|---|---|---|
| 9 | 1001 | Start | start data taking with next orbit |
| 10 | 1010 | Stop | stop data taking with next orbit |

## 3.4 Interface to aTTS

Each of the 8 DAQ partition controllers on the TCS board sends 4 LVDS signals to aTTS (Table 6). The signals show the status of central TCS and are defined like the Fast Signals from the front-end and trigger electronics (except for the additional Idle state). The four signals are applied as voltage levels as long as the corresponding status is valid. The inactive level corresponds to a disconnected cable. The TCS status have the following interpretation:

- READY: TCS and all partitions connected to this DAQ partition are ready (=AND of input READY signals). TCS allows L1A.
- BUSY: TCS or one of the partitions connected to this DAQ partition is busy (=OR of input BUSY signals). TCS does not allow L1A.
- WARNING: TCS is running with reduced trigger rate because one of the connected partitions is applying a WARNING
- OUT_OF_SYNC: One of the partitions connected to this DAQ partition has sent an Out_of_Sync (=OR of Out_of_Sync signals). TCS is going to run a RESYNC procedure or has entered the error state.
- DISCONNECTED: TCS has not been initialized.
- IDLE: TCS has been set-up and waits for the START of the run.
- ERROR: TCS cannot resynchronize the partitions and doesn‘t work anymore.

**Table 6:** Status signals from central TCS to aTTS (DAQ)

| T3 (Ready) | T2 (Busy) | T1 (OutofSync) | T0 (Warning) | TCS STATUS |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | DISCONNECTED * |
| 0 | 0 | 0 | 1 | OVERFLOW WARNING |
| 0 | 0 | 1 | 0 | OUT_OF_SYNC |
| 0 | 0 | 1 | 1 | forbidden |
| 0 | 1 | 0 | 0 | BUSY |
| 0 | 1 | 0 | 1 | forbidden |
| 0 | 1 | 1 | 0 | forbidden |
| 0 | 1 | 1 | 1 | forbidden |
| 1 | 0 | 0 | 0 | READY |
| 1 | 0 | 0 | 1 | forbidden |
| 1 | 0 | 1 | 0 | IDLE ** |
| 1 | 0 | 1 | 1 | forbidden |
| 1 | 1 | 0 | 0 | ERROR |
| 1 | 1 | 0 | 1 | forbidden |
| 1 | 1 | 1 | 0 | forbidden |
| 1 | 1 | 1 | 1 | DISCONNECTED * |

Notes:

*) Depends on the output level of unconnected LVDS receiver chips.

**) Status of TCS partition control logic. All other states are defined like normal TTS channels.

## 3.5 Partitioning

The central TCS provides control of 32 physical partitions. Identical units implement the L1A and fast command generation, as well as the trigger throttling and calibration functions, for each subdetector partition (see Figure 4).

The trigger type word tell the EVM to which subsystem the current L1A has been sent. The central TCS module is physically located in the Global Trigger crate in order to minimize the L1 trigger latency.

### L1A priority scheme

At each bunch crossing, up to 8 final-OR triggers may be active, each corresponding to one DAQ partition. However, it is required that at each crossing L1A is sent to only one partition. In order to handle multiple triggers, a priority scheme is defined.

In case all DAQ partitions have the same priority (low priority), the L1A are distributed in round robin mode (partition 1 receives trigger 1, then partition 2 receives trigger 2, etc.). The trigger rate of all partitions is limited to the lowest partition trigger rate.

If one partition is defined with high priority (main experiment partition), triggers to this partition have priority. The remaining partitions receive triggers in round robin mode as described before.

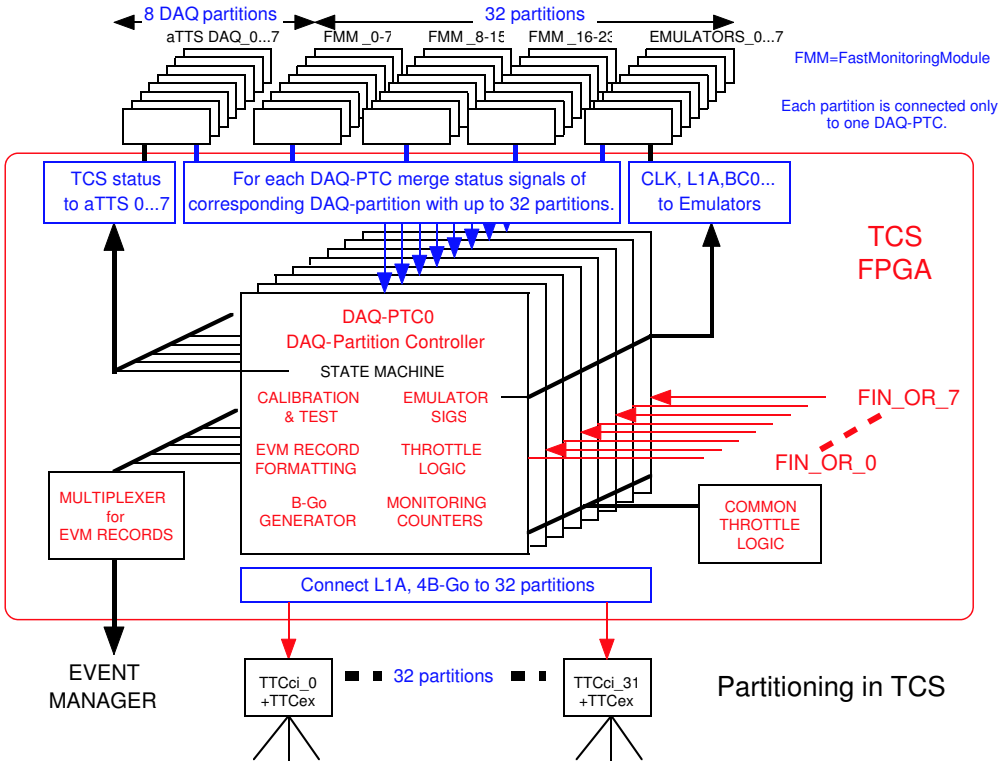



**Fig. 4:** Partitioning of the Trigger Control System.

## 3.6 TCS State Machine

The central TCS handles all the sTTS signals received from the subdetector partitions and from the aTTS with a state machine programmed either to stop L1A signals or to deliver Reset signals, if needed.

### Input signals to central TCS

The list of input signals to central TCS is described in Table 7. All 4 bits must be applied at the same clock.

**Table 7:** Input signals to central TCS

| T3 (Ready) | T2 (Busy) | T1 (OutofSync) | T0 (Warning) | PARTITION STATUS |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | DISCONNECTED * |
| 0 | 0 | 0 | 1 | OVERFLOW WARNING |
| 0 | 0 | 1 | 0 | OUT_OF_SYNC |
| 0 | 0 | 1 | 1 | forbidden |
| 0 | 1 | 0 | 0 | BUSY |
| 0 | 1 | 0 | 1 | forbidden |
| 0 | 1 | 1 | 0 | forbidden |
| 0 | 1 | 1 | 1 | forbidden |
| 1 | 0 | 0 | 0 | READY |
| 1 | 0 | 0 | 1 | forbidden |
| 1 | 0 | 1 | 0 | forbidden |
| 1 | 0 | 1 | 1 | forbidden |
| 1 | 1 | 0 | 0 | ERROR |
| 1 | 1 | 0 | 1 | forbidden |
| 1 | 1 | 1 | 0 | forbidden |
| 1 | 1 | 1 | 1 | DISCONNECTED * |

Notes:

*) Depends on the output level of unconnected LVDS receiver chips.

### Internal STATES of central TCS

(See also Table 6).

- *IDLE STATE:* TCS waits for Run Control Software
- *L1RESYNC_ini STATE:* TCS sends a Resync signal to initialize the partitions at begin of a Run
- *RESET ORBIT NUMBER and START:* TCS starts the partitions.
- *L1A DISABLED_ini STATE:* TCS inhibits L1A and waits until partitions become ready after resynchronisation.
- *RESET EVENT NUMBER STATE:* TCS initializes event numbers.
- *L1A DISABLED STATE:* TCS inhibits L1A and waits until partitions become ready again.
- *L1A ENABLED STATE:* TCS runs with normal trigger rate.

- *LOW RATE STATE:* TCS runs with low trigger rate.
- *HARD-RESET STATE:* TCS sends HardReset to partitions
- *RESYNC STATE:* TCS resynchronize partitions sending a Resync signal and waits then some time for the READY signal.
- *ERROR STATE:* TCS stops L1A, sets an error flag and waits for the software.
- *STOP STATE:* TCS sends a STOP_RUN to the partitions and goes into the IDLE state.

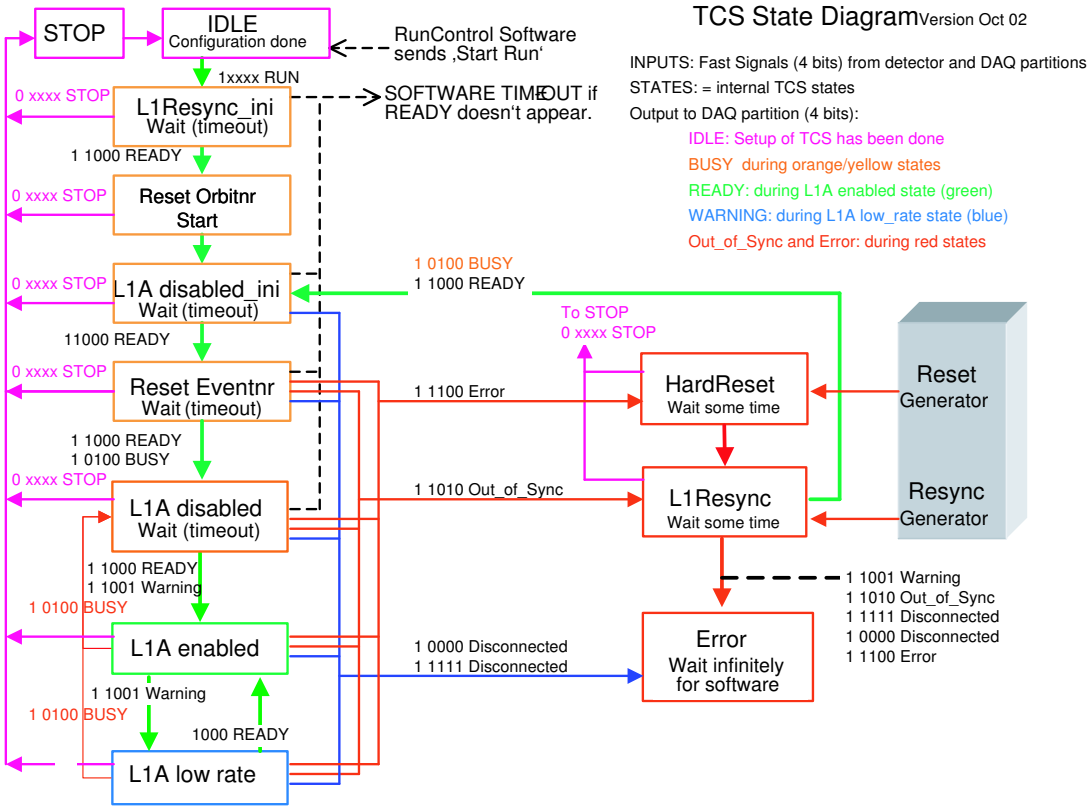



**Fig. 5:** TCS State Diagram.

## Response to TTS signals

1. Partition sends READY (subsystem accepts L1A)
   - TCS-hardware runs normally
2. Partition sends BUSY (subsystem cannot accept L1A)
   - TCS-hardware sets BUSY flag and inhibits L1A for this partition group.
   - If BUSY disappears and READY becomes active then TCS allows L1A again.
   - If BUSY does not disappear then:

   TCS-software goes into time-out after xx sec and pauses the data taking run.
3. Partition sends WARNING
   - TCS-hardware sets WARNING flag and reduces L1A rate for this partition group.

– If WARNING disappears then TCS allows normal L1A rate again

4. Partition is DISCONNECTED (setup not done, cables removed...)
   – TCS-hardware inhibits L1A for this partition group and sets Disconnected flag
   – and either ignores the Disconnected signal and waits for READY=1 or
   – starts a L1Resync procedure
5. Partition sends OUT OF SYNC (sync loss in subsystem)
   – TCS-hardware inhibits L1A for this partition group and sets the OUT OF SYNC flag
   – and starts a Resync procedure
6. Partition sends ERROR (error in subsystem)
   – TCS-hardware inhibits L1A for this partition group and sets the ERROR flag
   – and starts a HardReset procedure

## Fast Resets

Two fast reset commands are implemented:

1. *ReSync:*

– Intended to recover from synchronization losses.

– A request for ReSync may be generated by a subsystem on detection of a synchronization loss condition (sTTS signal OutOfSync)

2. *Hard Reset:*

– Intended to recover from some electronics problems. Reset of readout state machines or fast reconfiguration of FPGA's. Programmable parameters downloaded by software should not be affected by this reset.

– A request for Hard Reset may be generated by a subsystem on detection of an error condition (sTTS signal Error)

## ReSync and HardReset procedure

The fast reset procedure has the following steps (Figure 6):

1. Central TCS

– TCS-hardware receives OutOfSync

– TCS-hardware sets OutOfSync flag and inhibits L1A

– TCS-hardware waits a programmable time interval to allow Partition(s) to discard or to send pending events to DAQ, and to clear their readout buffers and pipelines.

– TCS-hardware (or software) sends ReSync to the TTCci (or all TTCci of a group)

2. Partitions

– assert Busy signal

– read pending data in readout buffers (or discard data), reset pipelines and readout buffers

– assert Ready signal

3. Central TCS:

– TCS sends Event Counter Reset

– TCS checks Ready signals

– TCS sends BC0 at the beginning of a new orbit and resumes L1As

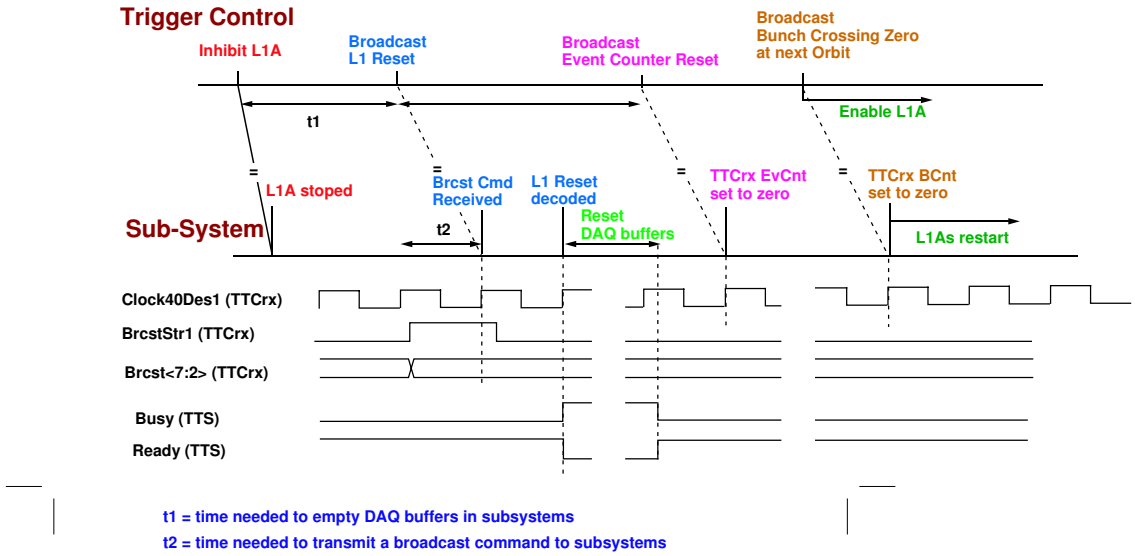



**Fig. 6:** ReSync timing diagram.

A similar procedure is followed for the Hard Reset command. The main difference is the type of reset performed on the electronics.

### Bunch Crossing Zero

The BC0 signal is sent to the TTCci's synchronously with the LHC Orbit signal. The phase of the BC0 commands relative to the Orbit signal is globally adjusted per subdetector partition with a programmable delay in the TTCci, and locally adjusted at the level of the receiver TTCrx in a smaller range (16 clock periods). It is the responsibility of the subdetectors to adjust the BC0 timing.

The BC0 command is used by the trigger system to synchronize the trigger data in the trigger processing pipeline. A BC0 flag should accompany the data corresponding to the last crossing without collisions at the end of the Abort Gap. The BC0 command is used by the readout systems to reset (or to check) the Bunch Counter. It is also used by the subdetectors to localize the orbit gap when particular actions are needed in that period (e.g. subdetector specific reset or test procedures).

The TTCci programmable capabilities are used to tailor the BC0 command to the particular needs of the subdetectors (timing, TTC command codes; see Section 5.).

BC0 is inhibited during the fast reset procedures.

## 3.7 Start and Stop Run procedures

Start run procedure:

- The hardware configuration has been done for all subsystems and the TCS.
- Run Control System (RCS) sends START_RUN instruction to TCS.
- TCS follow the state diagram of Figure 5.

Stop run procedure:

- Run Control Software (RCS) sends STOP_RUN instruction to TCS.
- TCS-hardware waits until the end of the orbit.
- TCS disables L1A and waits for the next gap.
- TCS sends STOP to partitions/subsystems during the gap.
- Partition(s) stop their counters (orbit counter,...). *//Used for crosscheck between subsystems.*

– TCS-hardware goes to IDLE state.

## 3.8 Trigger Rules

To attempt to limit overflows as much as possible, a set of Trigger Rules for minimal spacing for L1As are implemented in central TCS. Examples of these rules are:

i) No more than 1 Level 1 Accept per 75 ns (minimum 2 bx between L1A), dead time $5.10^{-3}$ (required by tracker and preshower)
ii) No more than 2 Level 1 Accepts per 625 ns (25 bx), dead time $1.3.10^{-3}$.
iii) No more than 3 Level 1 Accepts per 2.5 μs (100 bx), dead time $1.2.10^{-3}$.
iv) No more than 4 Level 1 Accepts per 6 μs (240 bx), dead time $1.4.10^{-3}$.

The total deadtime cost for such rules (estimated for L1A rate 100 kHz) is of the order of 0.9%.

## 3.9 Front-end Buffers Emulation

According to the front-end electronic logical model, the front-end derandomizers after the L1 latency pipelines are the first devices to overflow when the L1A rate is too high. The use of the fast Overflow Warning to stop the L1As is not effective in this situation. The latency of sTTS is of the order of a few microseconds, mainly due to cable lengths, during which additional triggers can occur. In practice a reserve of a few events in the derandomizer would be needed for these triggers, so that the effective length of the derandomizer is smaller than the real one, implying a larger inefficiency.

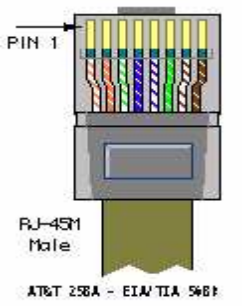
To handle this problem, readout buffers with deterministic behavior, as it is the case of the tracker and preshower derandomizers, are emulated centrally. The L1A is inhibited if it is found to create an overflow in the front end buffers.

The subdetectors that need frontend buffer emulation are the tracker and the preshower. The emulators are implemented in VME6U modules.

### Emulators Input/output

The emulators receives directly from the Central TCS and Local TCS the Clock, L1A, BC0 and Reset. These signals are electrical LVDS, RJ45 link, with the following pin assignment:

pin 1 = CLK_P
pin 2 = CLK_N
pin 3 = RESET_P
pin 4 = L1A_N
pin 5 = L1A_P
pin 6 = RESET_N
pin 7 = BC0_P
pin 8 = BC0_N




The 3 bits BCO, Reset and L1A encode the following commands:

(BC0,RESET,L1A)= (0,0,0) => ' inactive' //=no signal on the pins

(BC0,RESET,L1A)= (0,0,1) => L1A

(BC0,RESET,L1A)= (0,1,0) => RESYNC(L1Reset)

(BC0,RESET,L1A)= (0,1,1) => Reset Event Counter

(BC0,RESET,L1A)= (1,0,0) => BC0

(BC0,RESET,L1A)= (1,0,1) => Reset Orbit Counter

(BC0,RESET,L1A)= (others) => forbidden

A VME write to the emulator decides which TCS source is selected (or neither).

The final FMM status is sent to the emulators via a sTTS 4bit, LVDS, RJ45 link (see Section 2.4). In the emulator the FMM status is merged with that from the front-end chips. The combined status of the emulator and that of the FMM modules is sent back to both the Central and Local TCS via two independent 4bit, LVDS, RJ45 links (e.g. if the Central TCS is selected in the emulator the status sent back to the Local TCS will be ERROR whereas that sent back to the Central TCS might be ERROR, OOS, WARN, BUSY or READY).

The emulator records the history of the status signals sent to the Central and Local TCS as well as the internal FMM and emulator status. A memory record contains bunch crossing number (12 bit), the orbit number (32 bit) and the CTCS, LTCS, APVE, FMM status (4x4bits). The memory is 4k deep.

## 3.10 Deadtime Monitor and Counters

The TCS has a number of counters that are used for monitoring purposes and that are made available through RCS. Some of these counters are used to measure the experiment dead time.

There are 3564 beam crossings per LHC orbit, but during the so-called ‘gaps’ there will be by design no interactions between protons. Active beam crossing are the crossings were interactions between protons are possible (number of active beam crossings is equal to all beam crossings minus the gaps crossings).

The TCS has the following counters that are incremented for the complete orbits between the fast Start and Stop commands:

1. Number of orbits (Orbit Number).
2. Number of beam crossings where L1A was potentially inhibited.
3. Number of beam crossings where L1A was potentially inhibited, and L1A is True
4. Number of beam crossings where L1A was potentially inhibited, and L1A is False
5. Number of active beam crossings where L1A was potentially inhibited.
6. Number of active beam crossings where L1A was potentially inhibited, and L1A is True
7. Number of active beam crossings where L1A was potentially inhibited, and L1A is False
8. Number of active beam crossings where L1A was potentially inhibited, per inhibit condition
9. Number of physics triggers (physics L1As without dead time).
10. Number of distributed physics triggers (physics L1As with dead time).
11. Number of distributed calibration or test triggers
12. Total number of distributed triggers (Event Number)

These counters are implemented per partition-group (DAQ partition). For each new crossing the TCS combines all possible L1A inhibit conditions (trigger rules, buffer emulators, TTS requests to hold L1A, calibration or reset activities, etc.) and creates the final inhibit signal. This signal is used to prevent potential L1A to be distributed and to gate livetime counters.

Two NIM signals are available externally to gate the luminosity counters, the run signal, active between start and stop, and the L1A inhibit signal. These signals are available only for the first partition (global partition in normal psysics mode).

## 3.11 Calibration and Test Control

### Calibration and Test modes

Calibration and test operation modes:

1. *Subdetectors in standalone mode:*

Test and calibration sequences are generated locally, by the Local Trigger Controller or by the TTCci;

Data is captured with the subdetector local DAQ.

2. *Subdetectors in DAQ partition mode:*

   Central Trigger Controller generates test or calibration triggers at the rate required by the subdetector partition (up to 100 kHz, following trigger rules);

   Data is collected by the central DAQ.

3. *Central test or calibration triggers during a Physics Run:*

   Calibration triggers sequences are issued centrally and distributed to all partitions;

   All subsystems must deliver an event data block (the event block can be empty);

   Calibration/test triggers are issued at pre-programmed cycles in the LHC orbit;

   Data is collected by the central DAQ.

4. *Local test or calibration triggers during a Physics Run:*

   Test or calibration triggers can also be handled at the subsystem level during a Physics Run provided the test or calibration activities occur during the Private Gaps or Private Orbits marked by TCS. The Event Number is not incremented for these triggers.

**Table 8:** Calibration Control Signals

| Fast Control Signals | B-Go Channel | Comments |
|---|---|---|
| Test Enable | 2 | Broadcast command sent a fixed time before a calibration trigger. |
| Private Gap | 3 | Broadcast command marking the next gap for private use by the subdetectors |
| Private Orbit | 4 | Broadcast command marking the next orbit for private use by the subdetectors |

Table 8 lists the fast signals issued by TCS and distributed by TTC to control the calibration and test activities.

## Calibration Trigger Sequence

The TCS sends to all subsystems a Test Enable signal a fixed number of crossings before the corresponding L1A. The subsystems set up their tests so that the test data will be contained in the exact crossing indicated by the TCS Test Enable signal. The TCS inhibits normal triggers between Test Enable and the corresponding L1A. The TCS then sends out a Level 1 Accept for the appropriate crossing to read in the test data.

The timing of the test and calibration triggers can be programmed in order to happen at pre-defined bunch crossing numbers in the LHC orbit (Figure 7). Normally these triggers will occur during the LHC main gap but they can also be delivered in superposition to beam crossings to study pile-up effects. A programmable delay in the TTCci between external B-Go signals and B-channel transmission allow to adjust the timing of test pulses (see Section 5.).

When a TestEnable is sent, the corresponding L1A is always be generated, even if a given sub-system raises BUSY between the two. This is because the sub-systems are assuming the next trigger is a calibration and may have special procedures to handle these.

When received by the subsystems front-ends, the test and calibration L1As are treated in the same way as the physics triggers. The front-end readout sequence is the same, the Event Number is incremented and the data is collected by the central DAQ. As pointed-out before, the subsystems may respond with an empty event block.

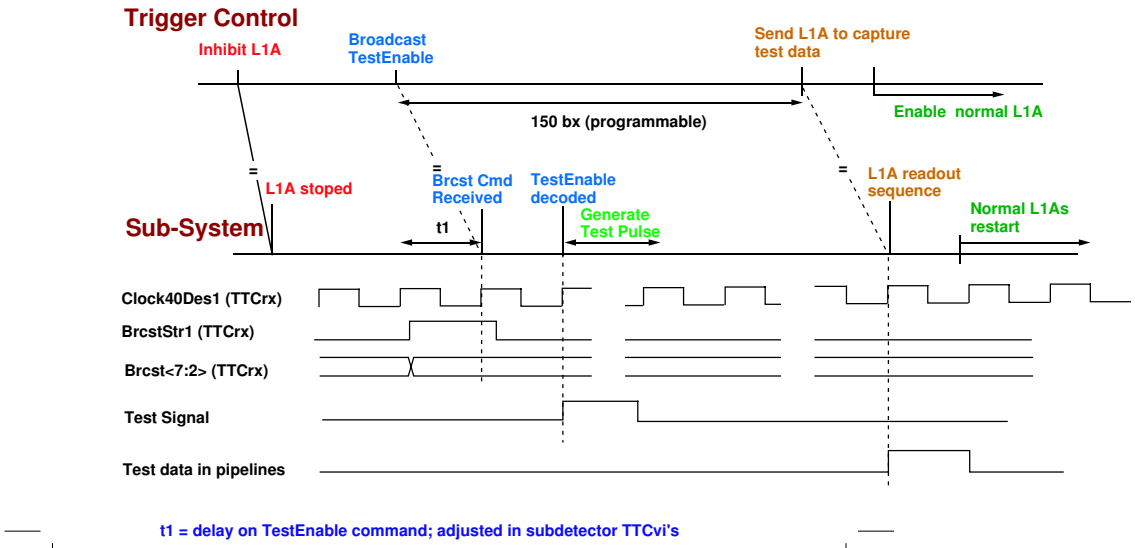



**Fig. 7:** Calibration trigger timing diagram.

## Calibration Logic

The Calibration logic in TCS controls calibration cycles in close collaboration with the connected TTCci board. A calibration cycle includes a ' Test Enable' signal, the generation of data in the subsystem (laser pulses, test pulses, test patterns...) and the L1A to read the data. The programmed parameters in the TCS Calibration module and in the TTCci board have to match correctly. The Calibration logic either inserts just one calibration cycle during normal data taking or runs calibration cycles continuously.

When a calibration cycle is initiated a B_Go(2) signal (Test Enable) is sent to the TTCci board at a programmed bunch crossing number to start a calibration cycle. The next preloaded command stored in the TTCci FIFO(2) is sent to the subsystem front-end to start the calibration procedure. The actual code of the synchronous commands stored in FIFO(2) is subdetector dependent. It may include one or more ' Test Enable' codes for different types of tests and an ' Empty Event' code if the subsystem is not interested in the test data. Some time later (of the order of 100 bx) at a preloaded bx number the Calibration logic sends a L1A to read the calibration data. During the time from B_Go(2) until the corresponding L1A the normal physics triggers are stopped.

Options for test and calibration trigger generation:

a) Synchronously to BC-nr and Orbit
b) Prescaling: every n-th orbit
c) Random calib-cycles at a given average rate

- For this mode a consecutive B-Go/L1A cycle is inhibited until the current cycle has been finished.
- Also the trigger rules are observed by the trigger throttling circuit.

Internal triggers may be generated in calibration mode (combined Test Enable - L1A) or in test mode (L1A alone).

## Private Gaps and Private Orbits

TCS will issue periodically the command Private Gap (Private Orbit) marking the next Gap (Orbit) available for private use by the subdetectors. The subsystems may use the gap to generate test/calibration signals at the top of the subsystem TTC (TTCci) or by an independent system. The Event Number is not incremented and the data is collected by the subsystem local DAQ.

1. Private Orbit sequence:

– TCS disables L1A at the end of the current orbit and sends a signal to B-Go channel 4 (Private Orbit)

– During the following orbit:
  • TCS ignores the Fast Signals
  • TCS does not send any B-Go command (except BC0) (passive mode).
  • TCS sends 'private' B-Go commands (active mode).
– After the private orbit
  • TCS enables L1A again, depending from the connected Fast Signals

2. Private Gap sequence:
– At a given bunch crossing TCS sends a signal to B-Go channel 3 (Private Gap)
– During the next gap
  • TCS does not send any B-Go command (passive mode).
  • TCS sends 'private' B-Go commands (active mode)

## 3.12 Timing issues

By definition, Bunch Crossing *n* in TCS is defined by the clock cycle synchronous with L1A signal from bunch *n* in TCS.

In order to guarantee that B-channel commands are synchronous, a number of rules are implemented to avoid overlapping of commands:

a) All B-Go signals from TCS must be separated by 88 clocks, at minimum. This requirement is guaranteed by the TCS configuration software.
b) B-Go signals are assigned to a precise BC number.
c) BC0 is assigned to bunch 0, that is the last bunch of the gap without collisions. Bunch 1 is the first bunch with collisions.
d) During data taking, Test Enable is assigned to bunch-n (programmable), and is issued with a programmable prescaling rate. PrivateGap and PrivateOrbit are sent in place of TestEnable (same bunch number) at a programmable frequency.
e) ReSync and HardReset are sent at bunch-n (programmable), when requested or periodically generated (avoiding collision with BC0). TestEnables are disabled during these cycles.

Per si these rules are insufficient to prevent overlaps on channel-B. However, once defined, it is easy to guarantee that the programmable delays applied in the TTCci to the B-Go signals do not cause any overlap (Section 5.). This task is performed by the configuration software.

## 3.13 Logging

The history of the fast signals received and of the control sequences generated by TCS are logged and can be made available for diagnostic.

# 4. Local Trigger Controller

## 4.1 Overview

The Local Trigger Controller (LTC) is physically located in the subdetector trigger control crate (VME6U), which contains also the TTCci and TTCex modules (Figure 2).

The LTC allows to control the partitions of a subdetector, replacing the central TCS in its functionality (Section 3.1). The LTC control up to six TTC partitions, but it is not required to run multiple partition-groups in parallel. One single partition-group or DAQ partition can run at a given time.

## 4.2 Input/Output

The LTC has the following hardware interfaces to external subsystems:

1. Input to 6 local triggers (NIM).
2. Interface to LHC machine Clock and Orbit signals (two cables, ECL). Internal clock and orbit generation is possible.
3. Clock outputs (2) in ECL
4. Interface to 6 TTCci modules, to distribute L1A and Fast Commands to the subdetector partitions (6 cables with 8 LVDS signals: L1A, B-Go3...0, Strobe, Orbit, Clock)
5. Interface to one TTCvi module: L1A (ECL), B-Go3...0 (NIM)
6. Interface to sTTS to collect information on the status of the front-end electronics (6 cables with 4 bits parallel LVDS, RJ45)
7. Interface to Emulators: 4 cables to emulators, each sending Clock, L1A, BC0, L1 Reset, ECR, OCR. For each connection, 4bit, LVDS, RJ45 link.
8. Interface to the DAQ Event Manager (S-Link), to transmit L1A blocklet (Event Number, Partition, Trigger Type, Trigger bits).
9. Interface to aTTS, 1 TTS input and 1 TTS output.
10. Interface to BST system, one TTC input link to a dedicated TTCrx (see Section 2.1).
11. Interface to Run Control/Detector Control Systems (VME interface).

## 4.3 Interface to TTC Partitions

The LTC sends to each TTCci in the subdetector the same B-Go signals as the central TCS (Section 3.3). The list of B-Go commands is given in Table 5.

For backward compatibility, the LTC provides one output for the TTCvi module. The 4 B-Go NIM outputs carry the following subset of commands in Table 5:

B-Go 0 - BC0

B-Go 1 - Test Enable

B-Go 2 - ReSync

B-Go 3 - Reset Event Counter

## 4.4 Interface to aTTS

LTC has an interface to one aTTS partition. This interface is identical to the central TCS (Section 3.4). It is be used for tests or for operation in test beam.

## 4.5 Partitioning

LTC provides control of up to 6 physical partitions (TTC partitions). However, multi-partition operation is not possible.

## 4.6 Trigger Rules

LTC implements the same type of Trigger Rules as the central TCS (Section 3.8).

## 4.7 Front-end Buffers Emulation

The Emulator Modules combine its own TTS states with the partition TTS states (from FMM), into a combined TTS state available at Emulator output. This output can be used as sTTS input in the LTC.

The emulators receives directly from LTC the Clock, L1A, BC0, L1 Reset, Event Counter Reset and Orbit Counter Reset. These signals are electrical LVDS, RJ45 link. The interface is identical to that in central TCS (see Section 3.9).

## 4.8 Deadtime Monitor and Counters

The LTC has the following counters that are incremented for the complete orbits between the fast Start and Stop commands:

1. Number of orbits (Orbit Number).
2. Number of beam crossings where L1A was potentially inhibited.
3. Number of beam crossings where L1A was potentially inhibited, and L1A is True
4. Number of beam crossings where L1A was potentially inhibited, and L1A is False
5. Number of triggers (L1As without dead time).
6. Number of distributed L1As (Event Number).

For each new crossing, LTC combines all possible L1A inhibit conditions (trigger rules, buffer emulators, TTS requests to hold L1A, calibration or reset activities, etc.) and creates the final inhibit signal. This signal is used to prevent potential L1A to be distributed and to gate livetime counters.

Two signals are available externally to gate the luminosity counters, namely the run signal, active between start and stop, and the L1A inhibit signal.

## 4.9 Calibration and Test Control

LTC allow the subdetectors to run in standalone mode. Test and calibration triggers are generated locally, by the Local Trigger Controller and transmitted to the TTCci's. Data is captured with the subdetector local DAQ.

Table 9 lists the fast signal issued by LTC to control the calibration triggers. Internal generation of L1A is possible.

**Table 9:** LTC Calibration Control Signal

| Fast Control Signals | B-Go Channel | Comments |
|---|---|---|
| Test Enable | 2 | Broadcast command sent a fixed time before a test or calibration trigger |

### Calibration trigger sequence

The calibration trigger sequence is identical to the sequence generated by the central TCS.

The LTC sends a Test Enable signal a fixed number of crossings before the corresponding L1A and then sends out a Level 1 Accept for the appropriate crossing to read in the test data.

The timing of the calibration trigger can be programmed in order to happen at pre-defined bunch crossing numbers in the LHC orbit (see Figure 7). A programmable delay in the TTCci between external B-Go signals and B-channel transmission allow to adjust the timing of test pulses (see Section 5.).

When a TestEnable is sent, the corresponding L1A is always be generated, even if a given partition raises BUSY between the two. This is because the sub-systems are assuming the next trigger is a calibration and may have special procedures to handle these.

### Options for test trigger generation

LTC has the following option for internal trigger generation:

a) Synchronously to BC-nr and Orbit
b) Prescaling: every n-th orbit
c) Random calib-cycles at a given average rate

- For this mode a consecutive B-Go/L1A cycle is inhibited until the current cycle has been finished.
- Also the trigger rules are observed by the trigger throttling circuit.

Internal triggers may be generated in calibration mode (combined Test Enable - L1A) or in test mode (L1A alone).

## 4.10 Timing issues

By definition, Bunch Crossing *n* in LTC is defined by the clock cycle synchronous with L1A signal from bunch *n* in LTC. The LTC should have the capability to adjust the bunch counter offset for correct identification of the bunch number. The bunch counter offsets may be deduced from an histogram of the number of triggers as a function of bunch number.

In order to guarantee that B-channel commands are synchronous, a number of rules are implemented to avoid overlapping of commands:

a) All B-Go signals from LTC must be separated by 88 clocks, at minimum. This requirement is guaranteed by the LTC configuration software.
b) B-Go signals are assigned to a precise BC number.
c) BC0 is assigned to bunch 0, that is the last bunch of the gap without collisions. Bunch 1 is the first bunch with collisions.
d) During data taking, Test Enable is assigned to bunch-n (programmable), and is issued with a programmable prescaling rate.
e) ReSync and HardReset are sent at bunch-n (programmable), when requested or periodically generated (avoiding collision with BC0). TestEnables are disabled during these cycles.

## 4.11 Logging

The history of the fast signals received and of the control sequences generated by LTC are logged and can be made available for diagnostic.

# 5. TTCci Module

## 5.1 Overview

The TTCci (TTC CMS Interface) is the CMS version of the TTCvi module [3]. The TTCci is the top element of each of the 32 TTC partitions. It provides the interface between the (central or local) Trigger Controller and the TTC destinations for transmission of L1A and fast commands. Switching between central and local trigger control is achieved by software programming. For test purposes, the TTCci can operate in standalone mode.

The TTCci delivers Channel A and Channel B signals to the TTC transmitters (TTCex). Channel A is used to transmit L1A and Channel B is used to transmit framed and formatted commands and data. These can be either:

- short-format (8-bit) broadcast command/data cycles (deskewed in the TTCrx ASIC);
- long-format (32-bit) individually-addressed or broadcast command/data cycles (not deskewed in the TTCrx).

There are four methods to transmit commands/data:

i) synchronous to external B-Go signals;
ii) synchronous to Orbit input signal;
iii) asynchronously, under software control;
iv) through auxiliary B-data input channel.

The TTCci is implemented as a VME6U module.

## 5.2 Input/Output

### Clock and Orbit inputs

The TTCci module is driven by the 40.08 MHz clock it receives from the TTCex module. The clock input to TTCex is derived from the TTCmi crate (TTC machine interface). The phase of the input clock can be adjusted by means of a rotary switch on the TTCci. The phase is adjusted such that the A and B Channel output signals from the TTCci are delivered to the TTCex within the appropriate phase window.The external clock input is 50Ω AC coupled ECL.

An internal 40 MHz clock may be selected by module configuration.

The orbit signal is a pulse train of period 88.924 μs, received from the TTCmi. The LHC orbit input is 50Ω AC coupled ECL. For test purposes, internal generation of the orbit signal may be chosen.

Optionally, the TTCci may use the clock and orbit signals transmitted directly in the TCS/LTC connections (see next). The choice of the clock and orbit source is made by module configuration.

A clock output (ECL) is also provided.

### L1A and B-Go inputs

The TTCci has two main inputs for L1A and B-Go signals, one from the central TCS and another from local LTC. In both cases, the input is 8 LVDS signals: 1 L1A, 4 B-Go signals, 1 B-Go strobe, Orbit, Clock.

Encoding of external B-Go signals:

- Ch0 0000- Not used
- Ch1 0001- BC0
- Ch2 0010- Test Enable
- Ch3 0011- Private Gap
- Ch4 0100- Private Orbit

- Ch5 0101- Resync
- Ch6 0110- Hard Reset
- Ch7 0111- Reset Event Counter
- Ch8 1000- Reset Orbit Counter
- Ch9 1001- Start
- Ch10 1010- Stop
- Ch11 to 16 - Free

### Auxiliary triggers

Input for two external auxiliary triggers (NIM logic) is provided in the front-panel. The choice of the trigger source is programmable.

### B-data input

An auxiliary input for data to be transmitted in the Channel B is provided in the front-panel. This input is 8-bit data and a strobe in LVDS. Data strobed is transmitted with low priority through Channel B. If Channel B is being used, data is stored in an input FIFO waiting for channel availability.

The transmission of the 8-bit B-data uses a long-format broadcast command (address zero), in order to avoid conflict with pre-defined broadcast codes.

This transmission channel is used to send the frontend pipeline address, as computed in the central emulator, to the FEDs for cross-checking.

### Output

The TTCci has Channel A and Channel B outputs identical to the TTCvi [3].

## 5.3 TTCci Functions

### Input multiplexing

Multiplexing of L1A and B-Go inputs to allow control by Central TCS or by Local TCS (programmable). The trigger external input may be chosen between Central TCS, Local TCS and the two auxiliary trigger inputs.

### B-Go Channels

The TTCci has 16 B-Go Channels that allow the transmission of pre-loaded TTC command/data (short-format or long-format). The data is stored in FIFOs, one per B-Go Channel. Each FIFO is 32x16k bits. The 32-bit width allow to store both short and long commands/data.

For each of the 16 channels, transmission of the next command/data in the FIFO is initiated by a signal called B-Go<16..1>. Depending on the chosen B-Go channel operation mode, the B-Go signals are generated by:

- external B-Go input
- orbit signal
- VME write to specific address

For each channel, the internal B-Go signal starts a transmission cycle. After a programmable delay (multiple of clock period), a signal called Inhibit<16..1> is generated with a programmable duration. The actual B-channel transmission starts at the end of the Inhibit signal. The B-Go to Inhibit delay is set by a 0 - 100 µs programmable timer (12-bit) and the Inhibit duration by a 0 - 6.4 µs timer (8-bit).

The transmission cycles can be single or double. In a single cycle, the next item in the FIFO is transmitted at the end of the Inhibit signal. In a double cycle, at the end of the first transmission a second transmission cycle is automatically generated. After a second delay (counted from the end of the first transmission), another Inhibit is generated (with programmable duration). The transmission of the next item in the FIFO starts at the end of the second Inhibit. Double cycles are required for the operation of the ECAL calibration system.

Each Inhibit signal is assigned a different priority level, such that Inhibit<1> has higher priority than Inhibit<2>, etc. When an Inhibit signal becomes active, the transmission of any command associated with a lower priority Inhibit is allowed to complete. Further such commands are held off until the higher priority one has been sent. Since the Inhibit signals are always programmed to have a duration exceeding that required for the transmission of even a long-format cycle (about 1.05 μs), the higher priority signal is always transmitted at a determinate time relative to the LHC orbit.

By appropriate programming, one can ensure that synchronous commands do not overlap and are always sent at well defined times.

Sequences of B Channel cycles can be generated by loading the FIFO's with several command/data. For each new B-Go signal the next item in the FIFO is transmitted. The B-Go channels can be programmed in repetitive or non-repetitive mode. In repetitive mode, the sequence of commands/data stored in the FIFO is repeated continuously by resetting the read pointer of the FIFO as soon as it is empty. In non-repetitive mode, the sequence is transmitted only once. The cycles in a sequence can be single or double.

When using VME initiated B-Go cycles two options are available: a) single transmission or b) block transfer. In single transmission, the next item in the FIFO is transmitted at VME write in a specific address. In block transfer, a sequence of transmission cycles is initiated by a VME write in a specific address. The sequence is stopped when the whole FIFO content is transmitted. This last option is intended to load electronics configuration parameters through the TTCrx.

A special code "No-operation" can be inserted in the FIFO content, in order to program a prescaling of commands initiated by the external B-Go or by the Orbit signals. No actual B channel transmission will occur when this code is found.

The VME access to load the FIFOs is D32 and supports block transfer mode.

In summary, each B-Go channel may be configured accordingly to the following options:

a) B-Go signal generation: External, Orbit or VME write
b) Inhibit signals: delay and duration
c) Cycle type: single or double cycle
d) Sequence mode: repetitive or non-repetitive
e) VME B-Go: single transmission or block transmission

## Generation of Channel-A control sequences

In order to match the requirements of the on-detector electronics control of the pixel, tracker, preshower and ECAL subdetectors, the fast commands generated in the B-Go channels are optionally encoded as sequences of three pulses in channel-A.

When this option is selected for a given channel, a sequence of three bits at 40 MHz is transmitted to channel-A at the end of the Inhibit signal (first Inhibit in case of double cycles). The 3-bit sequence is programmable for each B-Go channel. Typically, these sequences encode the following commands (codes are given as example):

1. L1A trigger (100)
2. BC0 (101)
3. Test Enable (110)
4. Resync (111)

Care must be taken to avoid overlap between channel-A control sequences and L1A triggers. By construction, the commands BC0, Test Enable and Resync generated by the Trigger Controller (central or local) do not overlap with L1A.

When internal trigger generation is used, triggers are blocked during the transmission of a control sequence over channel-A.

## Asynchronous commands initiated by VME control

Asynchronous cycles may be initiated by writing the required data (a single byte for short-format or two 16-bit words for long-format) to specified TTCci VME addresses. In this mode the B-Go FIFOs are not used. Normally short-format cycles are used for broadcast commands or data while long-format cycles are used for individually addressed commands or data. However, a broadcast of 16 bits of data can be made with long-format cycles if TTCrx address 0 is chosen.

## Internal Trigger generation

The TTCci has the following options for internal Trigger generation:

a) Synchronous to Orbit signal:

– programmable delay between Orbit signal and L1A

– prescaling: L1A generated every n-th orbit

b) Random triggers at a given average rate (trigger rules are observed)

## Trigger Rules

TTCci implements the Trigger Rules defined in Section 3.8.

## Internal counters

The TTCci has internal Event Number and Orbit Number counters. For test purposes, the value of one of these counters can be transmitted by a long command immediately after L1A

# 6. sTTS Fast Merging Module

## 6.1 Overview

The Fast Merging Module (FMM) is the building block that allows to construct the sTTS tree. Its purpose is to perform logical operations on 32 groups of the 4 TTS binary signals. The result of the functions is available on 2 outputs (4 TTS lines each). The result of these operations can be summed again in a next module. Finally, the result of the sum will constitute the input to the central and local Trigger Controllers (one per partition).

## 6.2 Input/Output

The interfaces of the FMM are as follows:

- 32 * 4 bits LVDS differential inputs
- 2 * 4 bits LVDS differential output (one to Global TCS, one to Local TCS)
- TTC interface
- Control interface (Ethernet)

## 6.3 Logical operations

The logical operations performed on the input signals are programmable. Two options are available:

- logical OR
- majority with programmable level

The input signals are sampled with a 40 MHz clock. Dedicated logic is used to prevent spurious states on the signal transitions.

## 6.4 Signal History

Apart performing the logical functions, the FMM stores an history of the input signals. The history can be readout through the board control path.

When a transition occurs on any of the inputs (sampled with the bunch clock of 40MHz), all the input are saved in a circular memory along with an absolute time stamp provided by the bunch counter (12 bits) and the orbit counter (32 bits). A common reset to the counters in several FMMs is provided by the Trigger Controller.

The depth of the history will depend on the size of the memory and the number of transitions on the input signals.

## 6.5 Form Factor & Control

The FMM is implemented as a 1U module rack mounted. Input and output TTS signals use RJ45 connectors.

## Glossary

| | |
|---|---|
| aTTS | asynchronous TTS |
| BC0 | Bunch Crossing Zero |
| B-Go | signal starting transmission on TTC B-channel |
| BST | Beam Synchronous Timing |
| BU | Builder Unit |
| DCS | Detector Control System |
| DqD | DAQ Doctor |
| DSN | DAQ Services Network |
| ECR | Event Counter Reset |
| EVM | Event Manager |
| FDL | Final Decision Logic |
| FED | Front End Driver |
| FMM | Fast Merging Module |
| FRL | Frontend Readout Link |
| FU | Filter Unit |
| GTP | Global Trigger Processor |
| LTC | Local Trigger Controller |
| L1A | L1 Trigger Accept |
| OCR | Orbit Counter Reset |
| PTC | Partition Trigger Control |
| RCS | Run Control System |
| RU | Readout Unit |
| sTTS | synchronous TTS |
| TCS | Trigger Control System |
| TTC | Timing and Trigger Control |
| TTCex | TTC encoder and transmitter |
| TTCcf | TTC clock fanout |
| TTCci | TTC CMS interface |
| TTCmi | TTC machine interface |
| TTCrx | TTC receiver chip |
| TTCvi | TTC VME interface |
| TTS | Trigger Throttling System |

## References


[1] W. Smith, “CMS Synchronization Workshop Conclusions”, CMS IN-1999/016.

[2] J. Varela, “Timing and Synchronization in the LHC Experiments”, proceedings of the 6th Workshop on Electronics for LHC Experiments, Krakow, September 2000, CERN 2000-010, CERN/LHCC/2000-041.

[3] Ph. Farthouat, P. Gallno, “TTC-VMEbus Interface TTCvi”, RD12 Working Document.

[4] J. Christiansen, A. Marchioro, P. Moreira, TTCrx Reference Manual, RD12 Working Document.

[5] B. Taylor, “TTC Distribution for LHC Detectors”, IEEE Trans. Nuclear Science, Vol. 45, 1998.

[6] B.G. Taylor, LHC machine Timing Distribution for the Experiments, proceedings of the 6th Workshop on Electronics for LHC Experiments, Krakow, September 2000, CERN 2000-010, CERN/LHCC/2000-041.

[7] B.G. Taylor, TTC machine interface (TTCmi) User Manual, http://ttc.web.cern.ch/TTC/intro.html

[8] CMS DAQ Technical Design Report (to be published).

[9] J-J. Savioz, "TTCbi, a Timing, trigger and Control Interface for the Beam Synchronous Timing System" (draft document). http://ttc.web.cern.ch/TTC/intro.html