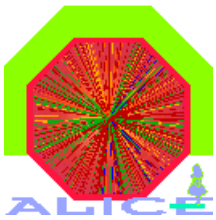


# ALICE Central Trigger Processor

## USER REQUIREMENT DOCUMENT

Draft 0.1 1 June 2000
Draft 02 1 December 2000
Draft 0.3 1 February 2001

**Draft 1.0 24 October 2001**

# DOCUMENT HISTORY

| Version | Date | Editor | Comments |
|---|---|---|---|
| Draft 0.1 | 01.06.00 | P. Jovanovic | First draft version |
| Draft 0.2 | 01.12.00 | P. Jovanovic | Numerous changes, following the CTP Workshops in Birmingham (July 2000) and in CERN (October 2000). Requirement numbering altered. |
| Draft 03 | 01.02.01 | P. Jovanovic | After the third CTP Workshop (December 2000, CERN), major changes in the following sections:<br>• *The generation of the L2a message,*<br>• *The L1 trigger-type word,*<br>• *Transmission of the trigger outputs,*<br>• *Region of Interest,*<br>• *Requirements on the sub-detectors.* |
| Draft 1.0 | 24/10/01 | D. Evans | Following extended requirements, the main changes from version 0.3 are:<br>• A large increase in trigger inputs and<br>• A very large increase in the number of trigger classes from 12 to 50.<br>In order for the CTP to cope with these changes the following have also been introduced:<br>• Sub-detector clusters and<br>• A CTP dead-time, after L0, of up to 2 μs. |

# TABLE OF CONTENTS

1.0 INTRODUCTION ..... 1
1.1 Purpose and scope of the document ..... 1
1.2 User profiles ..... 1
1.3 Document overview ..... 1
1.4 Acknowledgement ..... 1
1.5 Definitions and acronyms ..... 1
1.6 References ..... 2
2.0 GENERAL DESCRIPTION ..... 4
2.1 Overview ..... 4
2.2 CTP interfaces ..... 5
2.3 Other requirements ..... 6
3.0 SPECIFIC REQUIREMENTS ..... 7
3.1 Performance requirements ..... 7
*3.1.1 System capacity ..... 7*
*3.1.2 Sub-detector clusters ..... 7*
*3.1.3 CTP Dead-time ..... 7*
*3.1.4 Trigger input signals ..... 7*
*3.1.5 Trigger input synchronisation and alignment ..... 8*
*3.1.6 Random and periodical triggers ..... 9*
*3.1.7 Software trigger ..... 9*
*3.1.8 Calibration trigger ..... 10*
*3.1.9 Bunch-crossing mask bits ..... 11*
*3.1.10 BUSY inputs ..... 11*
*3.1.11 Programmable mask ..... 12*
*3.1.12 Interaction signal ..... 12*
*3.1.13 Past-future protection ..... 13*
*3.1.14 Trigger classes ..... 14*
*3.1.15 Test class ..... 14*
*3.1.16 Pre-scalers ..... 15*
*3.1.17 The CTP tasks ..... 15*
General requirements ..... 15
The generation of the L0 class-trigger ..... 16
The generation of the L1 class-trigger ..... 17
The generation of the L2 class-trigger ..... 17
The generation of the L2a message ..... 18
*3.1.18 The CTP readout ..... 18*
L2a messages ..... 18
Recording of interactions ..... 19
*3.1.19 Scalers ..... 20*
*3.1.20 On-line monitoring of the CTP ..... 21*
*3.1.21 The CTP outputs ..... 22*
Pre-pulse ..... 22
The *L1 trigger-type word* ..... 22
The *L2a message* and the *L2r word* ..... 24
Transmission of the trigger outputs ..... 24

*3.1.19 Region of Interest ..... 26*
*3.1.20 Event identification ..... 30*
*3.1.21 General ..... 31*
3.2 Control requirements ..... 31
3.3 Monitoring requirements ..... 32
3.4 Test requirements ..... 33
3.5 Maintenance requirements ..... 34
3.6 Requirements on the external system connections to the CTP ..... 34
*3.6.1 Requirements on the sub-detectors ..... 34*
*3.6.2 Requirements on the DAQ system ..... 36*
*3.6.3 Requirements on the CTP RORC ..... 37*
*3.6.4 Requirements on the run-control and DCS ..... 37*

# 1.0 INTRODUCTION

## 1.1 Purpose and scope of the document

## 1.2 User profiles

The document is intended for:

- the electronics experts who will design, build and commission the system, and make upgrades, if necessary;
- the hardware experts who will maintain the system *in situ* during the life of the experiment;
- the software experts who will develop and maintain the code necessary to operate - the equipment in the experiment; the software includes diagnostic and test programs, calibration, control and monitoring programs, as well as off-line programs;
- the ALICE trigger experts who will define the trigger selection criteria;
- the shift operators who will interact with the system via the run-control;
- the physicists who will use the data collected by the CTP, including the data read out from the CTP.

## 1.3 Document overview

The present section (***Introduction***) gives a brief account of the purpose and scope of the document, explains its structure, defines terms and acronyms and lists cited documents. The following section (***General Description***) gives a "plain-English" description of the functional requirements of the CTP. The final section (***Specific Requirements***) sets out, in a more formal way, requirements on the CTP and on external systems to which it is connected.

## 1.4 Acknowledgement

## 1.5 Definitions and acronyms

**BC** Bunch Crossing (clock) - the 40.08 MHz clock, locked to the LHC machine, used to synchronise the pipeline processing system.

**BUSY** Signal formed by a sub-detector to indicate that it cannot accept another **L0** trigger.

**CTP** Central Trigger Processor - electronic system that receives inputs from ALICE trigger sub-detectors and generates, in each bunch crossing, **L0**, **L1** and **L2** yes/no trigger decisions for all sub-detectors.

**DAQ** Data Acquisition (system).

**DCS** Detector Control System.

**DDL** Detector Data Link - an optical link between sub-detector front-end electronics and the RORC (DAQ).

**L0** Level-0 trigger (signal).

**L1** Level-1 trigger (signal).

**L2** Level-2 trigger (signal).

**L2a** Level-2 accept signal.

**L2r** Level-2 reject signal.

**LTC** Local Trigger Crate - a VME crate, part of the sub-detector electronics; contains the local system processor, the LTU, the TTC boards *etc*. .

**LTU** Local Trigger Unit (board).

**NRZ** Non Return to Zero encoding - one bit of data transmitted per clock cycle, with 1 or 0 level maintained during the entire cycle.

**RoI** Region of Interest (option, logic, data).

**RORC** Read-out Receiver Card.

**TRD** Transition Radiation Detector (source of **RoI** data).

**TTC** Timing, Trigger and Control (system) [6][7].

**TTCrx** Timing, Trigger and Control Receiver ASIC [10].

**TTCvi** TTC -VMEbus interface board [11].

## 1.6 References

[1] S. Quinton et al., *A Quality System for the LHC Experiment Electronics*, Proceedings of the First Workshop on Electronics for LHC Experiments, Lisbon, September 11-15, 1995, pp. 12-16.

[2] *ISO 9001, Quality Systems*, published by BSI and other national standards bodies.

[3] *Software Engineering Standard ESA PSS-05*, ISBN 0-13-106568-8.

[4] *ATLAS Level-1 Central Trigger Processor: User Requirements Document*, (Version 3.0), ATLAS working document ATL-DA-ES-0003, available on the EDMS system.

[5] *ATLAS Level-1 Central Trigger Processor: Technical Specification*, (Version 0), ATLAS working document ATL-DA-ES-0006, available on the EDMS system.

[6] B.G. Taylor, *TTC Distribution for LHC Detectors*, IEEE Trans. Nuclear Science, Vol. 45, June 1998, pp. 821-828.

[7] TTC web site: http://www.cern.ch/TTC/intro.html.

[8] A. Bhasin et al., *New Developments for the ALICE Trigger*, Proceedings of the Fifth Workshop on Electronics for LHC Experiments, Snowmass, Colorado, September 20-24, 1999, pp. 285-289.

[9] B.G. Taylor, *TTC machine interface (TTCmi) User Manual*, Rev 1.0, available on the TTC web site: http://www.cern.ch/TTC/intro.html.

[10] J. Christiansen et al., *TTCrx Reference Manual*, Version 3.0, available on the TTC web site: http://www.cern.ch/TTC/intro.html.

[11] Ph. Farthouat et al., *TTC-VMEbus Interface (TTCvi-MkII),* Rev 1.6, available on the TTC web site: http://www.cern.ch/TTC/intro.html.

# 2.0 GENERAL DESCRIPTION

## 2.1 Overview

The role of the CTP is to combine information from the ALICE trigger sub-detectors (trigger inputs) and the read-out sub-detectors (calibration requests, **BUSY**), and to form, in each BC interval, yes/no trigger decisions for levels **L0**, **L1** and **L2**. The decisions are sent to the TTC system, which distributes them to the front-end electronics of the sub-detectors, where they initiate data taking and control the readout. The TTC also receives from the CTP and broadcasts to the front-ends the information about the trigger type, the event identifier and the list of participating sub-detectors.

The algorithms used to form different types of triggers are implemented through trigger-class definitions. *Trigger class* is a basic logic block of the CTP and the hardware provides for a number of classes that are processed in parallel. All trigger-class parameters are independently and fully programmable.

The sub-detectors will be grouped into sub-detector clusters, each with its own past-future protection period.

The trigger-class definition contains the sub-detector cluster and the required patterns of the trigger inputs for levels **L0**, **L1** and **L2**. Internally generated random, periodical and software triggers could also be included in the **L0** pattern.

The **L0** class-trigger generation may be inhibited by a bunch-crossing mask which reflects the bunch "quality" (full, empty, noisy, etc.), a programmable mask, a veto derived from **BUSY** logic, and by the status of the past-future protection logic; the **L0** class-trigger could also be pre-scaled. The **L1** and **L2** decisions, on the other hand, may be inhibited only by the past-future protection.

The **BUSY** logic synchronises the CTP operation with the sub-detector front-end electronics and the DAQ read-out chain. The asserted **BUSY** signal indicates that the corresponding sub-detector is not ready to accept another event (e.g. buffers in the front-end are full) and inhibits further **L0** triggers. *The CTP can generate its own dead time after each a L0 trigger that will last for up to 2$\mu$s.*

The past-future protection logic keeps track of the time since the last significant interaction and determines, for each sub-detector cluster, whether sufficient time has elapsed to avoid pile-up in any of the cluster sub-detectors (and its consequences on the pattern recognition).

The reasons for having three trigger levels (**L0**, **L1** and **L2**) are explained elsewhere [8]. The trigger generation logic is discussed in detail in the following section (***Specific Requirements***).

In addition to generating the trigger signals, the CTP provides the DAC with the list of participating sub-detectors for all the **L2**-selected events, and a complete record of bunch interactions. The data are archived and can be used for off-line checking of the trigger system, monitoring of experimental conditions such as beam quality, and for calculation of trigger efficiency; the interaction data should significantly simplify the event reconstruction process.

The CTP includes a number of scalers that are used to measure trigger rates and monitor dead time. The scalers are read in regular intervals, the data are processed in real time and the results are written to a permanent storage to be used by the off-line analysis. Any potential problem is reported to the run-control.

The CTP also contains diagnostic, "snap-shot" memories that sample, at full speed, trigger inputs and outputs as well as the data at various points in the processing chain. The data are used for the on-line monitoring of the CTP performance and for the luminosity assessment of individual beam bunches.

## 2.2 CTP interfaces

The interfaces between the CTP and external systems are shown on the context diagram (**Figure 2.2.1**).

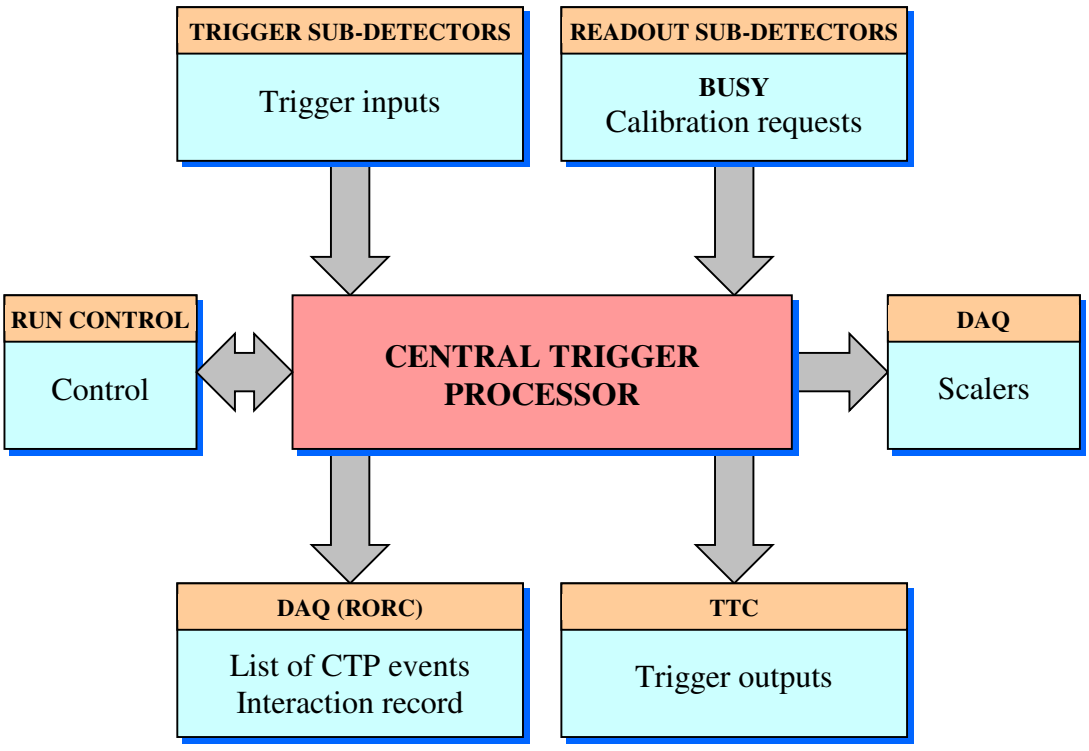


**Figure 2.2.1** Context diagram for the CTP

The trigger inputs related to the same bunch crossing, but generated by different trigger sub-detectors, will generally arrive at the CTP at different

times. The CTP therefore has to include programmable elements that delay the signals that arrive early to be aligned in time with those that arrive last.

Calibration requests enable sub-detectors to demand calibration triggers during the physics run (for stand-alone tests, the sub-detectors can introduce their own triggers at the level of their local TTC system). The CTP also provides a *pre-pulse* facility used to generate calibration inputs prior to calibration triggers.

The link to the control system is used to configure the CTP and perform tests. It also carries error and warning massages which emerge from real time analysis of data coming from scalars and diagnostic memories.

Apart from information sent to the DAQ for all **L2**-selected events, the CTP also supplies, in regular intervals, the data from scalers. The mechanism of data transmission is still under discussion.

The **L0** trigger is the only signal transmitted via a dedicated, low propagation delay cable; the TTC system is used to distribute all the other trigger outputs to the front-end electronics of the sub-detectors.

## 2.3 Other requirements

A very important requirement for all parts of the CTP system is that the latency should be as small as possible. This is particularly critical for the generation of **L0** and **L1** trigger, where the latency should be less than 100 ns.

There are numerous requirements concerning the operation of the CTP. They relate to control, monitoring that the CTP is working as specified, test and diagnostic facilities, reliability and maintenance. Those requirements are discussed in detail in the following section (***Specific Requirements***).

# 3.0 SPECIFIC REQUIREMENTS

## 3.1 Performance requirements

### 3.1.1 *System capacity*

**1.1.1** The CTP shall be able to accommodate up to **20** readout sub-detectors.

Note 1: The number includes a spare capacity for possible expansions during the lifetime of the ALICE experiments.

### 3.1.2 *Sub-detector clusters*

1.2.1 The CTP shall form up to **6** programmable sub-detector clusters plus one internal cluster for software triggers.

Note 1: Each sub-detector cluster may consist of any combination of sub-detectors from just one sub-detector to all sub-detectors.

1.2.2 The sub-detector cluster shall be used to determine the selection of a past-future protection circuit.

### 3.1.3 *CTP Dead-time*

1.3.1 The CTP shall have a dead-time of up to **2μs** after the **L0** trigger.

### 3.1.4 *Trigger input signals*

**1.4.1** The CTP shall accept and process:

- up to **24** trigger inputs participating in the **L0** decision;
- up to **20** trigger inputs participating in the **L1** decision;
- up to **6** trigger inputs participating in the **L2** decision.

Note 1: A list is of all available, potential and spare inputs for the trigger levels **L0**, **L1** and **L2** is given in appendix A (to follow).

**1.4.2** All trigger inputs, the **L2** including, shall be generated with the *time precision of a single bunch crossing*.

Note 1: In exceptional cases, trigger inputs with somewhat lower time precision might also be acceptable. Those signals shall remain asserted during the full length of their uncertainty interval.

**1.4.3** The trigger inputs shall have the NRZ format, be synchronous with the sub-detector BC clock and have the maximum edge jitter of **2 ns**.

Note 1: The signal levels and the details of the CTP inputs (cable type, connectors, pin assignments, polarity, etc.) must be specified before the Preliminary Design Review.

**1.4.4** The trigger sub-detectors shall provide the cables and shall be responsible for the connection of their trigger signals to the trigger inputs of the CTP system.

### 3.1.5 *Trigger input synchronisation and alignment*

**1.5.1** The CTP shall synchronise all trigger inputs with the local BC clock, and align them in time so that data presented to the processor logic correspond to a common bunch crossing.

The synchronisation will be automatic; during the process, the trigger sub-detectors shall be required to continuously transmit a pattern of triggers in alternate bunch-crossing intervals (local **BC** clock divided by two - see **Figure 3.6.1**).

**1.5.2** The timing alignment system shall allow for differences of up to $\mathbf{t_{L0}}$ ns between different **L0** trigger inputs.

Note 1: The maximum value of $\mathbf{t_{L0}}$ is to be decided.

**1.5.3** The timing alignment system shall allow for differences of up to $\mathbf{t_{L1}}$ ns between different **L1** trigger inputs.

Note 1: The maximum value of $\mathbf{t_{L1}}$ is to be decided.

**1.5.4** The timing alignment system shall allow for differences of up to $\mathbf{t_{L2}}$ ns between different **L2** trigger inputs.

Note 1: The maximum value of $\mathbf{t_{L2}}$ is to be decided.

Note 2: The **L2** decision involves both the **L2** trigger inputs and the past-future protection status bits.

If the end of the longest past-future protection interval is later than the last of the **L2** trigger inputs, additional delay will be required in order to align the **L2** trigger inputs with the end of the interval.

If the end of the past-future protection is earlier than the last of the **L2** inputs, the **L1** class-triggers will have to be delayed in order to align the end of the interval with the **L2** inputs.

Both possibilities need to be discussed and the appropriate delay requirements implemented.

**1.5.5** There shall be a strategy for timing-in the CTP together with the trigger sub-detectors and the TTC system, for beam-beam, cosmic ray and test trigger operation.

Note 1: The strategy for timing-in the system needs to be discussed between all groups working on the trigger system.

Note 2: It must be possible to time-in the CTP without relying on the physical access to the electronics (e.g. one must not rely on being able to access the system with a scope).

### 3.1.6 *Random and periodical triggers*

**1.6.1** The CTP shall generate internally **2** random-trigger signals with independently programmable rates in the range of **0.1-100 kHz**. The signals can be combined with other trigger conditions, or used inclusively.

**1.6.2** The CTP shall generate internally **2** pre-scaled **BC** clock signals with independently programmable pre-scale factor in the range of **1-16·10$^6$** (24 bits). The signals can be combined with other trigger conditions, or used inclusively.

Note 1: The pre-scale factor must not be a multiple or a sub-multiple of the number of bunches in an LHC orbit (3564) if one wants to sample all bunches equally.

### 3.1.7 *Software trigger*

**1.7.1** In addition to the trigger inputs generated by the trigger sub-detectors, there shall be a *software-trigger* input generated by the CTP control processor.

**1.7.2** There shall be two options for the *software-trigger*:

- *an asynchronous software-trigger* shall be generated at the moment when the trigger request is issued;
- *a synchronous software-trigger* shall be generated at a selected bunch-crossing; the bunch-crossing selection shall be programmable.

**1.7.3** A hardware flag shall be used to synchronise the CTP control processor action with the status of the software-trigger logic:

- the trigger request shall be issued only when the flag is cleared;
- the flag shall be set by the trigger request;
- the flag shall be cleared by the trigger hardware when an attempt is made to generate the trigger;
- the CTP control processor may re-program the *test-class* description (see section ***Test class***) only when the flag is cleared.

**1.7.4** There shall be a special trigger class, *test class*, dedicated to the *software trigger*. The programmable description of the *test class* (see section ***Test class***) shall include the list of participating sub-detectors and the selection of the

past-future protection circuit; the only **L0** class-trigger option for the *test class* shall be the *software-trigger*; neither **L1** nor **L2** trigger inputs shall be used.

**1.7.5** The CTP shall *attempt* to generate a trigger in response to a request set by the CTP control processor.

Vetoes, derived from **BUSY** logic, may inhibit the **L0** class-trigger generation; the past-future protection logic may prevent some **L0**, **L1** or **L2** class-triggers. Failed attempts shall not be automatically repeated; in order to generate the trigger, the CTP control processor must try again by issuing another request.

**1.7.6** Success or failure of trigger attempts shall be monitored by the CTP logic and the outcome shall be reported to the CTP control processor.

### 3.1.8 *Calibration trigger*

**1.8.1** During the physics run, the sub-detectors may demand a *calibration trigger*, or a burst of triggers, by sending a request to the CTP control processor.

Note 1: The format and the content of the request message are to be decided; the message must contain the *Readout Mode* data (4 bits, see section ***The L1 trigger-type word***).

Note 2: For stand-alone tests, the sub-detectors can introduce their own triggers at the level of their local TTC system.

Note 3: The role, the choice and the location of the CTP control processor, and the communication between the sub-detectors and the processor are to be discussed.

**1.8.2** A process, run by the CTP control processor, shall queue the requests and arbitrate request priorities.

Note 1: Since the priority arbitration shall be performed by the software, changes or modifications of the algorithm should be easy and quick; *'round robin'* priority allocation appears to satisfy current requirements.

**1.8.3** The CTP control processor shall re-program the *test-class* description (see section ***Test class***) according to the pending calibration-trigger request, and generate a *synchronous software-trigger* in the *calibration bunch-crossing interval*.

The *calibration bunch-crossing* shall *always* be located inside the *long LHC gap*. Apart from the calibration trigger, no other trigger class shall ever be generated during the interval. This enables both the sub-detector front-ends and the DAQ to unambiguously recognise the calibration-trigger by checking the bunch number in the event identifier.

The *calibration bunch-crossing interval* shall be a system constant; a possible change requires coordinated modifications in the configuration files for the CTP, the DAQ and the sub-detector front-ends.

**1.8.4** There shall be only one sub-detector involved in any calibration-trigger.

**1.8.5** Following the request, the software-trigger hardware shall attempt to generate the calibration-trigger during the next large LHC gap.

**1.8.6** Success or failure of trigger attempts shall be monitored by the CTP logic and the outcome shall be reported to the requesting sub-detector by the CTP control processor.

**1.8.7** Each calibration-trigger *attempt* shall be preceded by a calibration **Pre-pulse**.

Note 1: Vetoes derived from **BUSY** logic and the past-future protection may inhibit calibration-trigger generation (see section ***Software trigger***). *A calibration* **Pre-pulse***, on the other hand, shall be sent in response to each trigger request.*

**1.6.8** For the calibration trigger, the *Trigger-type* flag in the ***L1 trigger-type word*** shall be asserted (see section ***The L1 trigger-type word***).

### 3.1.9 *Bunch-crossing mask bits*

**1.9.1** The CTP shall generate internally **4** bunch-crossing mask bits and, for each bit, an arbitrary group of bunch crossings can be selected for which the mask shall be set.

Note 1: The bunch-crossing mask bits can be used in combination with other conditions to restrict certain trigger requirements to a subset of bunch crossings (empty, full, noisy, etc.).

Note 2: The bunch-crossing mask bits shall be used for the generation of synchronous software-triggers (see section ***Software trigger***).

### 3.1.10 *BUSY inputs*

**1.10.1** The CTP shall receive an external **BUSY** signal from each sub-detector. The signal shall be used to form the **L0** trigger vetoes for all trigger-classes in which the corresponding sub-detector participates.

**1.10.2** The **CTP BUSY** signal (see section ***The CTP readout***), from the CTP readout channel, shall be used to form the **L0** trigger veto *for all trigger classes*.

**1.10.3** The **DAQ BUSY** signal, from the DAQ system, shall also be used to form the **L0** trigger veto *for all trigger classes*.

**1.10.4** The CTP shall not send an **L0** trigger to a sub-detector while the corresponding **BUSY** signal is asserted. *The veto shall become active from the moment the signal reaches the CTP decision-making logic.*

Note 1: The **BUSY** set-up time is to be specified.

**1.10.5** The status of **BUSY** signals shall not affect the generation of the **L1** and **L2** signals that have already passed the **L0** stage.

**1.10.6** After an **L0** trigger, *regardless of the status of the* **BUSY** *signal*, the CTP shall not send another **L0** to the corresponding sub-detector until the first bunch-crossing following the **L1** decision.

**1.10.7** During the initialisation phase, the **BUSY** signals shall be asserted until the initialisation is complete.

**1.10.8** The **BUSY** input shall automatically be asserted ("*busy*" status) if the corresponding sub-detector electronics is switched off, or if the cable is disconnected.

**1.10.9** The sub-detectors shall provide the cables and connect the **BUSY** inputs to the CTP system.

Note 1: The signal levels and the details of the CTP inputs (cable type, connectors, pin assignments, polarity, etc.) must be specified before the Preliminary Design Review.

Note 2: The introduction of the LTU board is likely to cancel the requirement (see section ***Transmission of the trigger outputs***).

### 3.1.11 *Programmable mask*

**1.11.1** It shall be possible to mask any subset of trigger classes using a *programmable mask word.*

### *3.1.12 Interaction signal*

1.12.1 The CTP shall be able to handle two definitions of interaction, **INT** and $\mathbf{INT_2}$. The purposes of the **interaction** definition (**INT**) are that it is used to activate past-future protection circuits and for the interaction record written every orbit.

Note 1: Only **INT** shall be used for the interaction record and both **INT** and $\mathbf{INT_2}$ shall be used by the past-future protection circuits.

Note 2: The default definition for **INT** will be $\mathbf{V0_{min.\ bias}}$. For physics reasons, the TRD wake-up (at ~ 100 ns) should use the same definition for **INT** as the CTP. The TRD should, therefore, install inputs from any detector likely to be used in the definition of **INT**.

1.12.2 The **interaction** signal shall be a programmable logic function of a subset of **4** out of all **L0** trigger inputs. For test purposes, the random and the periodical triggers (see section ***Random and periodical triggers***) could also be used as **Interaction.**

1.12.3 The **Interaction test** signal shall also be a programmable logic function of the same subset of **L0** trigger inputs. The signal shall be used only for the on-line tests (rate monitoring).

### 3.1.13 *Past-future protection*

**1.13.1** The CTP shall include the logic to check the violation of the past-future protection. Both the duration of the protection interval and the number of permitted interactions during the interval shall be programmable.

The maximum duration of the protection interval shall be **±100 μs** with a step size of about 1% and a jitter of less than 1%.

The maximum number of permitted interactions shall be **32.**

Note 1: No interaction during the protection interval might be required for calibration; single interaction is a normal requirement; in certain modes, some sub-detectors could tolerate several interactions.

1.13.2 The past-future protection circuits shall have two interaction inputs, **INT** and $\mathbf{INT_2}$, and shall use either one or both to set up the past-future protection.

Note 1: The past-future protection circuits shall be able to differentiate between low multiplicity and high multiplicity interactions so that, for example, several low multiplicity interactions could be allowed in the past-future protection period but no other type of interaction. The second type of interaction, used only for this purpose, shall be defined as $\mathbf{INT_{low} = V0_{min.\ bias} \cdot V0_{semi\ cen}}$.

Note 2: It should be noted that, for physics reasons, $\mathbf{INT_{low}}$ ($\mathbf{INT_2}$) must always be a subset of **INT**.

**1.13.3** There shall be **4** independently programmable past-future protection circuits; all will allow multiple interactions.

Note 1: A circuit can be used simultaneously for any number of classes.

**1.13.5** Each past-future protection circuit shall generate in each bunch-crossing interval the protection status bits, required for **L0**, **L1** and **L2** trigger decisions.

### 3.1.14 *Trigger classes*

**1.14.1** The CTP shall form **50** independently programmable "physics" trigger classes. The trigger class shall be the basic processing structure throughout the CTP logic.

**1.14.2** The logic of all trigger classes shall only consist of **AND**s of trigger inputs.

**1.14.3** The programmable definition of a trigger class shall include:

- the sub-detector cluster (which will define the past-future protection circuit),
- the **L0** class-trigger pattern,
- the **L1** class-trigger pattern,
- the **L2** class-trigger pattern,
- the trigger-class pre-scaling factor.

**1.14.4** The **L0** class-trigger pattern shall define:

- the status (*1, 0*, or *"don't care"*) of all the **L0** trigger inputs,
- the status (*1* or *"don't care"*) of the random and periodic triggers,
- the selection of the bunch-crossing mask (including *"don't care"* option).

Note 1: The inclusion of "*0*" status in the definition of the **L0** class-trigger pattern is to be reconsidered.

**1.14.5** The **L1** class-trigger pattern shall define the status (*1, 0*, or "*don't care*") of all the **L1** trigger inputs.

Note 1: The inclusion of "*0*" status in the definition of the **L1** class-trigger pattern is to be re-considered.

**1.14.6** The **L2** class-trigger pattern shall define the status (*1*, *0*, or *"don't care"*) of all the **L2** trigger inputs.

Note 1: The inclusion of "*0"* status in the definition of the **L2** class-trigger pattern is to be re-considered.

### 3.1.15 *Test class*

**1.15.1** On top of the normal, "physics" trigger classes (see section ***Trigger classes***), there shall be an additional, special trigger class - the *test class*.

Note 1: Apart from the calibration, the only other application foreseen so far for the *test class* is the synchronisation of sub-detector event identifications (bunch number, orbit number). Generation of the *test class* is not time critical and, during the physics run, its rate is going

to be very low. It is therefore sufficient to have only one dedicated class, shared by all the applications.

**1.15.2** The only **L0** class-trigger option for the *test class* shall be the *software trigger*; neither **L1** nor **L2** trigger inputs shall be used; the application of the past-future protection shall be the same as for normal classes.

**1.15.3** The *test class* shall have priority over any normal class and it shall always be the only class triggered in any given bunch-crossing interval.

Note 1: The complex definitions of all normal trigger classes are included in the configuration file for both CTP and the DAQ, and they remain unchanged during a run. The description of the test class, on the other hand, changes continuously since the class is shared among a number of applications. The fact that, in any bunch crossing, the test class is always the only active class, and that the corresponding readout data contain the list of participating sub-detectors (see section ***The generation of the L2a message***), enables the DAQ to adjust its operation "on the fly"- no other communication between the CTP and the DAQ is required.

**1.15.4** The *calibration trigger* is a special case of the *test class* trigger. It is triggered by a *synchronous software trigger*, generated in the *calibration bunch-crossing interval* (see section ***Calibration trigger***). There shall be only one sub-detector involved in any *calibration trigger*.

### 3.1.16 *Pre-scalers*

**1.16.1** The CTP shall include individually programmable pre-scalers for each **L0** class-trigger. These will pass every **N**th occurrence of the class-trigger, where **N** is a programmable number.

The maximum value of the pre-scaling factor **N** shall be **1,048,575** (20 bits).

Note 1: A pre-scaler is not required for the *test class*.

### 3.1.17 *The CTP tasks*

**General requirements**

**1.17.1** The **L0** decision shall be made, *with the minimum latency*, following the arrival to the CTP of the last of the **L0** trigger inputs.

The **L0** decision time shall be the same for all trigger classes, regardless of their class definition.

Note 1: The **L0** decision time is to be specified (the current estimate is 900 ns after the interaction).

**1.17.2** The **L1** decision shall be made, *with the minimum latency,* following the arrival to the CTP of the last of the **L1** trigger inputs.

The **L1** decision time shall be the same for all trigger classes, regardless of their class definition.

Note 1: The **L1** decision time is to be specified (the current estimate is 5.2 μs after the interaction).

**1.17.3** The **L1** decision time shall be programmable within the range xx μs to yy μs.

**1.17.4** The **L2** decision shall be made following the arrival to the CTP of the last of the **L2** trigger inputs, or at the end of the longest past-future protection interval, whichever is the latest.

The **L2** decision time shall be the same for all trigger classes, regardless of their class definition and regardless of whether the decision is positive (**L2a**) or negative (**L2r**).

Note 1: The **L2** decision time is to be specified (the current estimate is 88 μs after the interaction).

**1.17.5** The **L2** decision time shall be programmable within the range xx μs to yy μs.

**The generation of the L0 class-trigger**

**1.17.6** The CTP shall form an **L0** class-trigger for each trigger class whose **L0** class-trigger pattern definition (see section ***Trigger classes***) matches the current status of the **L0** trigger inputs, internally generated random and scaled-clock inputs, and the bunch-crossing masks.

Programmable mask, vetoes (derived from **BUSY** logic) and the **L0** past-future protection status bits may inhibit some or all **L0** class-triggers.

Pre-scale conditions (see section ***Pre-scalers***) may be applied independently to each trigger class.

The sub-detector **L0** triggers shall be obtained from the logic *OR* of the lists of participating detectors for all the triggered classes, after the application of inhibit conditions.

The **L0** latency shall be as short as possible and shall not exceed **100 ns**, measured from the time of arrival of the last trigger input to the CTP until the **L0** trigger appears at the output of the CTP.

Note 1: A “legitimate” **L0** class-trigger shall be rejected if it coincides with the *test-class* **L0 -** the test class has the priority and is required to be the only class in a bunch-crossing interval (see section ***Test class***).

**1.17.7** After a L0 trigger, the CTP shall have a dead time of up to **2 μs**.

**1.17.8** Apart from an occasional calibration trigger in the *calibration bunch-crossing interval* (see section ***Calibration trigger***), no other **L0** class-trigger shall be generated during the long LHC gap (119 bunch-crossing intervals, 2.975 μs).

Note 1: Following the injection of a calibration charge, some sub-detectors require time to recover and get ready for the next physics trigger. With the restriction in place, the sub-detectors can use the existing **BUSY** signal to prevent further **L0** triggers until the effects of charge injection have ceased (see section ***Requirements to the sub-detectors***).

Note 2: The restriction does not affect the LHC physics since there are no filled bunches during the gap; in case of a test run using cosmic triggers, the restriction introduces only a small percentage of dead time.

**The generation of the L1 class-trigger**

**1.17.9** The **L0** class-trigger signals shall be delayed in order to align them with the **L1** trigger inputs. The delay shall be programmable and equal for all trigger classes. The delay is determined by the arrival time of the last **L1** trigger input.

The maximum value of the programmable delay shall be **6.4 μs**.

**1.17.10** The CTP shall form a **L1** class-trigger for each class triggered at the **L0** level whose **L1** class-trigger pattern definition (see section ***Trigger classes***) matches the current status of the **L1** trigger inputs.

The **L1** past-future protection status bits may inhibit some of the **L1** class-triggers.

The sub-detector **L1** triggers shall be obtained from the logic *OR* of the lists of participating detectors for all the triggered classes, after the application of inhibit conditions.

The **L1** latency shall be as short as possible and shall not exceed **100 ns**, measured from the time of arrival of the last trigger input to the CTP until the **L1** trigger appears at the output of the CTP.

**The generation of the L2 class-trigger**

**1.17.11** The **L1** class-trigger signals shall be delayed in order to align them with both the **L2** trigger inputs and the end of the longest past-future protection interval.

The delay shall be programmable and its maximum value shall be **102.4 μs** (4096 **BC** periods).

**1.17.12** The CTP shall form an **L2** class-trigger for each class triggered at the **L1** level whose **L2** class-trigger pattern definition (see section ***Trigger classes***) matches the current status of the **L2** trigger inputs.

The **L2** past-future protection status bits may inhibit some of the **L2** class-triggers.

The sub-detector **L2a** (Level 2 accept) triggers shall be obtained from the logic *OR* of the lists of participating detectors for all the triggered classes, after the application of inhibit conditions.

The sub-detector **L2r** (Level 2 reject) signals shall be obtained from the logic *OR* of the lists of participating detectors for all classes triggered at the level **L1**, excluding the **L2a**-selected sub-detectors.

Note 1: In "plain English": all the sub-detectors that are sent the **L1** trigger shall receive either **L2a** or **L2r**.

**The generation of the L2a message**

**1.17.13** For each **L2a**-selected event, the CTP shall generate the ***L2a*** *message.*

**1.17.14** The **L2a** message shall include:

- the *Trigger-type* flag (set to 0 for physics triggers and 1 for software triggers; see section ***The L1 trigger-type word***);
- for software triggers: the *Calibration* flag (asserted for the *calibration triggers* and cleared for other types of *software triggers*);
- the *event identifier* (bunch number, orbit number; see section ***Event identification***);
- the *list* (1 bit per sub-detector) of *all participating sub-detectors* .

Note 1: The s*tatus word* provides unambiguous information about the status of all trigger inputs (levels **L0**, **L1** and **L2**) that participate in the trigger class decision; the status of trigger inputs declared as *"don't care"* remains undefined.

Note 2: The s*tatus word* provides unambiguous information about the list of sub-detectors that participate in the event.

Note 3: The format of the ***L2a*** *message* is to be defined in collaboration with the DAQ group.

## 3.1.18 *The CTP readout*

**L2a messages**

**1.18.1** The ***L2a*** *messages* (see the previous section) for all the **L2a**-selected events shall be retained in a derandomising buffer-memory and read-out to the RORC (*via* the standard readout link).

Note 1: The latency in transferring the data to the RORC is to be specified.

**1.18.2** The CTP shall not change the order of the messages sent to the RORC.

**1.18.3** The throughput of the readout system shall provide for the expected worst case of the CTP trigger rate.

Note 1: The necessary CTP read-out throughput is to be decided.

**1.18.4** The size of the derandomising buffer and the throughput of the system shall be sufficiently large to avoid, in normal operating conditions, the introduction of significant dead-time *via* the **CTP BUSY** mechanism.

Note 1: The acceptable level of dead-time due to the CTP operation is to be specified.

**1.18.5** The CTP shall generate the **CTP BUSY** signal if a derandomising buffer is *nearly full*.

Note 1: The criterion for determining when the buffer is *"nearly full"* is to be specified. It will depend on the maximum number, at any time, of triggers being simultaneously processed in the CTP logic.

**1.18.6** During the initialisation phase, the **CTP BUSY** signal shall remain asserted until the initialisation is complete.

**Recording of interactions**

**1.18.7** For *each* LHC orbit, a message - *Interaction record*, shall be sent to the DAQ system. The message shall contain:

- the orbit number (24 bits),
- a list of all instances of the **Interaction** signal during the orbit (bunch number - 12 bits, for each interaction)
- the *Transmission Error* indicator - see the following requirement**.**

Note 1: The record of interaction instances should help significantly the event reconstruction process. The expected maximum interaction rate is 10 kHz in Pb-Pb mode and 200kHz in p-p mode.

Note 2: The format of the *Interaction record* message is to be defined in collaboration with the DAQ group.

**1.18.8** The messages shall be retained in a derandomising buffer-memory of sufficient capacity and read-out to RORC (*via* standard read-out links).

A possible overflow of the buffer shall be indicated by assertion of the *Transmission Error* flag, also included in the message.

Note 1: A single data link to the DAQ could be used for transmission of both the *Interaction records* and the ***L2a*** *messages*, but the messages must keep separate derandomising buffers; a possible

overflow of the *Interaction record* buffer shall not affect the status of the **CTP BUSY** signal.

### 3.1.19 *Scalers*

**1.19.1** The CTP shall include scalers that will count continuously during a run. The frequency of scaler reading will probably be every two minutes and the output will be sent to the local monitoring processor. The scalers shall count the following:

- All the trigger input signals (**L0**, **L1** and **L2**),
- Two **interaction** and one **interaction test** signals,
- **INT Dead-Time-Bar** for all six detector clusters (needed to calculate cross-sections)

And for each trigger class:

- **L0** triggers before vetos etc.,
- **L0** triggers after vetos (scaling, busy, past-future etc.),
- **L1** triggers before past-future protection,
- **L1** triggers after past-future protection,
- **L2** trigger before past-future protection,
- **L2** triggers after past-future protection.

**1.19.2** The capacity of high rate scalers will be up to **32 bits**.

Note 1: Scalers should be able to read for at least **ten hours** under normal running conditions.

**1.19.3** The scalers shall reset at the beginning of a run and read out to the DAQ at the end of a run.

**1.19.4** The CTP shall also include scalers to count the number of bunch crossings during which the following signals are asserted:

- all the sub-detector **BUSY** signals,
- **CTP BUSY**,
- **DAQ BUSY**.

**1.19.5** The capacity of these scalers is to be decided.

Note 1: Scalers should be able to read for at least **two minutes** under normal running conditions.

**1.19.6** The scalers shall be read in regular intervals by the local monitoring processor. Real-time analysis of the data shall be performed, and any potential problem shall be reported to the run-control system. The data shall be written to

permanent storage in the run log and made available for use by the off-line analysis.

Note 1: The role, the choice and the location of the local monitoring processor are to be discussed.

Note 2: The interface to the run-control is to be decided.

### 3.1.20 *On-line monitoring of the CTP*

**1.20.1** The CTP shall contain diagnostic, "snap-shot" memories to capture data at various points in the CTP processing chain. The following signals shall be recorded:

- whatever signals are needed for CTP diagnostic tests and adequate on-line monitoring. These are still being considered by the CTP group.

**1.20.2** In normal operation, the data shall be stored in the diagnostic memories at full speed, in bunch-per-bunch mode, over the monitoring period. The data shall be read-out by the monitoring processor and the real-time analysis shall be performed.

Note 1: The duration of the monitoring period is determined by the capacity of the memory ICs. The capacity of the diagnostic memory is to be decided.

Example: a memory with a capacity of one million words could allow a monitoring period of 26.2 ms, or 294 full LHC orbits.

**1.20.3** In normal operation, the "snap-shot" data taking, followed by the analysis, shall be performed by the local monitoring processor in regular intervals throughout a run. Any potential problem shall be reported to the run-control system. The data shall be written to permanent storage in the run log and made available for use by the off-line analysis.

Note 1: The frequency of the "snap-shot" data taking is to be considered. The rate is strongly affected by the readout speed and the power of the local monitoring processor.

Note 2: The role, the choice and the location of the local monitoring processor are to be discussed.

Note 3: The interface to the run-control is to be decided.

Note 4: The data required for use by the off-line analysis are to be considered.

Note 5: The "snap-shot" data enable the on-line monitoring of the CTP logic by comparing the output generated by the system with the result predicted by the software model of the CTP.

Note 6: Bunch-per-bunch monitoring of the trigger inputs enables dynamic assessment of the "quality" (full, empty, noisy, etc.) and luminosity of individual bunches. The data is used as an input to the trigger decision logic (mask generation, choice of the trigger algorithm) and for the off-line analysis. The time required for achieving sufficient statistics needs to be assessed.

### 3.1.21 *The CTP outputs*

**1.21.1** For each sub-detector, the CTP shall generate:

- the calibration **Pre-pulse**;
- the **L0** and **L1** trigger signals;
- the ***L1** trigger-type word*;
- the ***L2a** message;*
- the ***L2r** word*.

**1.21.2** The **L0** and **L1** trigger outputs, and the calibration **Pre-pulse** shall have the NRZ format, be synchronous with the **BC** clock and have the maximum edge jitter of **2 ns**.

Note 1: The signal levels and the details of the CTP outputs (cable type, connectors, pin assignments, polarity, etc.) must be specified before the Preliminary Design Review.

**Pre-pulse**

**1.21.3** The CTP shall provide the **Pre-pulse** signal to be used for generation of calibration inputs (charge injection, etc.) prior to the calibration trigger.

**1.21.4** The **Pre-pulse** lead time in respect to the corresponding **L0** calibration trigger shall be programmable, but, at any time, it shall be the same for all the sub-detectors. *The precise delay adjustment shall be the responsibility of the sub-detectors, according to their individual requirements.*

**1.21.5** The maximum **Pre-pulse** lead time shall be **3.2 µs**.

Note 1: The **Pre-pulse** lead time is to be reconsidered.

**The *L1 trigger-type word***

**1.21.6** For each **L1** trigger, the CTP shall generate individual ***L1** trigger-type words* for all participating sub-detectors. The words shall be transmitted to the sub-detector front-end electronics synchronously with the **L1** trigger.

Note 1: The ***L1*** *trigger-type* word shall be transmitted to the sub-detector front-end electronics *via* the *channel B* of the TTC system. The channel is also used for transmission of a number of other signals, most of which are given higher priority, in order to preserve their synchronous nature. As a result, the word is likely to encounter additional delays and a transmission jitter of several microseconds.

The worst-case delay and the maximum jitter need to be evaluated.

**1.21.7** The ***L1*** *trigger-type word* shall be 8 bits wide (restriction imposed by the TTCvi board [11]); the word shall contain the following data:

- *Trigger-type* flag (1 bit);
- *Readout Mode* (4 bits);
- *Enable Segmented Readout* flag (1 bit);
- spare (2 bits, set to 0).

**1.21.8** The same *Trigger-type* flag shall be broadcast to all the sub-detectors that participate in the **L1** trigger. The state of the flag shall be set to:

- 0, for normal, physics triggers;
- 1, for all *software triggers* (including the calibration trigger).

**1.21.9** For physics triggers, the *Readout Mode* data (4 bits) shall be used to select the readout method of the front-end electronics. The bit pattern depends on the active trigger class(classes) and it is sub-detector specific; in all the cases, the content shall be set in agreement with sub-detector groups.

For the calibration triggers, the data shall be used to select a calibration option. The data shall be part of the calibration-request message and shall be defined by a sub-detector on a request-by-request basis.

**Note 1: Sub-detectors who wish to use the Readout Mode data must specify how they wish all combinations of 50 classes to be converted into a 4 bit word.**

**1.21.10** The same *Enable Segmented Readout* flag shall be broadcast to all the sub-detectors that participate in the **L1** trigger. The status of the flag shall be derived from the list of active trigger classes.

If the flag is cleared, the *Region of Interest* (**RoI**) option shall be disabled; the source of the ***RoI*** *data* (TRD sub-detector) shall not transmit the ***RoI*** *data* message; the sub-detectors that participate in the option should not expect to receive the message.

If the flag is set, the TRD shall generate the ***RoI*** *data* (see section ***Region of Interest***) and broadcast the message to the *sub-detectors that participate in the option*.

Note 1: The *Enable Segmented Readout* flag shall be automatically cleared for all the **L1** triggers that do not include the TRD sub-detector (the source of the ***RoI*** *data*).

Note 2: When the *Enable Segmented Readout* flag is set, the TRD is obliged to broadcast the ***RoI*** *data* message; but, even in that case, the TRD can still "suspend" a partial readout by appropriately setting the ***RoI*** *data.*

**1.21.11** For all the **L2a**-selected events, the corresponding ***L1*** *trigger-type word* shall be attached to the raw data before they are sent over the DDL to the RORC (see section ***Requirements on the sub-detectors***).

**The *L2a message* and the *L2r word***

**1.21.12** Following the **L2** decision, either the ***L2a*** *message* or the ***L2r*** *word* shall be sent to all participating sub-detectors.

The ***L2a*** *message* shall be a copy of the message sent to the DAQ (see section ***The generation of the L2a message***).

The ***L2r*** *word* contains no data, but it shall be transmitted in the same form and over the same hardware link as the ***L2a*** *message* in order to prevent "signal overtaking".

Note 1: Both the ***L2a*** *message* and the ***L2r*** *word* shall be transmitted to the sub-detector front-end electronics *via* the *channel B* of the TTC system. The channel is also used for transmission of a number of other signals, most of which are given higher priority, in order to preserve their synchronous nature. As a result, the **L2** signals are likely to encounter further delays and a transmission jitter of several microseconds. The order of signals, on the other hand, shall *always* be retained.

**1.21.13** The ***L2a*** *message* shall be attached to the raw event data before they are sent over the DDL to the RORC (see section ***Requirements on the sub-detectors***).

**Transmission of the trigger outputs**

The RD12 TTC system has originally been developed for single trigger-level systems (ATLAS, CMS; L1 level only). In case of ALICE (three trigger-levels: L0, L1 and L2), the TTC still provides a convenient and efficient way of "connecting" the CTP to the front-end electronics, but additional hardware is required in order to enable sub-detectors to operate in a *stand-alone* mode. The missing electronics is an interface that provides *orderly* and *uniform,* computer controlled "switching" between the global, *physics run* mode and a local, *stand-alone* mode. The "interface", the ***Local Trigger Unit*** (LTU), could also incorporate a number of common functions, which, otherwise, would have to be part of sub-detector electronics; the LTU would simplify and tidy up the electrical connection between the CTP and the sub-detectors.

The LTU could be realised as another VME board to accompany the TTCvi board [11] – the current VME interface of the TTC system. Its main function would be a computer-controlled selection among three sources of trigger signals:

- the CTP (in the *physics run* mode);
- the *Local System Processor (*in a *stand-alone,* test mode);
- a local trigger generator (also in a *stand-alone* mode).

The following common functions could also be implemented on the board:

- generation of the **L0/L1/L2** sequence in stand-alone mode;
- programmable delay of the local **L1** signal in respect to the **L0**;
- programmable delay of the local **L2a/L2r** in respect to the **L1**;
- generation, in stand-alone mode, of the ***L1*** *trigger-type word*, and access to the 8-bit port/connector of the TTCvi board;
- generation, in stand-alone mode, of the ***L2a*** *message* and the ***L2r*** *word*, and access to the corresponding FIFO of the TTCvi board;
- scalers that measure locally the rate of trigger signals and enable on-line checking of the CTP/sub-detector link;
- a "snap-shot" memory that samples at full speed the trigger signals over a prolonged period of time; the same memory could be used as an alternative trigger source in either a single-shot or a cyclic mode;
- etc.

The way in which the CTP outputs are transmitted is strongly affected by the final configuration of the LTU board. Since the concept and the role of the board are still being considered, the text in this section remains incomplete.

**1.21.14** The ***L2a*** *message,* the ***L2r*** *word* and the ***L1*** *trigger-type word* shall all be transmitted to the sub-detector Local Trigger Crate (LTC) over a shared parallel data bus, synchronized by the **BC** clock (25 ns). The transmission of the ***L1*** *trigger-type word* shall have the highest priority; the transmission of the ***L2a*** *message* and the ***L2r*** *word* shall be organized on "first come, first served" basis.

Note 1: A proposal for a 10-bit wide (8-bit data) bus is shown in **Figure 3.1.1.** The data format is compatible with the proposed TTC frame format (see the next requirement, Note 2).

**1.21.15** The trigger outputs shall be transmitted to the sub-detector front-end electronics in the following way:

| | |
|---|---|
| **L0** | - dedicated cable; |
| **L1** | - TTC, *channel A;* |
| **L1 trigger-type word** | - TTC, *channel B*, TTCvi [11]; |

| | |
|---|---|
| **Orbit** | - TTC, *channel B,* priority 0 (the highest); |
| **Pre-pulse** | - TTC, *channel B*, priority 1; |
| **L2a message** | - TTC, *channel B,* priority 3; |
| **L2r word** | - TTC, *channel B*, priority 3; |
| **RoI data** | - TTC, *channel B*, priority 3. |

Note 1: The feasibility of the scheme needs to be *urgently* considered.

Note 2: A proposal for the TTC frame format and the allocation of the TTC frame addresses is shown in **Figure 3.1.2**, **3.1.3** and **3.1.4**.

**1.21.16** The sub-detectors shall provide the cables and shall be responsible for the connection to the outputs of the CTP system.

Note 1: The signal levels and the details of the trigger outputs (cable type, connectors, pin assignments, polarity, etc.) must be specified before the Preliminary Design Review.

Note 2: The introduction of the Local Trigger Unit (LTU) could make the requirement unnecessary.

Note 3: Having 50 trigger classes will mean that the L2a message may have to be longer than that shown in figures 3.1.1 and 3.1.3.

### 3.1.19 *Region of Interest*

**1.19.1** The implementation of a *segmented readout*, controlled by the *Region of Interest* (**RoI**) logic, shall be an *add-on option*, rather than an integral part of the *baseline* CTP.

**1.19.2** For each **L1** trigger, with the asserted *Enable Segmented Readout* flag in the corresponding ***L1*** *trigger-type word (see section* ***The L1 trigger-type word****)*, the TRD shall generate the ***RoI*** *data* and broadcast them to all the sub-detectors participating in the **RoI** option. The broadcast shall take place after the **L1** trigger.

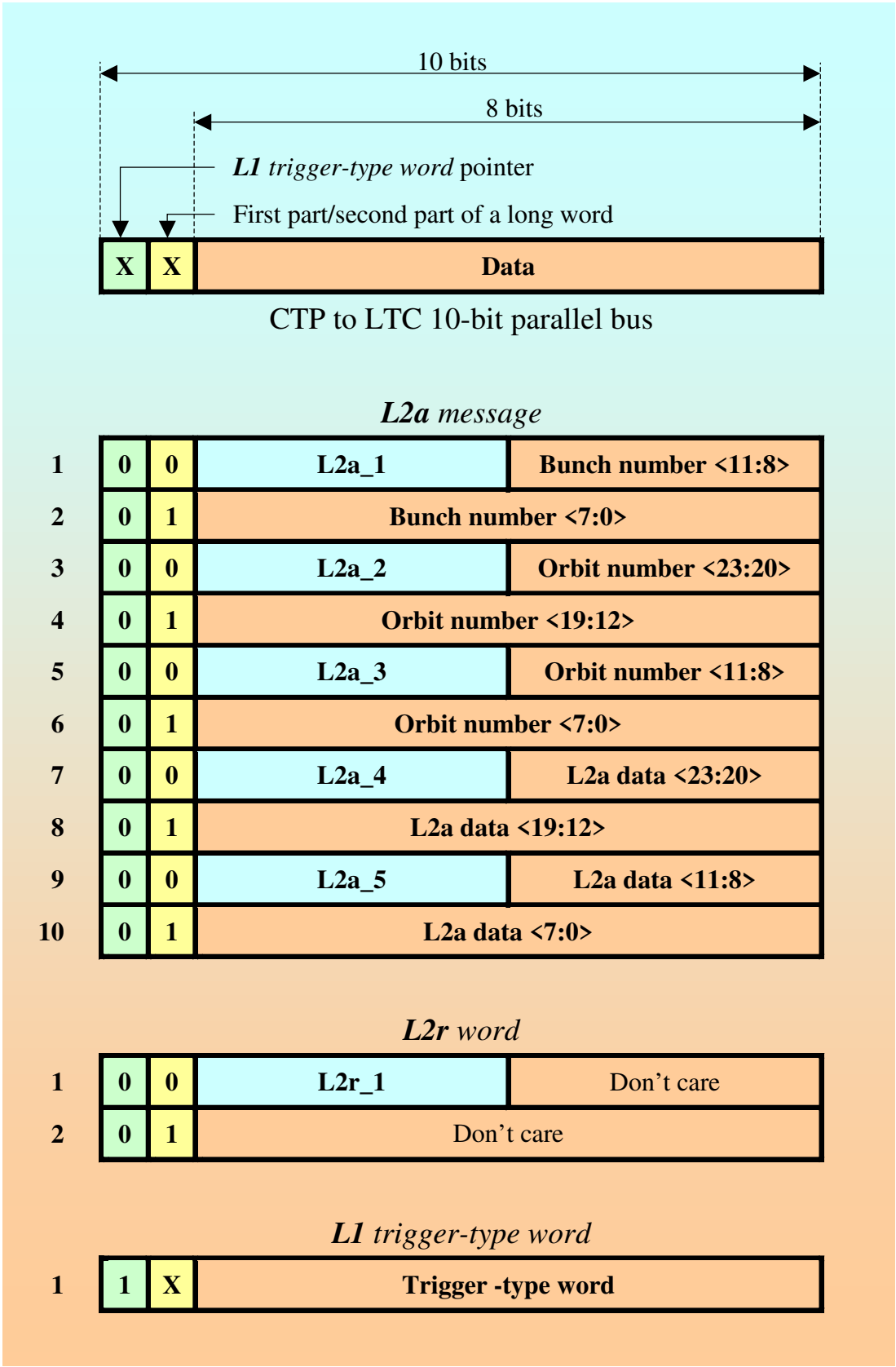


**Figure 3.1.1** CTP to LTC parallel bus - *proposal*

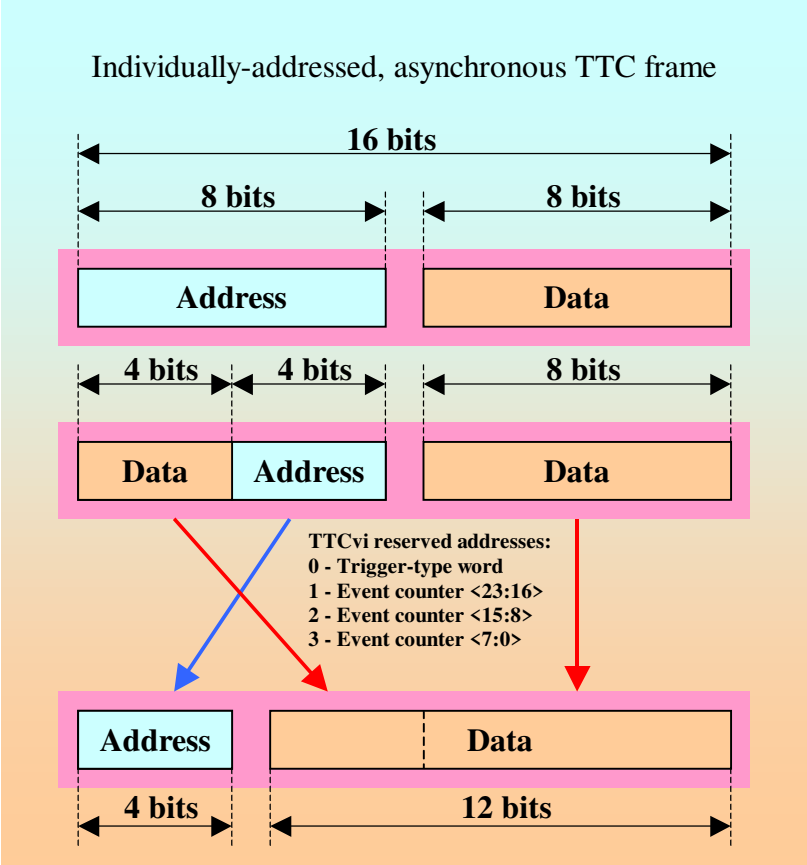


**Figure 3.1.2** TTC frame format - *proposal*

**1.19.3** The ***RoI*** *data* shall contain the event *bunch number* (12 bits) and **36** bits of regional information.

Note 1: The format of the ***RoI*** *data* needs to be defined in close collaboration with the DAQ group and the sub-detectors that participate in the **RoI** option.

**1.19.4** For all the **L2a**-selected events for which the ***RoI*** *data* have been generated (the events with the corresponding *Enable Segmented Readout* flag asserted), the ***RoI*** *data* shall be attached to the raw event data before they are sent over the DDL to the RORC (see section ***Requirements on the sub-detectors***).

**1.19.5** The CTP system and the readout sub-detectors shall be fully compatible with the transmission of the ***RoI*** *data* to the front-end electronics and shall provide the environment for inclusion of the **RoI** hardware.

Note 1: In the current proposal, the ***RoI*** *data* shall be transmitted asynchronously *via* the TTC *channel B*; the priority-level 3 shall be shared with the ***L2a*** *message* and the ***L2r*** *word*; the access to the corresponding FIFO on the TTCvi board [11] shall be from the VMEbus (with a lower bus-grant priority than the ***L2a*** *message/**L2r***

*word* channel). A proposal for the data format and the frame address allocation is shown in **Figure 3.1.3.**

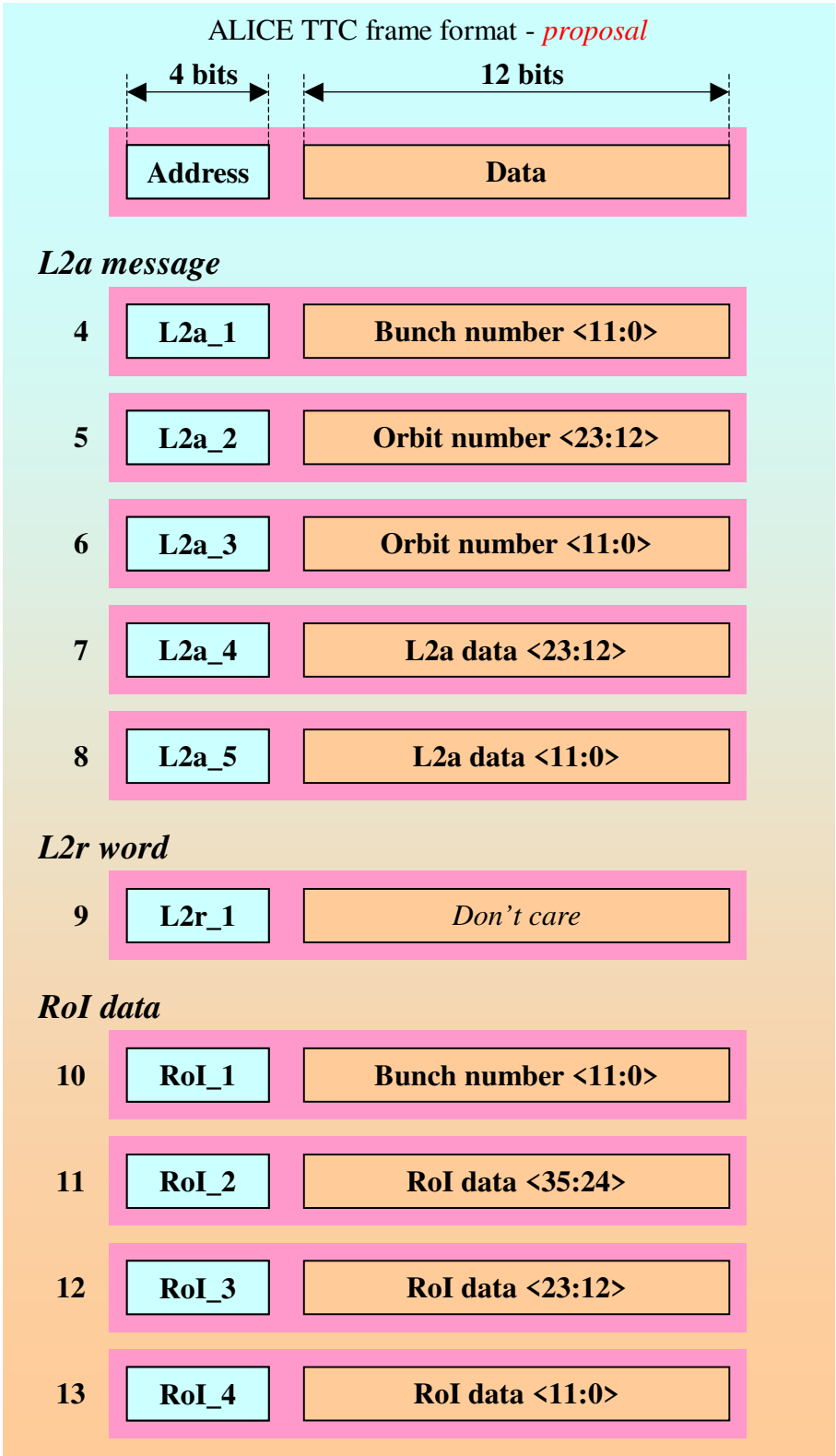

**Figure 3.1.3** TTC frame address allocation - *proposal*

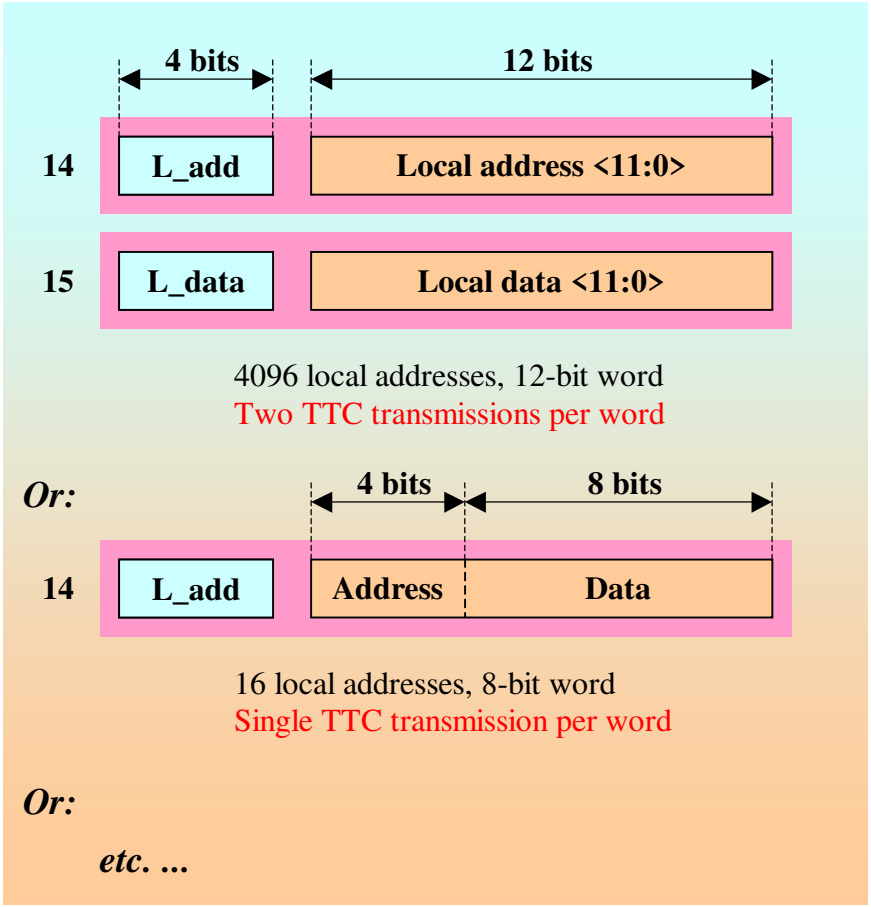


**Figure 3.1.4** Local address allocation - *proposal*

### 3.1.20 *Event identification*

**1.20.1** The ALICE *event-identifier* shall be the datum of the corresponding interaction: the *bunch number* (**12 bits**) and the *orbit number* (**24 bits**).

Note 1: The CTP shall follow the LHC convention: bunches are numbered from 0 to 3563; bunch 0 is the last bunch of the long LHC gap.

**1.20.2** The *event-identifier* shall be generated *centrally* at the CTP and, following the **L2a** decision, shall be distributed to the DAQ and all the participating sub-detectors as part of the ***L2a** message* (see section ***The generation of the L2a message***). The message shall be attached to the raw event data at the sub-detector front-end, before the data are sent to the RORC (DAQ).

**1.20.3** Following the arrival of the **L1** signal, the sub-detectors shall read the corresponding *bunch number*, generated by the TTCrx counter [10]. The sub-detectors shall attach the number (the full length - 12 bits, or a reduced length - only the least significant part) to the raw event data; the number shall serve as a *mini event-identifier*.

The number of bits of the *mini event-identifier* shall be decided by individual sub-detectors to suit, among other things, the architecture of their readout electronics.

Note 1: The DAQ group would prefer the sub-detectors to agree on a unique size for all the *mini event-identifiers.*

**1.20.4** At the sub-detector front-end, prior to their transmission to the RORC, all the **L2a**-selected events shall have the following event identification *elements* already attached to the raw data:

- the *event-identifier*, broadcast from the CTP in the ***L2a*** *message;*
- the *mini event-identifier,* read from the counter in the TTCrx chip at the arrival of the **L1** trigger;
- the *bunch-number*, generated by the **RoI** logic and included in the ***RoI*** *data* (the case of sub-detectors that participate in the option).

Note 1: The availability of the event-identification elements enables an early (at the front-end) check of the event data integrity; a detected error might lead to a recovery procedure; etc. . Any required action would be a part of the overall DAQ system strategy and the detailed specification is the sole responsibility of the DAQ group. The decisions have to be made *urgently* since they affect the design of the sub-detector front-end and control electronics, the task that is already in progress.

### 3.1.21 *General*

**1.21.1** The parameters used to define the configuration and the operation of the CTP shall be logged and archived for each run and each change of status; the data shall be available for use by the off-line analysis.

**1.21.2** The construction of the CTP shall conform to CERN safety rules (fire safety of cables, etc.).

**1.21.3** The rack-space requirements of the CTP shall be agreed with the ALICE Technical Coordination. A broad discussion shall take place to optimise the overall latency of the **L0** and **L1** triggers (best location of each electronics crate, best routing of cables).

**1.21.4** The cooling requirements of the CTP shall be agreed with the ALICE Technical Coordination.

## 3.2 Control requirements

**2.0.1** All control functions shall be performed within the context of the standard ALICE run-control and detector-control systems.

Note 1: The roles of the run-control and detector-control systems need to be clarified, including issues of the physical path that will be used to send control commands and parameters.

Note 2: Run-control shall be used for configuring the system.

Note 3: DCS shall be used for controlling power supplies and for monitoring operational parameters (temperatures, voltages, etc.).

Note 4: The read-out path (run-control or DCS) for scaler information needs to be defined.

**2.0.2** It must be possible to read back (non-destructively) the values of the programmable parameters in the system.

**2.0.3** The control path shall be independent of the readout path.

**2.0.4** All parameters related to setting-up the timing of the system shall be programmable. These include programmable delays and related circuits, and phase adjustments for clocks in the processing pipeline.

**2.0.5** All parameters for the CTP algorithm shall be programmable.

**2.0.6** It shall be possible to carry out all control operations without having to touch the hardware. This requirement also applies to setting up the timing for the CTP.

**2.0.7** In normal operation, the CTP system shall be able to perform control operations within the following time limits:

- set up system from power on in under 15 minutes;
- full reload of all parameters in under 1 minute.

**2.0.8** Control parameters for the CTP shall be stored in the standard (Trigger/DAQ) database system.

**2.0.9** There shall be software tools to set up the CTP parameters without requiring any detailed knowledge of the actual hardware.

## 3.3 Monitoring requirements

**3.0.1** Local monitoring shall be independent of the main DAQ system.

Note 1: It must be defined where the on-line system monitoring tasks should run.

**3.0.2** In normal operation, the correct functioning of the CTP shall be checked using the data read from the diagnostic, “snap-shot” memories; in particular, checks shall be made that the output data is consistent with the corresponding input data.

**3.0.3** Scalers shall be provided (see section ***Scalers***) to monitor the quality of the trigger inputs to the CTP and the overall performance of the trigger system.

**3.0.4** Can some monitoring of the input links be achieved by comparing the readout data from the trigger sub-detectors and from the CTP?

**3.0.5** The CTP system must maintain detailed statistics on errors, allowing efficient fault diagnosis.

**3.0.6** The system shall have sufficient error checking facilities to detect and diagnose most failures in 2 minutes.

Note 1: The maximum time should be re-discussed.

## 3.4 Test requirements

**4.0.1** It shall be possible to operate together in test mode the trigger sub-detectors, the CTP, the TTCs and the CTP RORC.

**4.0.2** It shall be possible to test and to diagnose faults in the CTP *in situ*, without having to touch the hardware.

**4.0.3** It shall be possible to test all programmable memories (control registers, look-up tables, diagnostic memories) for data and addressing errors; for all such memories it shall be possible to read back non-destructively their contents.

**4.0.4** It shall be possible to test the logic in the system by comparing the output data with the expected values while test data are being fed into the system at full speed.

**4.0.5** It shall be possible to check back-plane connections in the system by reading the data at both ends of such links while test data are being fed into the system at full speed.

**4.0.6** It shall be possible to generate test data at the input to the CTP to test and debug the CTP in isolation from the trigger sub-detectors.

**4.0.7** It shall be possible to capture data in diagnostic memories at the input to the CTP.

**4.0.8** It shall be possible to test the correct functioning of the input links to the CTP by using test data generated by the triggering sub-detectors.

Note 1: The generation of test data needs to be discussed with the trigger sub-detector groups.

**4.0.9** It shall be possible to capture data in diagnostic memories at various points in the CTP processing chain in order to test and debug the preceding processing logic. It shall be possible to capture the data sent on the output link to the TTCs.

**4.0.10** It shall be possible to test the correct functioning of the output link from the CTP to the RORC using the data that are read out from the RORC while the test data are being fed into the system at full speed.

**4.0.11** It shall be possible to complete test procedures within the following time limits:

- standard test sequence in under 10 minutes;
- comprehensive "soak" test in under 8 hours;
- switch from test mode back to normal mode in under 15 minutes.

## 3.5 Maintenance requirements

**5.0.1** It shall be possible to replace a faulty module, power supply, cable (electrical or optical) or a crate (back-plane problem, etc.) in the system by a spare one, and to make a standard test of the system after the repair has been made, in under two hours.

Note 1: To achieve this it might be necessary to have in place spare cables of identical length.

**5.0.2** Supplies of fully tested spare modules, power supplies, cables, fans, back-planes, crates etc. for all parts of the CTP shall be provided at CERN.

**5.0.3** Test station shall be provided at CERN to allow full and independent diagnosis, repair and testing of all CTP components.

**5.0.4** During running, experts capable of maintaining the system shall be available at CERN.

## 3.6 Requirements on the external system connections to the CTP

### 3.6.1 *Requirements on the sub-detectors*

**6.1.1** In order to enable automatic synchronization of trigger inputs with the **BC** clock in the CTP (see requirement **1.3.1**), *all trigger sub-detectors* shall be required to transmit a pattern of triggers in alternate bunch-crossing intervals (sub-detector BC clock divided by two).

Note 1: A circuit that generates both the NRZ trigger input and the required synchronisation pattern is shown in **Figure 3.6.1**.

**6.1.2** Before an **L2a**-selected event is sent over the DDL to the RORC (DAQ), *the readout sub-detectors* shall attach the following trigger information to the raw data:

- *event-identification elements* (see section ***Event identification***),
- ***L1*** *trigger-type word* (see section ***The L1 trigger-type word***),

- ***L2a** message* (see section ***The generation of the L2a message***).

Note 1: The format of the additional data needs to be defined in close collaboration with the DAQ group.

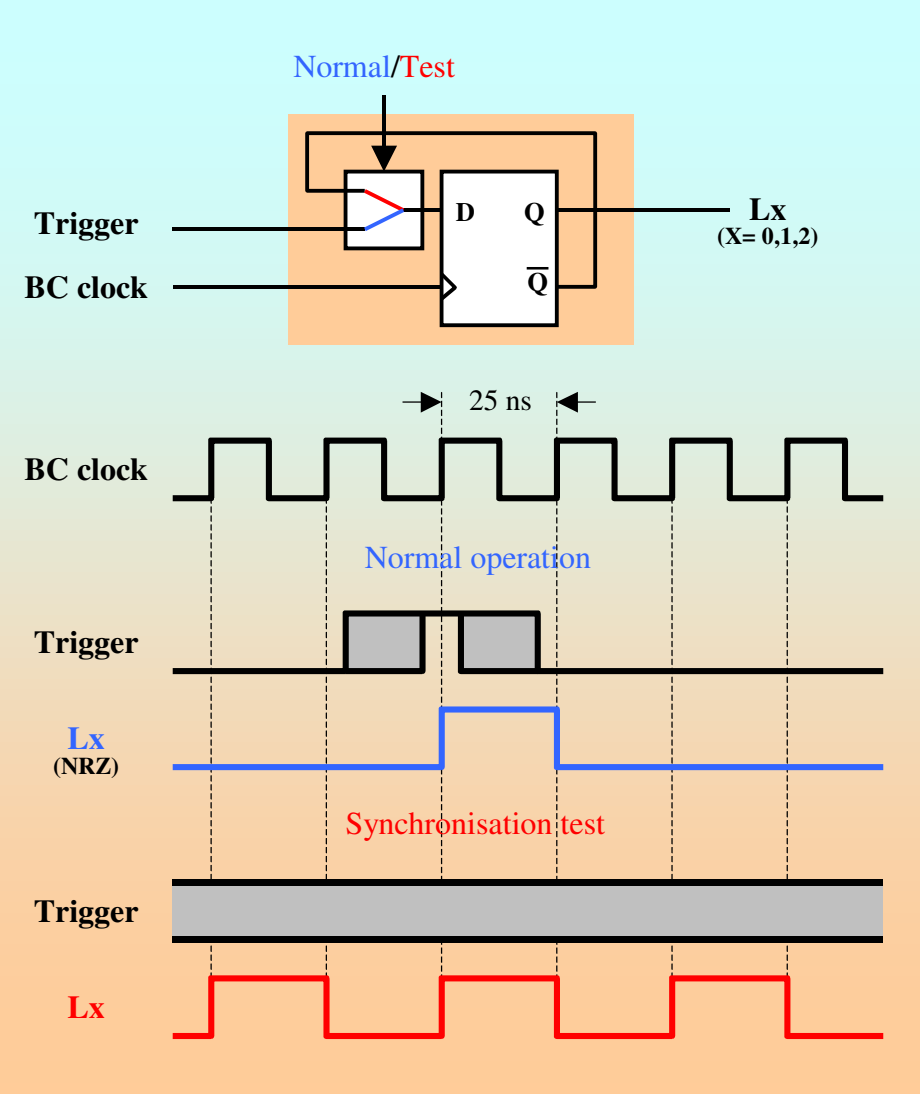


**Figure 3.6.1** Synchronisation of trigger inputs

**6.1.3** The *readout sub-detectors that participate in the **RoI** option* shall also attach the ***RoI** data* (see section ***Region of Interest***).

Note 1: The format of the attached ***RoI** data* needs to be defined in close collaboration with the DAQ group.

Note 2: The ***RoI** data* are available only for events with the asserted *Enable Segmented Readout* flag in the ***L1** trigger-type word* (see section ***The L1 trigger-type word***).

**6.1.4** During the physics run, the sub-detectors shall reduce their calibration trigger requests to a necessary minimum.

Note 1: Although the calibration triggers are orderly interleaved with the physics triggers, the calibration, nevertheless, interferes with the normal data-taking since it produces significantly larger events and keeps the sub-detectors busy for an extended time; also, in cases when calibration charge is injected, the concurrent physics events could be affected.

The sub-detectors should aim to perform most of their calibration tasks in a stand-alone mode, before they join the run. In order to further reduce the calibration requirements during the run, the sub-detectors should consider the following:

- In some cases, pedestal monitoring can be done "on the fly", by "spying" on the raw physics data before the zero suppression has been applied.
- Gain fluctuations are usually long-term effects and it could be sufficient to perform the gain calibration (which requires the charge injection) only before and/or after the run.
- The on-line status of the hardware could be "extracted" from the physics data - "activity histograms" clearly reveal dead or noisy channels.
- Etc.

Minor modifications or additions to the sub-detector electronics could help to significant reduce the requirements for calibration triggers during the run.

**6.1.5** If the calibration charge is injected following a **Pre-pulse**, but no **L0** is generated (the **BUSY** logic, or past-future protection veto), some sub-detectors require time to recover and get ready for the next physics trigger. In order to preserve event integrity, the sub-detectors shall, in response to the **Pre-pulse,** assert their **BUSY** signal in such a way that it permits the generation of the calibration **L0**, but prevents further physics **L0**s until the effects of charge injection have ceased.

Note 1: The timing of the **Pre-pulse** and the calibration **L0** shall allow enough propagation time and wide margins for setting of the sub-detector **BUSY** signal and its transmission to the CTP.

### 3.6.2 *Requirements on the DAQ system*

**6.2.1** The **DAQ** system shall generate the **DAQ BUSY** signal that halts the generation of **L0** triggers *for all trigger classes*.

Note 1: The nature of the **DAQ BUSY** (hardware signal, or a software message) and the implementation will be discussed with the DAQ group.

Note 2: An alternative signal name, *Enable trigger*, has been favoured by the DAQ group.

### 3.6.3 *Requirements on the CTP RORC*

**6.3.1** It shall be possible to operate the CTP with the RORC when running in test mode.

**6.3.2** In case the DAQ is not available, or in a test mode, it shall be possible to read-out the CTP through the VME, at a lower rate.

### 3.6.4 *Requirements on the run-control and DCS*

Major system processes, like *Run Control*, *On-line Monitoring*, *Error-message Utility*, *System Control*, *Detector Control*, *etc.* , haven't yet been partitioned among ALICE subsystems. As a result, the text in this section remains very preliminary.

**6.4.1** The run-control system shall provide for loading and checking of all parameters in the CTP as part of the initialisation procedure.

**6.4.2** The run-control system shall provide for setting up different modes of operation.

**6.4.3** The run-control shall include test and diagnostic programs that perform all of the checks described in the ***Test requirements*** section.

**6.4.4** The run-control system shall include monitoring programs that analyse the CTP event data for **L2** selected events.

**6.4.5** The run-control system or DCS shall include monitoring programs that read and analyse the scaler information.

**6.4.6** The run-control system or the DCS shall handle error messages received from the on-line monitoring programs that analyse data samples read-out from the diagnostic, "snap-shot" memories.

**6.4.7** The DCS shall include programs that monitor and control the power supplies and the cooling system.