The *CTP Readout* and the *Interaction Record* data format - *proposal*

Last updated: 28 June 2002

Introduction

The *CTP Readout* and the *Interaction Record* data shall be generated by the CTP and transmitted to the DAQ *via* the ALICE optical Detector Data Link (DDL) [1]. The hardware and the communication procedure shall be standard - identical to the channels that transmit the sub-detector readout. The nature of the data, and the timing and rate of their generation, on the other hand, differ significantly from the sub-detector readout and shall be best served by a "customised" data format. That data format is the subject of the proposal.

Neither the *CTP Readout* nor the *Interaction Record* data contribute to the event-building task. The CPT Readout is *only a redundant channel* that carries *exactly the same information* broadcast, at the time of **L2a** decision, to all the participating sub-detectors (***L2a** Message*). It could be used by the ALICE data-driven DAQ system to resolve error conditions in some rather uncommon operating modes (case of a cluster that contains only one or two sub-detectors). Otherwise, the DAQ performance would not be impeded by the absence of the CTP data.

The *Interaction Record* is an *aid* to the pattern recognition task. The generation of the record is continuous, rather than "triggered" by any CTP or DAQ action. The data do not "interfere" with any on-line operation - they only need to be archived for the off-line use. The CTP URD [2] "permits" occasional gaps in the recording.

The proposed data format significantly simplifies the hardware blocks in the CTP that deal with the record generation: the data word is effectively reduced to 12(13) bits; the sequence closely follows the data flow in the CTP; no "editing" is required; *etc*. . The data transmitted to the DAQ contain a very different information compared to the raw event data that make up the sub-detector readout and they will have to be "processed" by a *dedicated* DAQ software. If a "unified" format, equal or similar to the sub-detector readout format, offers advantages further up the processing chain, the data can be "re-edited", headers appended, *etc*. by the dedicated software at no cost and in a more flexible way. The main motivation for the proposal is the welcome simplification of the CTP hardware.

With the progress of the preliminary design of the CTP, some implementation details are becoming known. Even at this very early stage, they help to get a more clear picture of the final system and to better understand the implications of some design decisions. For that reason, although the implementation is not the subject of this proposal, the text also contains brief descriptions of some aspects of the CTP hardware.

References

[1] ALICE DDL - *Interface Control Document*, ALICE/96-43, Internal Note/DAQ, 12 December 1996, Revision 9.1.

ALICE DDL - *Hardware Guide for the Front-end Designers*, ALICE/98-21, Internal Note/DAQ, 19 May 1998, Revision 1.3

[2] ALICE CTP - *User Requirement Document*, Draft 1.0, 24 October 2001.

The *CTP Readout* data format

For *each* **L2a** decision, the CTP transmits to the DAQ a data block, the *CTP Readout*, containing the same information that is also broadcast to all the participating sub-detectors. The transmission medium, the DDL dedicated to the CTP, is also used for the *Interaction Record*; the *CTP Readout* block is uniquely "marked" by the *Block Identifier* bit cleared to 0. The proposed data format is shown in **Table 1**.

The following abbreviations have been used:

BlockID	*Block Identifier* bit: cleared (0) for the *CTP Readout*; asserted (1) in case of the *Interaction Record*.
BCID[11..0]	*Bunch-crossing number*, part of the *Event Identifier*.
OrbitID[23..0]	*Orbit number*, part of the *Event Identifier*.
ESR	*Enable Segmented Readout* flag (RoI option).
L2SwC	*Software Class L2 trigger* status: cleared for the *physics trigger*; asserted for *the software trigger.*
L2Cluster[6..1]	*Cluster [6..1] L2 trigger* status flag.
L2Class[50..1]	*Class [50..1] L2 trigger* status flag.
ClT	*Calibration Trigger* flag.
L2Detector[24..1]	*Detector [24..1] L2 trigger* status flag.
EOBTR	*End of Block Transmission,* front-end command that closes any block transmission transaction (DDL).

The *CTP Readout* data block has a constant length of 8 DDL words. The block is terminated by the **EOBTR** command (*Word 9*), in compliance with the DDL interface protocol [1].

The *"don't care"* bits carry no information and should be ignored; if required, they could be made to read 0.

The first three words always contain the *Event Identifier* (**BC** and **Orbit** number of the corresponding event). The remaining words (*Word 4-8*) carry different information in case of the *physics trigger* (**L2SwC** = 0) and in case of the *software trigger* (**L2SwC** = 1).

Implementation

The *CTP Readout* data blocks shall be generated "on the fly" from the serialized *L2 Data.* In case of the *software trigger*, some data must be extracted from the requestor's *trigger description*; for that reason, a "copy" of the description, normally required only on the *Fan-out* (FO) boards, must also be stored on the *L2 board*. The block generation shall be completed during the minimum time between consecutive **L2a** triggers (equal to the CTP ***L0*** *dead time* and currently set to 2μs) - no data queuing shall be required at this stage.

The formatted *CTP Readout* data blocks shall be stored into a FIFO buffer of a "modest" capacity, sufficient to "smooth" possible irregularities of the DDL operation and enable the sharing of the DDL link with the transmission of the *Interaction Record*.

In case of concurrent requests for the transmission over the common DDL, the *CPT Readout* data blocks shall be given priority over the *Interaction Record* blocks.

The transmission of the *CTP Readout* block (9 words) over the DDL takes 225ns; with the compulsory gap of 16 clock periods between consecutive transmissions [1], the complete operation lasts 400ns.

Word	[31..14]	[13]	[12]	[11..0]
1	*Don't care* (0)	**BlockID** = 0	*Don't care* (0)	**BCID**[11..0]
2				**OrbitID**[23..12]
3				**OrbitID**[11..0]

Physics trigger

	Bit	*Data*
Word 4	[31..14]	*Don't care* (0)
	[13]	**BlockID** = 0
	[12..11]	*Don't care* (0)
	[10]	**ESR**
	[9]	*Don't care* (0)
	[8]	**L2SwC** = 0
	[7..2]	**L2Cluster** [6..1]
	[1..0]	**L2Class** [50..49]

Software trigger

	Bit	*Data*
Word 4	[31..14]	*Don't care* (0)
	[13]	**BlockID** = 0
	[12..11]	*Don't care* (0)
	[10]	*Don't care* (0)
	[9]	**CIT**
	[8]	**L2SwC** = 1
	[7..2]	*Don't care* (0)
	[1..0]	*Don't care* (0)

Word	[31..14]	[13]	[12]	[11..0]
5	*Don't care* (0)	**BlockID** = 0	*Don't care* (0)	**L2Class** [48..37]
6				**L2Class** [36..25]
7				**L2Class** [24..13]
8				**L2Class** [12..1]

Word	[31..14]	[13]	[12]	[11..0]
5	*Don't care* (0)	**BlockID** = 0	*Don't care* (0)	**L2Detector** [24..13]
6				**L2Detector** [12..1]
7				*Don't care* (0)
8				*Don't care* (0)

Word	[31..14]	[13]	[12]	[11..0]
9	*Don't care* (0)	**BlockID** = 0	*Don't care* (0)	**EOBTR** (h0B4)

Table 1 *CTP Readout* data format

"Proper" buffering of the *CTP Record* (and the *Interaction Record*) data shall be provided by the CTP's Local Data Concentrator (LDC). The capacity of the buffer is massive, and especially so if the small size of the CTP data blocks is taken into account.

If, for whatever reason, the "back-pressure" of the DDL halts the data transmission and the *Nearly Full* mark of the CTP FIFO is reached, the **CTP BUSY** signal shall be asserted which, in turn, disables further generation of the **L0** triggers. The action is fully in accordance with the procedure described in the CTP URD [2]. The status of the **CTP BUSY** signal is monitored and timed.

The remaining free capacity of the FIFO must be sufficient to "absorb" all the **L0/L1** triggers already in the pipeline that are waiting for the corresponding **L2** decisions. In real conditions, their *maximum* number is likely to be rather small - 5 to 10 events. But the hardware must be capable of error-free operation in *all possible* situations, regardless of how unlikely they might be. Assuming at least 4 non-overlapping sub-detector clusters, with at least 15-event effective capacity of their front-end multi-event buffers, with no past-future protection, the CTP **L0** dead time reduced to 1.5μs, the **L1** decision time of 6μs and the **L2** decision time of 90μs, the *theoretical* maximum number of the **L0/L1** triggers already in the pipeline would be 60 events. It could also be assumed that the "moderate" FIFO capacity of 25 events is sufficient to "smooth" the irregularities of the DDL operation and enable the link sharing (see above). The resulting requirement for the FIFO capacity is 85 data blocks (765 words), with the *Nearly Full* status set to become active when the occupancy exceeds 25 events.

In case of the CTP DDL fault, the interlock based on the **CTP BUSY** signal would certainly secure a *fail-safe* behaviour of the system, but the trigger generation would cease. With the restricted access to the hardware, the CTP is likely to remain inoperative for a prolonged period of time. Since neither the *CTP Readout* nor the *Interaction Record* are essential (see ***Introduction***), it is prudent to make the **CTP BUSY** interlock a programmable option.

The *Interaction Record* data format

The CTP shall transmit the *Interaction Record* data block *for each LHC orbit*, even if no interaction has been detected during the orbit. The transmission medium, the DDL dedicated to the CTP, is also used for the *CTP Readout*; the *Interaction Record* block is uniquely "marked" by the *Block Identifier* bit asserted to 1. The proposed data format is shown in **Table 2**.

The following abbreviations have been used:

BlockID	*Block Identifier* bit: cleared (0) for the *CTP Readout*; asserted (1) in case of the *Interaction Record*.
Err	*Transmission Error* flag.
InT	*Interaction Type* flag: cleared (0) for *peripheral* and asserted (1) for *semi-central* interaction.
EOBTR	*End of Block Transmission,* front-end command that closes any block transmission transaction (DDL).

The first two words of the data block contain the number of the corresponding LHC orbit (24 bits). They are followed by a string of words containing numbers (12 bits) of bunch-crossings in which the interactions have been detected and the corresponding *Interaction Type* (**InT**) descriptors.

The maximum number of "bunch-crossing" words in the data block is set to 250. The limit allows nearly an order of magnitude margin even for the highest quoted interaction rate of 300kHz (proton-proton mode; nominal luminosity). Should the limit be nevertheless exceeded (example shown in **Table 2**), further interaction "recording" in the corresponding orbit would be abandoned and the *Incomplete Record* message (*peripheral* interaction at a virtual bunch-crossing 4095 - hexadecimal FFFF) would be appended as a word 253. The "loss" of recording hardly matters since, in the circumstances, the pattern recognition would be no longer an option.

In case when no interaction is detected during the orbit, the data block contains only the orbit number (words 1 and 2).

The block is always terminated by the **EOBTR** command, in compliance with the DDL interface protocol [1].

The *"don't care"* bits carry no information and should be ignored; if required, they could be made to read 0.

During a run, under normal circumstances, the DAQ should receive, in sequential order, the *Interaction Record* data blocks for *all* the LHC orbits. If, for whatever reason, the transmission of a block, or of a number of consecutive blocks, fails, the first block *after* the "gap" shall have the *Transmission Error* flag (**Err**) asserted. The *Interaction Record* data are not essential (see ***Introduction***) and, in the CTP URD [2], provisions have been made for instances of "gap" occurrence. The conditions that could lead to transmission errors and their likelihood are discussed in ***Implementation***.

Word	[31..14]	[13]	[12]	[11..0]
1	*Don't care* (read 0)	**BlockID** = 1 (*Interaction Record*)	**Err**	**Orbit** number [23..12]
2			**Err**	**Orbit** number [11..0]
3			**InT**	**BC** number [11..0]
4			**InT**	**BC** number [11..0]
⋮			⋮	⋮
n			**InT**	**BC** number [11..0]
⋮			⋮	⋮
251			**InT**	**BC** number [11..0]
252			**InT**	**BC** number [11..0]
253			0	*Incomplete record* (hFFF)
254			0	**EOBTR** (h0B4)

Table 2 *Interaction Record* data block

Implementation

The *Interaction Record* data shall be generated by two identical hardware circuits that "record" the interactions in an orbit and then, during the following orbit, complete the formatting (attach the orbit number, the *Incomplete record* word - if appropriate, and the **EOBTR** command) and transfer the data block to the DAQ. The two circuits operate in turn: while one is "recording" the other one is transmitting, and then the roles are swapped.

The data shall be transmitted over the CTP DDL. As already mentioned, the link shall also be used for transmission of the *CTP Readout* data (see previous section) and, in case of concurrent transmission requests, the *CPT Readout* data blocks shall be given the priority over the *Interaction Record* blocks.

The transmission over the DDL of the longest possible *Interaction Record* block (254 words, see **Table 2**), including the compulsory gap of 16 clock periods between consecutive transmissions [1], takes 6.75μs. Assuming again the hypothetical "deluge" of 60 **L2a** triggers during the orbit duration (see ***Implementation*** in the previous section) and the corresponding 60 transmissions of the *CTP Readout* data blocks (400ns per transmission) over the common link, the overall maximum utilization of the DDL throughput amounts to less than 35%. In those circumstances, no buffering is required on the CTP side. As in the case of the *CTP Readout*, "proper" buffering of the *Interaction Record* data shall be provided by the CTP's Local Data Concentrator (LDC). The capacity of the buffer is massive, and especially so if the small size of the data blocks is taken into account.

The two hardware circuits that generate the *Interaction Record* data blocks would normally "swap roles" (see above) precisely at the end of the current orbit. If, at that point, for whatever reason, the *transmission has not yet started*, the data block shall be "abandoned" and the error flag (**Err)** shall be asserted; the flag shall be included in the *next* transmission, indicating that the previous data block, or a number of consecutive data blocks, have been lost. If, at the end of the orbit, the *transmission is still in progress*, it will be allowed to continue, but the circuits shall not "swap their roles": the circuit that has just completed the "recording" shall immediately start the "recording" of the new orbit by overwriting the data from the previous orbit; the other circuit shall continue the transmission until its completion. The **Err** flag shall be asserted and, as before, included in the *next* transmission.