The I/O Dataformat for the Tilecal Read Out System

A proposal document

J. Castelo

Jose.Castelo@ific.uv.es

IFIC, University of Valencia. SPAIN

Abstract

This note has been born as an initial proposal and after several discussions with detector community, in order to get a clear format for the input and output data for the Read Out Drivers of the Tilecal hadronic calorimeter in ATLAS. Various considerations have been done for decreasing at maximum the input and output bandwidth of the optical links, always having in mind to loose the minimum quantity of information over the raw data received and giving flexibility in the headers for future changes in hardware structure.

INDEX

Figure Index

Table Index

1 Input data format

1.1 The Tilecal cell/channels mapping

On the Figure 1 is represented the Tilecal cells for extended an central barrel with $\eta>0$. The negative part is symetric.

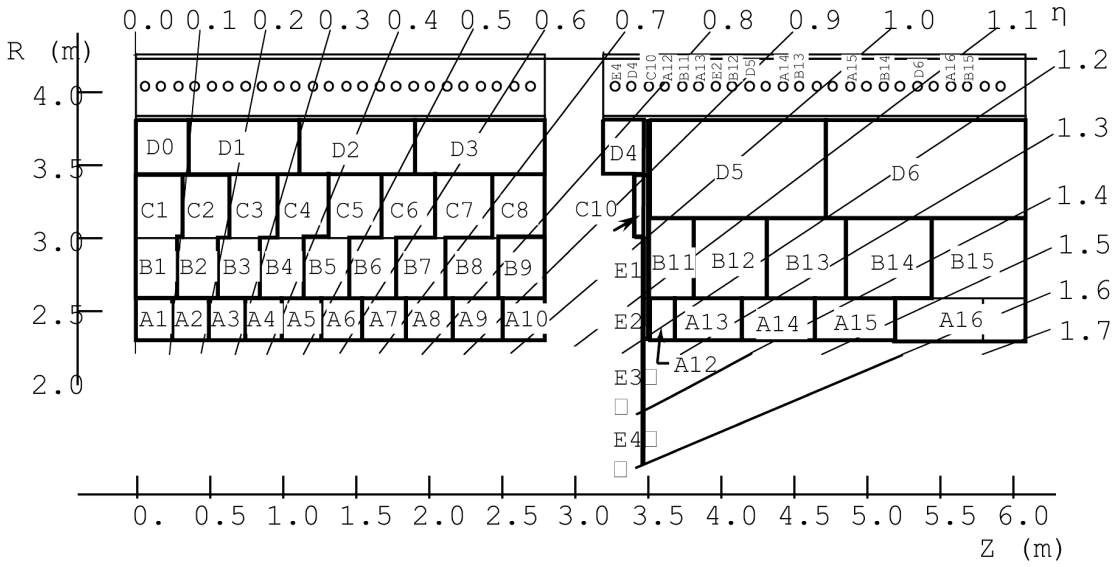


Figure 1 Tilecal towers for CB and EB

Signals from the scintillators are grouped into the rectangular cells shown and each cell is sensed by two photomultipliers (called L and R in the tables below). In the barrel section the cells Bi and Ci are viewed by the same pair PMT, and the cell dimensions are chosen to obtain pseudo-projective towers of nominal width $\Delta\eta$ = 0.1. The D cells have a width $\Delta\eta$= 0.2 because they usually have low energy. One PMT is put in each of the two appropriate $\Delta\eta$= 0.1 towers. Thus each tower contains three longitudinal segments as shown in figure 1.

Eta	PMTs by cell	PMT Positions
0.0-0.1	A1R A1L BC1R BC1L D0R	5 2 3 4 1
0.1-0.2	A2R A2L BC2R BC2L D1L	9 6 7 8 14
0.2-0.3	A3R A3L BC3R BC3L D1R	11 10 13 12 15
0.3-0.4	A4R A4L BC4R BC4L D2L	19 16 17 18 26
0.4-0.5	A5R A5L BC5R BC5L D2R	21 20 23 22 27
0.5-0.6	A6R A6L BC6R BC6L D3L	25 24 29 30 40
0.6-0.7	A7R A7L BC7R BC7L D3R	31 28 35 36 43
0.7-0.8	A8R A8L BC8R BC8L	37 34 41 42
0.8-0.9	A9R A9L B9R B9L	39 38 45 46
0.9-1.0	A10R A10L	47 48

Table 1 PMT cells map for Barrel trigger towers

Eta	PMTs by cell	PMT Positions
0.8-0.9	D4R D4L	3 4
0.9-1.0	C10R C10L D5R	5 6 17
1.0-1.1	B11R B11L D5L	9 10 18
1.1-1.2	A12R A12L B12R B12L D6R	7 8 15 16 37
1.2-1.3	A13R A13L B13R B13L D6L	11 12 23 24 38
1.3-1.4	A14R A14L B14R B14L	21 22 33 34
1.4-1.6	A15R A15L B15R B15L A16R A16L	29 30 43 44 41 42

Table 2 PMT cells map for Extended Barrel trigger towers

Note that the PMT and adder positions correspond to the holes for PMTs in a superdrawer. The cell D0 is read by one PMT on the CB eta>0 (D0R) and another PMT on the other CB eta<0 (D0L). They number from 1 to 48 starting at the innermost end. Signals from the "plug extension" or "gap" scintillators (E1,E2) and from the "inter-cryostat" scintillators (E3,E4) are not proposed for the LVL1 since their use is specialized. They will be available for LVL2. They use PMTs 13, 14, 1, 2, respectively.

Therefore, with a configuration like this, the system has 45 channels for each central barrel, and 28 channels plus 4 channels (ITC) more (32ch) for each extended barrel readout.

1.2 Tilecal front-end electronics configuration. The Digitizer

The tilecal front-end system is instrumented in drawers inside one of the 64 modules of the detector. There are one superdrawer for the central barrel (equivalent to 2 drawers managing 45 channels each) and two drawers for extended barrel (32 channels each). Data are presented to the digitizer boards by the 3-in-1 system via 100 Ohm differential lines, which deliver two versions of each signal, a high and a low gain version, where the gain ratio is 64. The data is converted to discrete time format in digitizers formed mainly by commercial ADCs (Analog Devices AD9050 10bits@40MHz), TTCrx for receipt of TTC info and custom ASIC chip TileDMU (Data Management Unit). Data store only the high gain data unless it overflows (or underflows) in which case the low gain data is used. TileDMU is responsible for reformatting and reordering the digitized data and to send it to the interface links (LSC G-Link cards compliant with S-Link specifications). TileDMU is composed by digital pipeline memories, fifos, serializers, etc…

Each TileDMU manages 3 ADCs, and each digitizer board has 2 TileDMUs. Therefore there are 8 Digitizer board (up to 48ch, only 45 are needed) for central barrels, and 6 for extended (36ch possible, only 32ch are needed). This configuration can be seen in Figure 2.

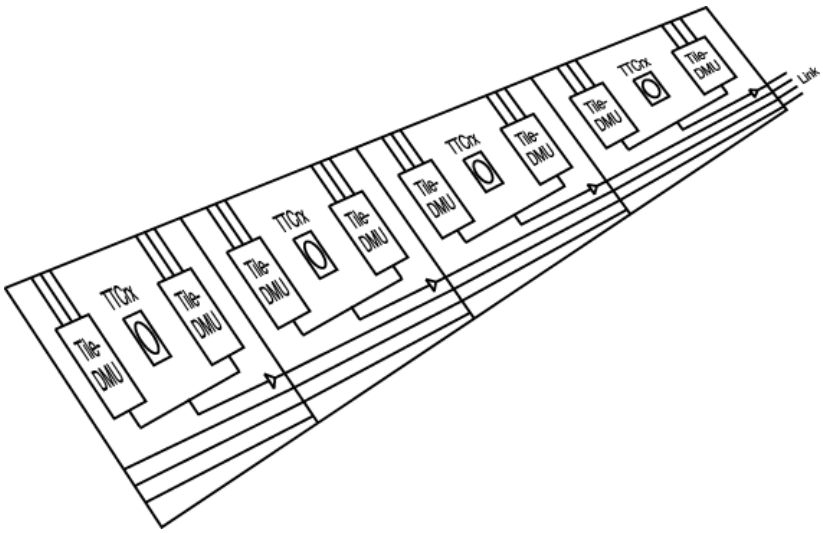


Figure 2 Digitizers map per drawer in CB

The event fragment is organized as indicated in Table 3.

MSB	LSB	
Data word, high/low gain sample 7		Last Word
Data word, high/low gain sample 6		
Data word, high/low gain sample 5		
Data word, high/low gain sample 4		
Data word, high/low gain sample 3		
Data word, high/low gain sample 2		
Data word, high/low gain sample 1		
Header event		First Word

Table 3 Event fragment format

The data word have 32 bits and are aligned as shown in the Table 4. Each fragment contains the data for three PMT channels and a bit of odd parity calculated for all 32 bits. Header format is represented in Table 5.

MSB				LSB
0 P	Data channel 1 (10 bits)	Data channel 2 (10 bits)	Data channel 3 (10 bits)	

Table 4: Three channels data format (32bits)

MSB											LSB
1	p	llll	e	s	d	r	vvvv	0	mm	ggg	bbbb bbbbbbbb

Table 5 Fragment header format

The header starts with bit 31 set to indicate a header. The other bits are as follows:

p	**Parity** (odd)	
l	**Derandomizer length** (number of samples. Up to 16).	
e	**Parity error**. A parity error was detected from the memory in the last readout.	
s	**SEstr** Single Error Strobe received from the ttc.	
d	**DEstr** Double Error Strobe received from the ttc.	
r	**Register parity**. Parity from the registers in the chip.	
v	**Variable parity**. Parity from the variables in the chip.	
0	**Not used**. (Set to 0)	
m	**Mode**.	00 - Normal mode 01 - Calibration mode 10 - Test mode 11 - Not used
g	**High/low gain**. Indicates high(1) or low(0) amplification from the 3-in-1 cards.	
b	**Bunch Crossing**.	

All this data are serialized in two bits by the TileDMU.

The data coming from 16 TileDMU chips are serialized and sent to the link interface card with the format of the Table 6.

Chip 15		Chip 16		Chip 13		Chip 14		Chip 11		Chip 12		Chip 9		Chip 10		Chip 2		Chip 1		Chip 4		Chip 3		Chip 6		Chip 5		Chip 8		Chip 7		
16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	CRC16, bit 16
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Etc
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	CRC16, bit 1
31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	Data, low or high gain sample "n", bit 31 & 32
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Etc
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	Data, low or high gain sample "n", bit 1 & 2
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Etc
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Etc
31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	Data, low or high gain sample 2, bit 31 & 32
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Etc
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	Data, low or high gain sample 2, bit 1 & 2
31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	Data, low or high gain sample 1, bit 31 & 32
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Etc
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	Data, low or high gain sample 1, bit 1 & 2
31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	31	32	Header, bits 31&32
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Etc
3	4	3	4	3	4	3	4	3	4	3	4	3	4	3	4	3	4	3	4	3	4	3	4	3	4	3	4	3	4	3	4	Header, bits 3 & 4
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	Header, bits 1 & 2

Table 6 TILEDMU intermediate format with interface links

For each column a CRC16 is calculated for data consistency checking.

1.3 Interface links in front end motherboards

The interface links are responsible for receiving (one fiber) the TTC information and to send it to Digitizers equipped with the TTCrx chip (8 LVDS signals), and for receiving the data (LVDS) from 8 digitizers boards (2 TILEDMU with 3 channels each), deserialize it and send it through an optical link to the input stage of the RODs.

The actual design of the interface links is based in S-LINK protocol over HP G-Link chips as a physical layer. The implementation is an integrated G-LINK LSC card (640Mbit/s) working at 16 bits in 40 MHz mode (chip HDMP1032). Dual channel readout is implemented for providing redundancy.

The data format of one event to be sent to RODs in case of seven samples is in the next table Table 7[1]:

Word	Contents
1	*Control word* (**0x11111110**)
2	*Fragment ID*
3	*Header* from **DMU1**
4	*Data word*, high/low gain **sample 1** from **DMU1**
5	*Data word*, high/low gain **sample 2** from **DMU1**
6	*Data word*, high/low gain **sample 3** from **DMU1**
7	*Data word*, high/low gain **sample 4** from **DMU1**
8	*Data word*, high/low gain **sample 5** from **DMU1**
9	*Data word*, high/low gain **sample 6** from **DMU1**
10	*Data word*, high/low gain **sample 7** from **DMU1**
11	*CRC word* from **DMU1**
12	*Header* from **DMU2**
13	*Data word*, high/low gain **sample 1** from **DMU2**
14	*Data word*, high/low gain **sample 2** from **DMU2**
15	*Data word*, high/low gain **sample 3** from **DMU2**
16	*Data word*, high/low gain **sample 4** from **DMU2**
17	*Data word*, high/low gain **sample 5** from **DMU2**
18	*Data word*, high/low gain **sample 6** from **DMU2**
19	*Data word*, high/low gain **sample 7** from **DMU2**
20	*CRC word* from **DMU2**
.....	.....
138	*Header* from **DMU16**
139	*Data word*, high/low gain **sample 1** from **DMU16**
140	*Data word*, high/low gain **sample 2** from **DMU16**
141	*Data word*, high/low gain **sample 3** from **DMU16**
142	*Data word*, high/low gain **sample 4** from **DMU16**
143	*Data word*, high/low gain **sample 5** from **DMU16**
144	*Data word*, high/low gain **sample 6** from **DMU16**
145	*Data word*, high/low gain **sample 7** from **DMU16**
146	*CRC word* from **DMU16**
147	*Control word* (**0xFFFFFFF0**)

Table 7 Interface board deserialized data

Note:

[1] The data and header format could be seen at Table 4 and Table 5 respectively.

- Fragment ID is the number of drawer. A proposal for final ATLAS is something like this:

Detector	Fragment ID
Negative Barrel	0x000 ... 0x03F
Positive Barrel	0x100 ... 0x13F
Negative Ext.Bar.	0x200 ... 0x23F
Positive Ext.Bar.	0x300 ... 0x33F

1.4 Input bandwidth

The bandwidth needed for this configuration of seven samples of ten bits at 100kHz level1 trigger rate is: 100kHz * 146words * 32bits/word = **467,2 Mbit/sec** (link limit is *640Mbit/sec*, 27% BW not used). Other data rates are shown in table 8 for different FEB configurations.

	½ Barrel or 1 Extended Barrel Module					
Number of PMTs	Up to 48					
Bits per sample	10					
Samples	15	13	11	9	7	5
Number of words	275	243	211	179	147	115
Needed BW in Mbits/sec (ATLAS rate 100kHz)	**840**	**742**	**644**	**547**	**449**	**351**
LSC S-link BW (Mbits/sec)	1221					

Table 8 Interface link bandwidths

The raw data event size is **147** 32bit words with seven samples at atlas level 1 trigger of 100kHz per drawer. The tile calorimeter has 256 front end electronics drawers, then **144Kbytes** are sent each 100Khz. The data rate for tilecal raw data is **14,6Mbytes/sec**.

Because of the serializer, the ½ Barrel (45ch) bandwidth and the extended barrel bandwidth (32ch) will be the same. This method could send up to 48 channels, thus 3ch[2] and 16ch[3] are not used for ½ central barrel and the extended one respectively.

[2] 1 chip only sends 1ch
[3] 5 chips aren't used and 1 chip sends 2ch instead of 3.

2 Output Data Format

The motherboard has several Processing Units (actually, up to four). Each PU processes a precise number of channels (one, two or four drawers depending on the processing power). One PU has information about its own channels, only. Because of this we need to have a PU data format, and a global data format for the Motherboard and all PUs.

- The **Intermediate Data Format** at the output of each Processing Unit (PU).
- The **ROD Data Format** which is the union of the data of four PUs with the DAQ-1 event format according to ATLAS DAQ requirements.

2.1 Intermediate data format. PU event fragment

Each PU formats the data into four 32-bit word blocks as shown bellow in Table 9, framed with one header word which contains the total size of the event[4] and an end of event marker word which has a fixed value 0x0EOE.

Total number of words in the event (=sum of all blocks)	
Block #0	Control and status word (variable length)
Block #1	Energy sums (variable length)
Block #2	Energy per channel (variable length)
Block #3	Time and quality of fit information (variable length)
Block #4	Raw data (variable length)
1	CRC Word
End of event marker (0x00000EOE)	

Table 9 Intermediate PU ROD event data format.

In the following a detailed description of each data block is given.

In case of errors it is sufficient that the PU sends only the first block with the appropriate flags.

[4] Only the data words should be counted, without including the trailer 0xEOE word. This is for the right behaviour of the OC.

2.1.1 **Variable block #0**

It contains the header and control word information and will be present in all the events. See Table 10.

	31 … 24	23 … 16	15 … 13	12 … 11	8 … 5	4 … 0
1	Nb words in block #4	Nb words in block #3	Nb words in block #2		Nb words in block #1	Nb words in block #0
2	Number of channels per PU	FEB Drawer Number	ND	Mode	TTC_BCID[5]	
3	EM	Status Flags				
4	Chip Error Flags 2nd drawer[7]		Chip Error Flags 1st drawer			
5	Chip Error Flags 3rd drawer[8]		Chip Error Flags 4th drawer[9]			
6	Mask Channels 31…0					
7	Mask Channels 63…32[6]					
8	Mask Channels 95…64[7]					
9	Mask Channels 127…96[8]					
10	Mask Channels 160…128[9]					
11	Extra info in block #3 channels 31…0					
12	Extra info in block #3 Channels 63…32[6]					
13	Extra info in block #3 Channels 95…64[7]					
14	Extra info in block #3 Channels 127…96[8]					
15	Extra info in block #3 Channels 160…128[9]					
16	Extra info in block #4 channels 31…0					
17	Extra info in block #4 Channels 63…32[6]					
18	Extra info in block #4 Channels 95…64[7]					
19	Extra info in block #4 Channels 127…96[8]					
20	Extra info in block #4 Channels 160…128[9]					

Table 10 Variable block #0

Note:

- First Word: indicates the number of words in each block.

Bits 0…4	Nb words in block #0	4 bits (0…32 words)
Bits 5…8	Nb words in block #1	4 bits (0…16 words)
Bits 9…15	Nb words in block #2	7bits (0…128 words)
Bits 16…22	Nb words in block #3	7bits (0…128 words)
Bits 23…31	Nb words in block #4	9 bits (0…512 words)

[5] This Bunch Cross ID is the one received at PU level in two ways, the TTC system and the FEBoards data. This value is tested for error synchronisation.
[6] This word only appears if the processed drawer is a CB (45 channels)
[7] This word only appears when processing a CB+EB with one PU (45+32 channels)
[8] When processing two CB in one PU (90 channels)
[9] Only when processing four drawers (2EB+2CB) in one PU (154 channels)

- Second word: contain the information summarized in next table

Bits 0…11	12 bits counter Bunch crossing ID			12 bits
Bits 12,13	Mode			2 bits
		Normal	00	
		Calibration	01	
		Test	10	
		Not Used	11	
Bits 14,15	ND: Number of drawer processed by one PU			2 bits
		One Drawer	00	
		Two Drawers	01	
		Three Drawers	10	
		Four Drawers	11	
Bits 16…23	FEB Drawer Number: specifies the ID of the drawer processed by this PU. Is the same as Fragment ID from FEB			8bits (0…255)
Bits 24…31	Number of channels per PU: the PMT channels that this PU is processing			8 bits

- The StatusFlags bit-field needs to be defined, and it could depend of the final hardware of the RODs. The highest bit **EM** (energy mode) indicates which Energy packing option is selected, then, if EM=1 the Energy, time, and χ^2 is packaged in one 32 bits word (and always are sent time and χ^2); and if EM=0 the Energy is sent always packaged two channels Energy in one 32 bits words, and the time and χ^2 is sent only if the energy is above a threshold. For more details see section "2.1.3 Variable block #2".
- Chip Error Flags. Indicates which TileDMU chip has flagged an error (There are up to 16 chips per drawer, with 3 ADCS each). Setting a bit to 1 indicates that an error has been flagged from the motherboards.
- The Mask bit field indicates which of the PMT channels for the PU are masked from the readout. Setting the corresponding bit to 1 indicates a valid channel, and 0 a masked channel.
- Extra info: setting a bit to 1 in these words indicates that additional information is provided for that channel in the specified block. Extra info in block #3[10] indicates that the energy is over threshold 1 for this channel (time+chi2 are sent), in block #4 means the energy is over threshold 2 for this channel (raw data is sent).
- This block is variable depending on the number of drawers processed. See footnotes 5, 6, 7 and 8.

2.1.2 Variable block #1

It contains the energy sum information of the trigger towers cells, present for all the events. At the PU level, it's only possible to relate the information of the channels of the drawers processed, thus we should have more tower cells energy information if we process one CB and EB in one PU. If the PU has enough processing power this will aid to reduce the number of PUs in the project and therefore the cost. This info could be used for debugging of level1 trigger.

[10] This is the case EM=0. If EM=1 then the Energy, time and chi2 are sent always.

	31	
1	Energy sum tower η = 0.1	CENTRAL BARREL
2	Energy sum tower η = 0.2	
3	Energy sum tower η = 0.3	
4	Energy sum tower η = 0.4	
5	Energy sum tower η = 0.5	
6	Energy sum tower η = 0.6	
7	Energy sum tower η = 0.7	
8	Energy sum tower η = 0.8	
9	Energy sum tower η = 0.9	
10	Energy sum tower η = 1.0	
11	Energy sum tower η = 1.1	EXTENDED BARREL
12	Energy sum tower η = 1.2	
13	Energy sum tower η = 1.3	
14	Energy sum tower η = 1.4	
15	Energy sum tower η = 1.5	
16	Energy sum tower η = 1.6	

Table 11 Variable block #1

2.1.3 Variable block #2

In this block, two options are offered for giving flexibility to the data format of the Energy, waiting for a final option that indeed could be taken dynamically. This two options are depending of the header bit **EM**. If EM=0 then is selected the *2.1.3.1* option, and if EM=1 the packaging indicated in *2.1.3.2* is selected.

The Energy is selected as a 15 bit unsigned integer. Having in mind that the samples are 10 bits, if we combine information from multiple samples, N, in calculating the energy, we improve the resolution by 1/sqrt(N). For example, if we read out 9 samples, we should improve the energy calculation by a factor 3, requiring 10 + 2 = 12 bits. If we allow for a sign bit, then we're at 13 bits, so the suggested 15 bits will be enough for the worst case of having up to 16 samples.

2.1.3.1 Variable block #2 with EM=0

It contains the energy information, and will be present for all the events in the format shown in Table 12 (two channels packaging in 32bits).

	31	30 … 16	15	14 … 0
1	G	Energy ch 2	G	Energy ch 1
2	G	Energy ch 4	G	Energy ch 3
…		……		
23	G	Energy ch 46	G	Energy ch 45
…		……		
39	G	Energy ch 78	G	Energy ch 77
…		……		
77	G	Energy ch 154	G	Energy ch 153

Table 12 Variable block #2 EM=0

Note:

- Fixed size of the block for 45 channels is **23 words**. In case of Extended barrels FEBoards (32 channels) will be 16 words deep. 39 words is the size of this block when processing a CB+EB (45+32 ch) in the same PU. The maximum size is 77 words in the case that four drawers are processed with one PU (154ch).
- Energy is 15bit unsigned integer.
- **G** is the gain bit. **1** for high gain and **0** for low gain.

2.1.3.2 Variable block #2 with EM=1

This mode allows to send Energy, time, and χ^2 always. If our idea is to send always the Energy, and only time and chi2 when the energy of this channels is above a threshold the previous option (EM=0) is better (Table 13). But if we plan to send always the Energy, time, and χ^2 then this is more optimised (in terms of number of words). Obviously, with this option the Variable block #3 will be empty.

	31	30 26	25 16	15	14 0
1	P	Fit ch 1	Time ch 1	G	Energy ch 1
2	P	Fit ch 2	Time ch 2	G	Energy ch 2
3	P	Fit ch 3	Time ch 3	G	Energy ch 3
…		……			
45	P	Fit ch 45	Time ch 45	G	Energy ch 45
…		……			
77	P	Fit ch 77	Time ch 77	G	Energy ch 77
…		……			
154	P	Fit ch 154	Time ch 154	G	Energy ch 154

Table 13 Variable block #2 EM=1

Note:

- The size of the block in 32 bit words is the same as the number of channels processed in each PU.
- Energy is 15bit unsigned integer.
- G is the gain bit. 1 for High gain and 0 for low gain.
- Time is 10 bits signed integer
- Fit 2 is the chi2 with 5 bits unsigned integer precision.
- P is a reserved bit, probably used for parity.

2.1.4 Variable block #3

It contains the time and quality of fit information, and will be present only in the case there are channels with energy above threshold. Obviously, only available if EM=0. See Table 14.

	31	30 26	25 16	15	14 11	9 0
1	P	Fit Ch 2	Time Ch 2	R	Fit Ch 1	Time Ch 1
2	P	Fit Ch 2	Time Ch 2	R	Fit Ch 1	Time Ch 1
…		……				
23	P	Fit Ch 46	Time Ch 46	R	Fit Ch 45	Time Ch 45
…		……				
39	P	Fit Ch 78	Time Ch 78	R	Fit Ch 77	Time Ch 77
…		……				
77	P	Fit Ch 154	Time Ch 154	R	Fit Ch 153	Time Ch 153

Table 14 Variable Block #3

Note:

- The **maximum** total length of this block is 23. In case of Extended barrels FEBoards (32 channels) would be 16. 39 words is the size of this block when processing a CB+EB (45+32 ch) in the same PU. The maximum size is 77 words in the case that four drawers are processed with one PU (154ch).
- The time here is a 10 bit signed integer. Our resolution on the time calculation from data is of order 0.1ns; then taking 50ps (more than needed), if we only decide to measure time within ± 25 ns, then $\frac{25ns}{50ps} = 500 \Rightarrow 2^9 + sign\_bit$ give us the idea that 9 bits plus a sign bit is enough for time measurement. A total of 10 bits are used for this purpose.

  If, however, we decide to measure timing over N*25 ranges, then we would need to include additional bits to give the time slice in 25 ns bins, e.g. for 7 samples: 9+3 = 12 bits.
- The fit will be a 5-bit integer. It's enough a 5 bit word because the fit only give us an estimation of the quality of the signal, then, a range of 0-32 it's enough to define a fit scale (0 will be a perfect fit, and 32 a very bad fit). In addition one "fit out of range" bit will be provided with the energy indicating a very bad fit for the corresponding channel. Thus 32 ranges maybe expressed as a % error on Energy.

2.1.5 Variable block #4

This block contains the raw data, or the samples taken directly from the front-end digitizers to next level system. This is done when an interesting event occurs, en this happens when the Energy of a channel is above a threshold predefined for sending raw data. For the formatting see Table 15.

We must define, in case of an interesting event, when we must send raw data for the channel above a threshold, for a channels and adjacent channels, or for all the channels in a drawer. Now we suppose only to send raw data for channels above this threshold, because if we send raw data for all channels a more optimised format could be used (e.g. serialized, not loosing the zeros info, etc). Nevertheless some other policies should be defined, for example to set a look-up table with different threshold for each calorimeter cell.

	31	30	29 20	19 10	9 0
1	r	R	Sample 3 Ch 1	Sample 2 Ch 1	Sample 1 Ch 1
2	r	R	Sample 6 Ch 1	Sample 5 Ch 1	Sample 4 Ch 1
3	r	R	00 0000 0000	00 0000 0000	Sample 7 Ch 1
4	r	R	Sample 3 Ch 2	Sample 2 Ch 2	Sample 1 Ch 2
5	r	R	Sample 6 Ch 2	Sample 5 Ch 2	Sample 4 Ch 2
6	r	R	00 0000 0000	00 0000 0000	Sample 7 Ch 2
…			……		
94	r	R	Sample 3 Ch 32	Sample 2 Ch 32	Sample 1 Ch 32
95	r	R	Sample 6 Ch 32	Sample 5 Ch 32	Sample 4 Ch 32
96	r	R	00 0000 0000	00 0000 0000	Sample 7 Ch 32
…			……		
133	r	R	Sample 3 Ch 45	Sample 2 Ch 45	Sample 1 Ch 45
134	r	R	Sample 6 Ch 45	Sample 5 Ch 45	Sample 4 Ch 45
135	r	R	00 0000 0000	00 0000 0000	Sample 7 Ch 45
…			……		
229	r	R	Sample 3 Ch 77	Sample 2 Ch 77	Sample 1 Ch 77
230	r	R	Sample 6 Ch 77	Sample 5 Ch 77	Sample 4 Ch 77
231	r	R	00 0000 0000	00 0000 0000	Sample 7 Ch 77
…			……		
460	r	R	Sample 3 Ch 154	Sample 2 Ch 154	Sample 1 Ch 154
461	r	R	Sample 6 Ch 154	Sample 5 Ch 154	Sample 4 Ch 154
462	r	R	00 0000 0000	00 0000 0000	Sample 7 Ch 154

Table 15 Variable block #4

Note:

- With 7 samples the **maximum** length of this block is 135 words in the case of central barrels. For extended barrels (32 ch) will be 96 words.
- If one EB+CB is processed in this PU, then the number of words is 231. And the maximum processing 4 drawers is 462 words.

2.1.6 PU event fragment length

In Table 16 and Table 17 are shown the size of the different data blocks in the PU event fragment for different EM values. We must consider two cases, when EM=0 (energy and chi2+time are sent in different blocks) and with EM=1 (Energy, time, and chi2 are packaged in one 32 bits word).

EM = 0

Block	Detector Block	Length	Typical Size 1	Typical Size 2	Typical Size 3	Maximum Size 1	Maximum Size 2	Maximum Size 3
-	Header	-	1	1	1	1	1	1
#0	Control and Status Words	Variable	10	14	20	10	14	20
#1	Energy sums	Variable	10	16	16	10	16	16
#2	Energy	Variable	23	39	77	23	39	77
#3	Time & Quality of Fit	Variable	3	4	8	23	39	77
#4	Raw Data Information	Variable	2	3	5	135	231	462
-	CRC word	fixed	1	1	1	1	1	1
-	End of event marker	-	1	1	1	1	1	1
Total			**49**	**77**	**127**	**202**	**340**	**653**

Table 16 The maximum and typical event size of the Intermediate Data Format for one PU in units of 32-bit words EM=0

EM = 1

Block	Detector Block	Length	Typical Size 1	Typical Size 2	Typical Size 3	Maximum Size 1	Maximum Size 2	Maximum Size 3
-	Header	-	1	1	1	1	1	1
#0	Control and Status Words	Variable	10	14	20	10	14	20
#1	Energy sums	Variable	10	16	16	10	16	16
#2	Energy	Variable	45	77	154	45	77	154
#3	Time & Quality of Fit	Variable	0	0	0	0	0	0
#4	Raw Data Information	Variable	2	3	5	135	231	462
-	CRC word	fixed	1	1	1	1	1	1
-	End of event marker	-	1	1	1	1	1	1
Total			**68**	**111**	**196**	**201**	**339**	**653**

Table 17 The maximum and typical event size of the Intermediate Data Format for one PU in units of 32-bit words EM=1

Note:

- The typical size 1 is calculated assuming that ~10% of the channels have energy above threshold for which the time and chi2 are sent, ~1% of the channels have high energy for which the raw data need to be available. The header and end of event marker words will not be transmitted in the output. Considering each PU process one CB drawer 45 channels with seven samples of ten bits
- The typical size 2 is calculated assuming all the above but processing two drawer per PU (45ch+32ch).
- The typical Size 3 is calculated assuming all the typical size 1 but with one PU processing four drawers (154ch).
- The Maximum Size 1 is calculated assuming that one PU process one CB drawer 45 channels with seven samples of ten bits and sending raw data for all 45 channels
- The Maximum Size 2 is calculated assuming all the maximum size 1 but with one PU processing CB+EB (77 ch)
- The Maximum Size 3 is calculated assuming all the maximum size 1 but with one PU processing four drawers (154ch).

In Figure 3 and Figure 4 the sizes per block are shown for EM=0 and EM=1. The conclusion that we get has very similar results when all time and chi2 are sent always for the two cases, but when only time and chi2 are sent above a threshold, then it is preferred the packaging with EM=0.

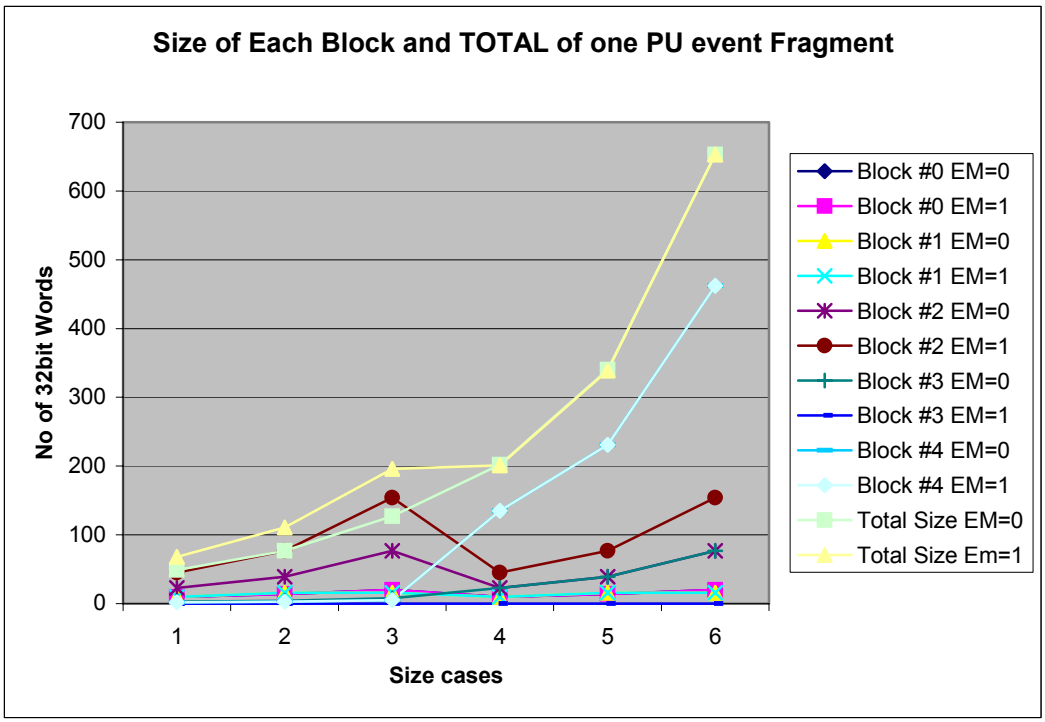


Figure 3 Size of Blocks and Total of PU fragment

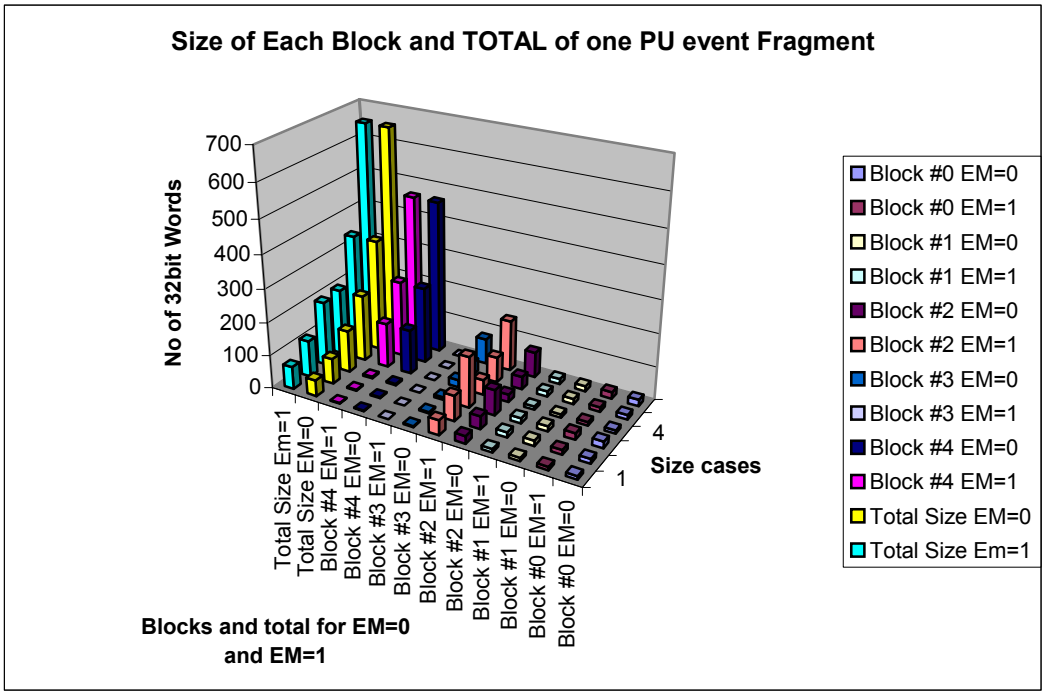


Figure 4 Size of Blocks and Total of PU fragment (3D Diagram)

2.2 ROD event fragment format

The output controller has to construct complete event fragments, according to the ATLAS DAQ/EF Prototype –1, event format [1] . In principle each rod motherboard has up to four PUs, then a union of four intermediate PU data format described above plus a standard header, trailer and status flags will conform the total ROD event fragment (Table 18).

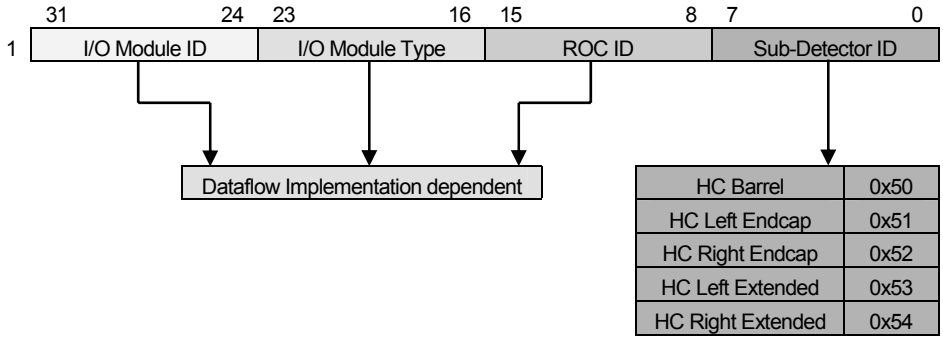
		Nb. Words	31		0
		0	**Beginning of fragment (0x00000B0F)**		
Header		1	Start of header marker (0xEEEEEEEE)		
		1	Header size (0x20 in bytes)		
		1	Format version number		
		1	Source Identifier		
		1	Level 1 ID		
		1	Bunch crossing ID[11]		
		1	Level 1 Trigger Type		
		1	Detector Event Type		
Detector Data	Status word	1	Nb of PU	PU Mask	Detector Format Version Number
	Processing Unit 1	Nb. #0	Variable Block #0		
		Nb. #1	Variable Block #1		
		Nb. #2	Variable Block #2		
		Nb. #3	Variable Block #3		
		Nb. #4	Variable Block #4		
	Processing Unit 2	Nb. #0	Variable Block #0		
		Nb. #1	Variable Block #1		
		Nb. #2	Variable Block #2		
		Nb. #3	Variable Block #3		
		Nb. #4	Variable Block #4		
	Processing Unit 3	Nb. #0	Variable Block #0		
		Nb. #1	Variable Block #1		
		Nb. #2	Variable Block #2		
		Nb. #3	Variable Block #3		
		Nb. #4	Variable Block #4		
	Processing Unit 4	Nb. #0	Variable Block #0		
		Nb. #1	Variable Block #1		
		Nb. #2	Variable Block #2		
		Nb. #3	Variable Block #3		
		Nb. #4	Variable Block #4		
	Status Word	1	Status flag from Output Controller		
Trailer		1	Number of Status Elements		
		1	Number of Data Elements		
		1	Status Block Position		
		0	**End of Fragment (0x0E0F)**		

Table 18 ROD Board complete output data format

[11] This Bunch Cross ID is the one received from TTC system and it's not compared with the one received from the front-end bards. Only the processing units know the TTC info coming from FEB and from TTCrx. Thus they are responsible for error synchronisation detection.

Note:

- The **Format Version number** is a 32-bit integer number loaded by VME at configuration time. It defines the version of the ROD data fragment header, not the format version of the detector data.
- The **Source Identifier** is the word that defines the fragment. It is sub-divided as shown below:



The exact implementation of these words will depend on the detector readout organization. Note that the combination of these fields allows a unique Source ID across all subdetectors.

With a configuration of four crates ROC ID range is 0x0 to 0x3. A lot of bits are not used but the compatibility with DAQ-1 event data format must be respected.

The I/O Module ID and I/O Module type must be defined for the final hardware implementation, but in principle there are enough bits to define all the possibilities that should be needed, then they will be defined during the implementation of the DataFlow.

- **Level 1 ID**: The event identifier generated by the Level 1 trigger system. 24 bits
- **Bunch Crossing ID**: The bunch crossing identifier generated by the level 1 trigger system. 12 bits
- **Level 1 Trigger Type**: The event type transmitted by the level 1 system. 8 bits

- **Detector event type**: This element identifies an event which may have been generated by a sub-detector, independent of other sub-detectors and the ATLAS trigger systems.

Physics	1
Laser	2
Pedestal	4
CIS	8
Cesium	16

- The **Detector Event Type** this is a sub-detector specific event type flag. It will be send by VME at configuration time.

- **Nb of PU** indicates the number of PU used for data processing.
- **PU Mask** is the 8-bit pattern with defines which of the PU of the ROD board is read out. The PUs are numbered from 1 to N with #1 the one on the top of the ROD board.
- **Detector Format Version Number** indicates the format version for the detector data block.
- **Number of status elements**, which gives the total length of the detector status block.
- **Number of data elements**, which gives the sum of the detector blocks #1 to #4.
- **Status block position**, which is set to 0, to indicate that the status block precedes the data, as it is in this case.
- The **Status flag from Output Controller** is used to mark events where the data from the PU are not correctly read. For example when there is a mismatch between the number of words to read from a PU and the end of event marker.

2.3 Output bandwidth

The output bandwidth for the cases considered are shown in Table 19 for EM=0, and for EM=1 in Table 20. Note that it is assumed the level 1 trigger rate of 100Khz.

EM = 0							
Event Fragment Block	**Length**	**Typical Size 1 (words*4PUs)**	**Typical Size 2 (words*2PUs)**	**Typical Size 3 (words*1PUs)**	**Maximum Size 1(words*4PUs)**	**Maximum Size 2 (words*2PUs)**	**Maximum Size 3(words*1PUs)**
Fragement Header	Fixed	8	8	8	8	8	8
Status word	Fixed	1	1	1	1	1	1
Detector Block #0	Variable	40	28	20	40	28	20
Detector Block #1	Variable	40	32	16	40	32	16
Detector Block #2	Variable	92	78	77	92	78	77
Detector Block #3	Variable	12	8	8	92	78	77
Detector Block #4	Variable	8	6	5	540	462	462
Status word	Fixed	1	1	1	1	1	1
Trailer	Fixed	3	3	3	3	3	3
Total number of Words/event		**205**	**165**	**139**	**817**	**691**	**665**
Output BW (Mbits/sec)		**656,00**	**528,00**	**444,80**	**2614,40**	**2211,20**	**2128,00**

Table 19 Output bandwith with EM=0

EM = 1							
Event Fragment Block	**Length**	**Typical Size 1 (words*4PUs)**	**Typical Size 2 (words*2PUs)**	**Typical Size 3 (words*1PUs)**	**Maximum Size 1(words*4PUs)**	**Maximum Size 2 (words*2PUs)**	**Maximum Size 3(words*1PUs)**
Fragement Header	Fixed	8	8	8	8	8	8
Status word	Fixed	1	1	1	1	1	1
Detector Block #0	Variable	40	28	20	40	28	20
Detector Block #1	Variable	40	32	16	40	32	16
Detector Block #2	Variable	180	154	154	180	154	154
Detector Block #3	Variable	0	0	0	0	0	0
Detector Block #4	Variable	8	6	5	540	462	462
Status word	Fixed	1	1	1	1	1	1
Trailer	Fixed	3	3	3	3	3	3
Total number of Words/event		**281**	**233**	**208**	**813**	**689**	**665**
Output BW (Mbits/sec)		**899,20**	**745,60**	**665,60**	**2601,60**	**2204,80**	**2128,00**

Table 20 Output bandwidth with EM=1

Note:

- Typical sizes and maximum sizes configurations are the same as described before.

Consider that the actual output link to the ROB is based on the integrated ODIN LSC of the TM4Plus1 transition module which bandwidth limit is 1280 Mbits/sec.

In Figure 5 and Figure 6 we represent these tables. The conclusions of both are the same. The two options for sending the Energy EM=0 and EM=1 are equivalent when sending always time, energy and chi2, but EM=0 is a better election when only the energy is sent, and the time and chi2 when the energy for this channels is above the predefined threshold.

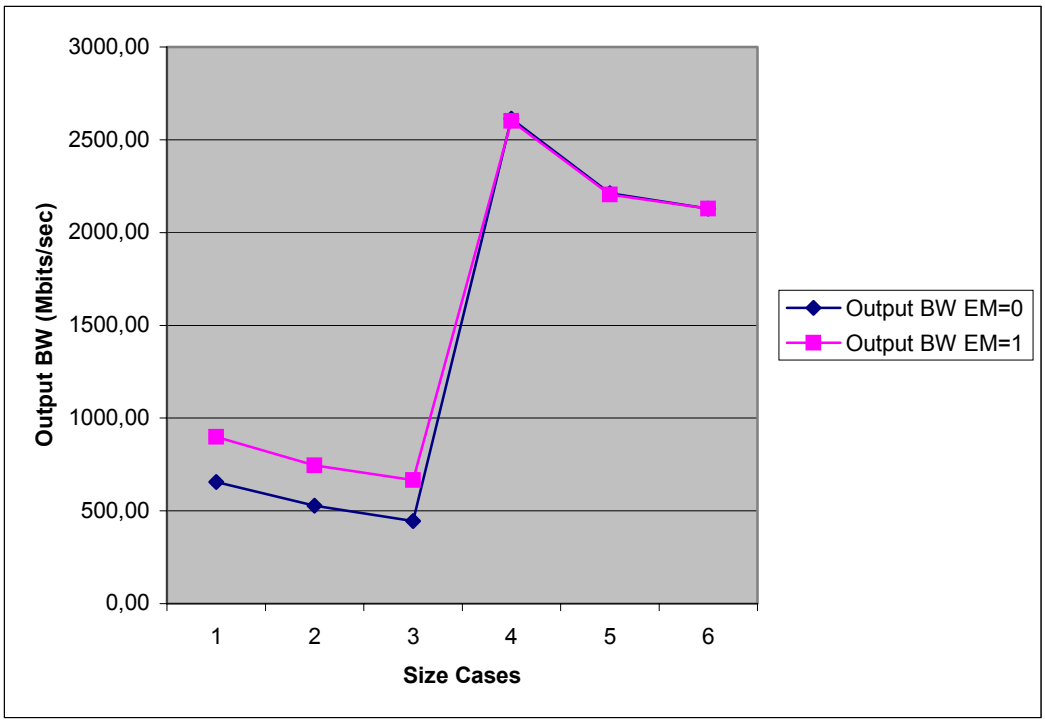


Figure 5 Output BW comparison with EM=0 and EM=1

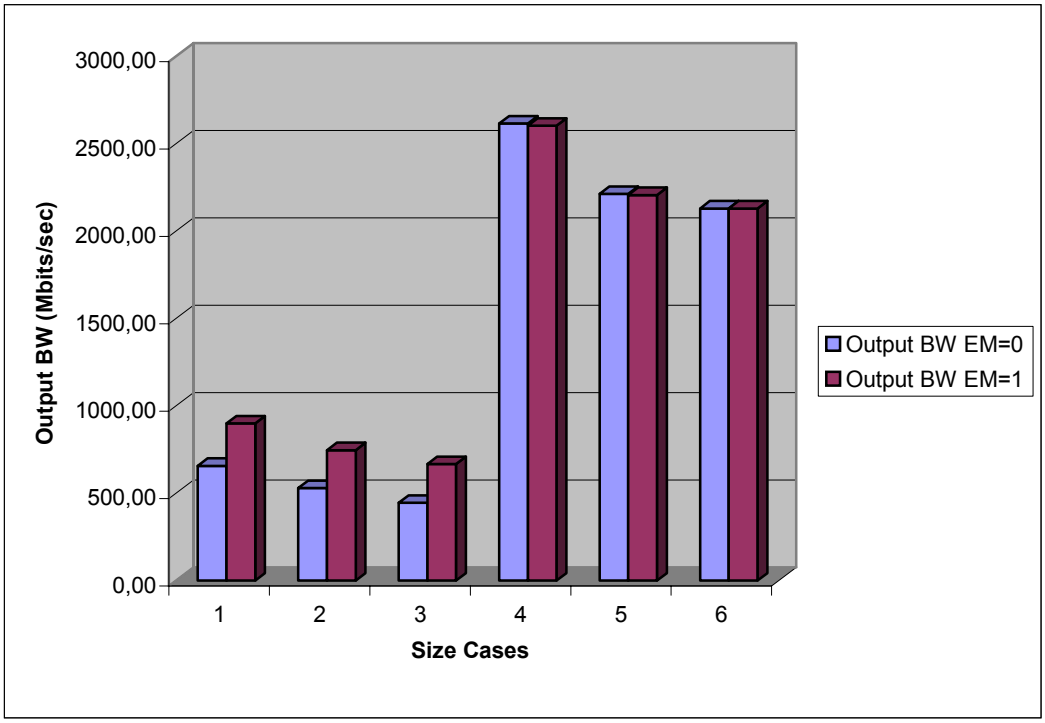


Figure 6 Output BW comparison with EM=0 and EM=1 (3D diagram)

2.4 Tilecal ROD requirements in numbers

The tilecal ROD requirements are summarized in Table 21.

Number of Channels	9856
Number of Drawers (FEB)	256
Number of channels per drawer (EB)	32
Number of channels per drawer (CB)	45
Number of Drawers (EB)	128
Number of Drawers (CB)	128
Input event size per FEB (7 samples) in kbytes	0,57
Input Data Bandwidth @ 100kHz Lvl1 ATLAS rate Gbytes/sec	14,02
Number of Drawers (FEB) per ROD	4
Number of RODs	64
Typical output event size per ROD (Typical Size 1) in kbytes	1,10
Output Data Bandwith @ 100kHz Lvl1 ATLAS rate Gbytes/sec	6,70
Number of PUs per ROD	4
Number of PU (DSP) instructions per channel (seven samples). Aplying Optimal filtering (E, t and chi2)	70
Total processing power in MIPs	68992

Table 21 Tilecal ROD numbers

2.5 Calibration Data Format

The ROD Controller local CPU will treat the calibration data for getting the new optimal filtering weights. The ROD could do the first pass in the analysis of the calibration data. This previous treatment with DSPs could aid the ROD crate controller on its calculations for a fastest calibration. It consists in calculate and send the mean, rms and number of events read for each channel for each step in the calibration process[12].

[12] A "*step*" is defined with a set of parameters, which are loaded in the calibration board. The ROD will know at boot time how many events are expected per step and should calculate the mean and rms of the channels for this number of events.

In this case the proposed format per PU is shown in Table 22.

	32		15	
0	Nb of events	Nb Step	Status Flags	PU ID
1	Sum E Channel 1 Sample 1		Sum E^2 Channel 1 Sample 1	
2	Sum E Channel 1 Sample 2		Sum E^2 Channel 1 Sample 2	
3	Sum E Channel 1 Sample 3		Sum E^2 Channel 1 Sample 3	
4	Sum E Channel 1 Sample 4		Sum E^2 Channel 1 Sample 4	
5	Sum E Channel 1 Sample 5		Sum E^2 Channel 1 Sample 5	
6	Sum E Channel 1 Sample 6		Sum E^2 Channel 1 Sample 6	
7	Sum E Channel 1 Sample 7		Sum E^2 Channel 1 Sample 7	
…	……...			
218	Sum E Channel 32 Sample 1		Sum E^2 Channel 32 Sample 1	
219	Sum E Channel 32 Sample 2		Sum E^2 Channel 32 Sample 2	
220	Sum E Channel 32 Sample 3		Sum E^2 Channel 32 Sample 3	
221	Sum E Channel 32 Sample 4		Sum E^2 Channel 32 Sample 4	
222	Sum E Channel 32 Sample 5		Sum E^2 Channel 32 Sample 5	
223	Sum E Channel 32 Sample 6		Sum E^2 Channel 32 Sample 6	
224	Sum E Channel 32 Sample 7		Sum E^2 Channel 32 Sample 7	
…	……...			
309	Sum E Channel 45 Sample 1		Sum E^2 Channel 45 Sample 1	
310	Sum E Channel 45 Sample 2		Sum E^2 Channel 45 Sample 2	
311	Sum E Channel 45 Sample 3		Sum E^2 Channel 45 Sample 3	
312	Sum E Channel 45 Sample 4		Sum E^2 Channel 45 Sample 4	
313	Sum E Channel 45 Sample 5		Sum E^2 Channel 45 Sample 5	
314	Sum E Channel 45 Sample 6		Sum E^2 Channel 45 Sample 6	
315	Sum E Channel 45 Sample 7		Sum E^2 Channel 45 Sample 7	
…	……...			
533	Sum E Channel 77 Sample 1		Sum E^2 Channel 77 Sample 1	
534	Sum E Channel 77 Sample 2		Sum E^2 Channel 77 Sample 2	
535	Sum E Channel 77 Sample 3		Sum E^2 Channel 77 Sample 3	
536	Sum E Channel 77 Sample 4		Sum E^2 Channel 77 Sample 4	
537	Sum E Channel 77 Sample 5		Sum E^2 Channel 77 Sample 5	
538	Sum E Channel 77 Sample 6		Sum E^2 Channel 77 Sample 6	
539	Sum E Channel 77 Sample 7		Sum E^2 Channel 77 Sample 7	
…	……...			
1072	Sum E Channel 154 Sample 1		Sum E^2 Channel 154 Sample 1	
1073	Sum E Channel 154 Sample 2		Sum E^2 Channel 154 Sample 2	
1074	Sum E Channel 154 Sample 3		Sum E^2 Channel 154 Sample 3	
1075	Sum E Channel 154 Sample 4		Sum E^2 Channel 154 Sample 4	
1076	Sum E Channel 154 Sample 5		Sum E^2 Channel 154 Sample 5	
1077	Sum E Channel 154 Sample 6		Sum E^2 Channel 154 Sample 6	
1078	Sum E Channel 154 Sample 7		Sum E^2 Channel 154 Sample 7	

Table 22 The PU event fragment for the calibration events considering 7 samples

Note:

- With 7 samples the **maximum** length of this block is 315 words in the case of central barrels. For extended barrels (32 ch) will be 224 words.
- If one EB+CB is processed in this PU, then the number of words is 539. And the maximum processing 4 drawers is 1078 words.

The Output controller will read the data from each four PU and send them via the VME interface into ROD Controller SBC.

The event fragment size for 4 PU per ROD in the case of 7 samples will be 1078 32-bit words corresponding to 4,21 kBytes. Assuming a VME bus speed of 10 MB/s this will require 0.42ms to transfer the pre-processed data to local controller

3 Conclusions

4 Acknowledgements

5 Acronyms

ADC : Analog to digital converter

BW : Bandwidth

CB : Central barrel

DAQ : Data Acquisition

EB : Extended barrel

FEB : Front end boards

ITC : Intermediate Tile Calorimeter

LDC : Link destination card

LSB : Less significative bit

LSC : Link source card

LVDS : Low voltage differential signal

MSB : More significative bit

PMT : Photomultiplier

PU : Processing unit

ROL : Read Out Link

SBC : Single board computer

TileDMU : Tile data management unit

TTC : Timing and trigger control

TTCrx : TTC receiver ASIC

6 References

[1] **The event format in the ATLAS DAQ/EF prototype –1**. *Authors*: C. Bee, O. Boyle, D. Francis, L. Mapelli, R. McLaren, G. Mornacchi, J. Petersen. *Note Number*: 050. *Version*: 1.5. *Date*: 15-10-98. *Reference*: http://atddoc.cern.ch/Atlas/Notes/050/Note050-1.html

[2] **TRIGGER & DAQ INTERFACES WITH FRONT-END SYSTEMS: REQUIREMENT DOCUMENT**. Authors: Atlas Trigger-DAQ Steering Group.

Note Number: 103. *Version*: 2.5 (draft). *Date*: 9-6-98.

Reference: **http://atlasinfo.cern.ch/Atlas/GROUPS/DAQTRIG/DIG/archive/document/FEdoc_2.5.pdf**

[3] **Definitions, acronyms and abbreviations in ATLAS DAQ/EF prototype -1**. *Author:* D. Francis. *Note Number*: NoteNumber : 046. *Version* : 1.1 . *Date*: 20/10/97.

Reference: http://atddoc.cern.ch/Atlas/Notes/046/Note046-1.html

[4] *Reference*: **http://ific.uv.es/tical/rod_new/IndexROD.htm**

[5] *Reference*: **http://hep.uchicago.edu/atlas/electr/electronics.html**

[6] *Reference*: **http://www.physto.se/~ker/designreview/dr.html**

[7] *Reference*: **http://atlasinfo.cern.ch/Atlas/GROUPS/DAQTRIG/DIG/Welcome.html**

[8] *Reference*: **http://atlasinfo.cern.ch/Atlas/SUB_DETECTORS/TILE/elec/electronics.html**