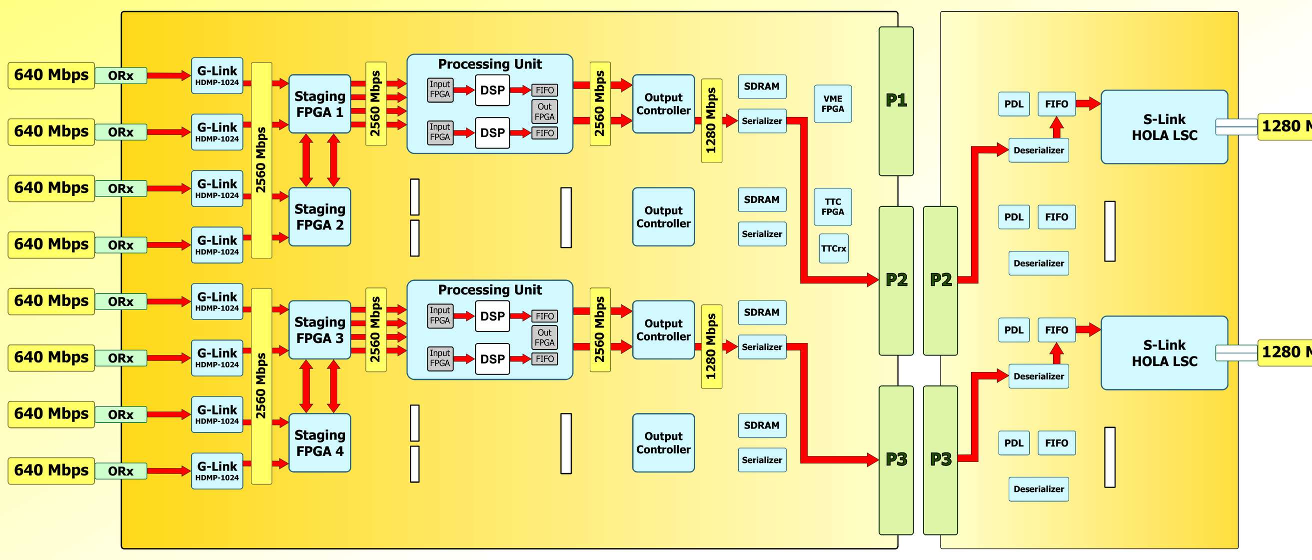
LOSSLESS COMPRESSION OF ATLAS TILE CALORIMETER DRAWER RAW DATA

Vakhtang TSISKARIDZE (Tbilisi State University, Georgia) on behalf of the ATLAS Tile Calorimeter

Problem statement

The ATLAS experiment Level-1 design trigger rate is 100 kHz. This implies dropping the raw data recording due to bandwidth limitations and relying completely on the online reconstruction of the TileCal signal. On the other hand, keeping the raw data is indispensable for offline reprocessing as well as for debugging and validation purposes. The proposed approach contributes to resolving this problem.

Bandwidth limits



Block diagram of the ROD System

The bandwidth at 100 kHz for 4 drawers is 12800 bits = 400 words.

Of these, 14 words are used for:

- S-Link Begin 1 word
- Frag Header 9 words
- Trailer 3 words
- S-Link End 1 word

Thus, for each drawer we have the limit of (400 – 14) / 4 = 96.5 words.

An additional 11 words are sent for each drawer:

- Header 3 words
- DQ 5 words
- TotalET 1 word
- Muon 2 words (for 1 Muon)

Finally, for drawer data itself we have the limit of 96 – 11 = 85 words.

Proposed method

We propose a lossless data compressing algorithm that better suits existing needs. This method was checked on SPLASH events (run 87851, contains 26 SPLASH events) and proved to be sufficient to save data from ALL channels using the existing bandwidth. Note that this method needs NO INFORMATION about the pulse shape function to compress the data. So this method can be used for recording distorted or piled up signal as well.

How it works

Assuming that there is more or less correct timing for cells (which is actually the case), we can use correlation between samples of neighboring cells. We order the cells in an appropriate manner and send the differences between the samples using this order. Our experiments show also that the way of ordering the cells affects considerably the performance of the algorithm. Depending on the range of differences we are using 4 different formats described below.

Ped (4 bytes)

Δ_5	Δ_4	Δ_3	Δ_2	Δ_1	$s_6 - s_0$	s_0

7+5+4+4+4+4+4 = 32 bits

- s_0 7 bits [0..127]
- $s_6 - s_0$ 5 bits [-16..15]
- $\Delta_i = s_i - a$ 4 bits [-8..7], $a = (s_0 + s_6)$ div 2

Pedestal with bad QF means small signal with amplitude less than 8 ADC counts with bad QF.

Amp6 (6 bytes)

Δ_5	Δ_4	Δ_3	Δ_2	Δ_1	s_{max}	s_{min}

8+10+6+6+6+6+6 = 48 bits

- s_{min} 8 bits [0..255]
- s_{max} 10 bits [0..1023]
- Δ_i 6 bits [-32..31], $i = 1, \ldots, 5$

We scale previous non pedestal samples (with ANY QF) on the current samples.

Amp8 (7 bytes)

Δ_5	Δ_4	Δ_3	Δ_2	Δ_1	s_{max}	s_{min}

6+10+8+8+8+8+8 = 56 bits

- s_{min} 6 bits [0..63]
- s_{max} 10 bits [0..1023]
- Δ_i 8 bits [-128..127], $i = 1, \ldots, 5$

We scale previous non pedestal samples (with ANY QF) on the current samples.

Full (9 bytes)

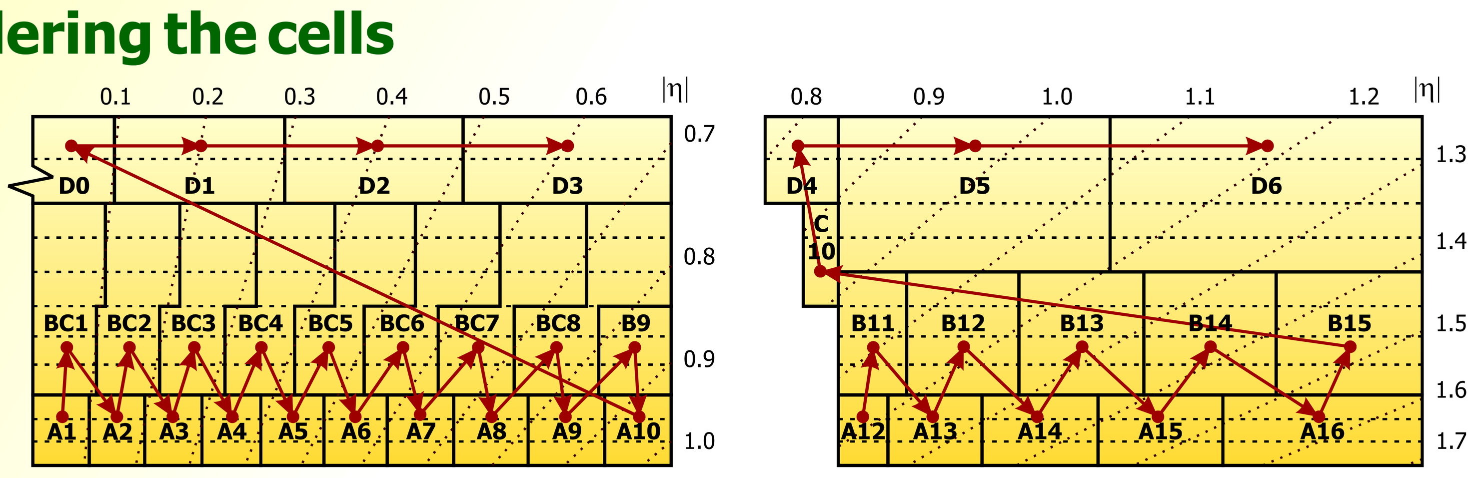
s_6	s_5	s_4	s_3	s_2	s_1	s_0

2+10+10+10+10+10+10+10 = 72 bits

- s_i 10 bits [0..1023], $i = 0, \ldots, 6$

We send all samples without compression, aligning them to byte.

Ordering the cells



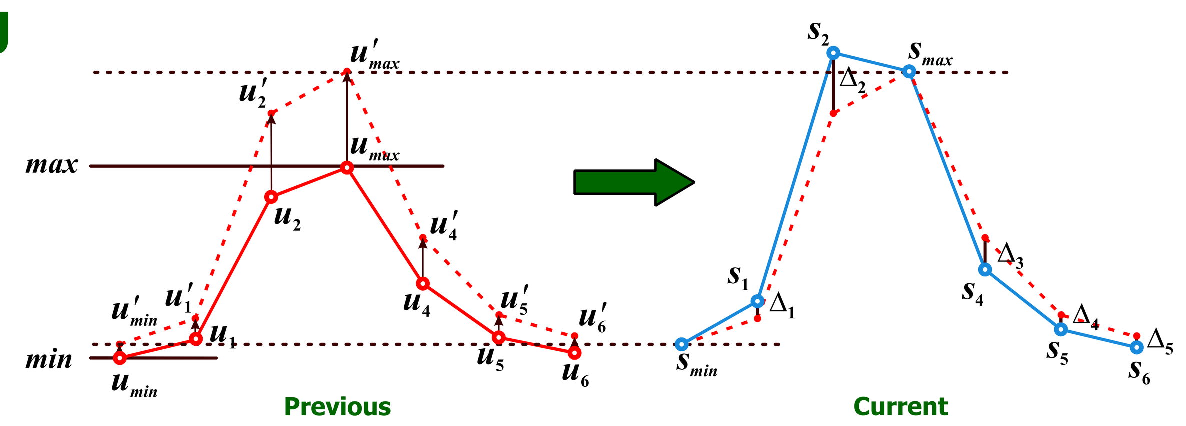
Half Barrel **Extended Barrel**

Samples scaling

$$k = \frac{s_{max} - s_{min}}{u_{max} - u_{min}}$$

$$u'_i = k \cdot (u_i - u_{min}) + s_{min}$$

$$\Delta_i = s_i - u'_i$$



Raw Data (72 bits)

Gain	1 bit	
Bad	1 bit	
Data	10 bits × 7 samples	

For transferring all channels we need:

Long Barrel: 45 channels × 9 bytes = 405 bytes ~ 101 words

Ext. Barrel: 32 channels × 9 bytes = 288 bytes ~ 72 words

Two stage processing

At present, a two-stage processing is considered.

On the first stage Amplitude, Time and Quality Factor (QF) parameters are calculated.

QF represents the sum of absolute deviations of data from the standard shape (after fitting).

The lower the value of QF, the better the quality of reconstruction.

Good Quality Factor (QF $\leq q_0$)

In the existing approach some boundary value QF = q_0 must be fixed.

It is assumed that the data with QF $\leq q_0$ is good enough to be fully processed.

In this case Optimal Filtering Reconstruction method is considered sufficient and only Amplitude, Time and QF are stored.

Reco Data (32 bits)

G	Amplitude	Time	B	QF

Gain (G)	1 bit
Amplitude	14 bits
Time	12 bits
Bad (B)	1 bit
QF	4 bits

Fragment for reconstructed Amplitude, Time and QF.

As we have at most 48 channels per drawer, the size of Reco Data is 48 words.

Thus, we have the rest 85 – 48 = 37 words for sending additional information.

Bad Quality Factor (QF > q_0)

In case of QF > q_0 the data quality is considered bad and it is necessary to store more detailed information for further studies. This is done during the second stage of data processing.

FragType1 (128 bits)

	Sample 0	G	Num	Channel ID
	Sample 2			Sample 1
	Sample 4			Sample 3
	Sample 6			Sample 5

Gain (G)	1 bit
Num	4 bits
Channel ID	8 bits
Data	10 bits × 7 samples

The above mentioned 37 available words could be used to send all data for some particular channels with bad QF. FragType1 fragment has been introduced for this purpose.

Uncompressed transfer

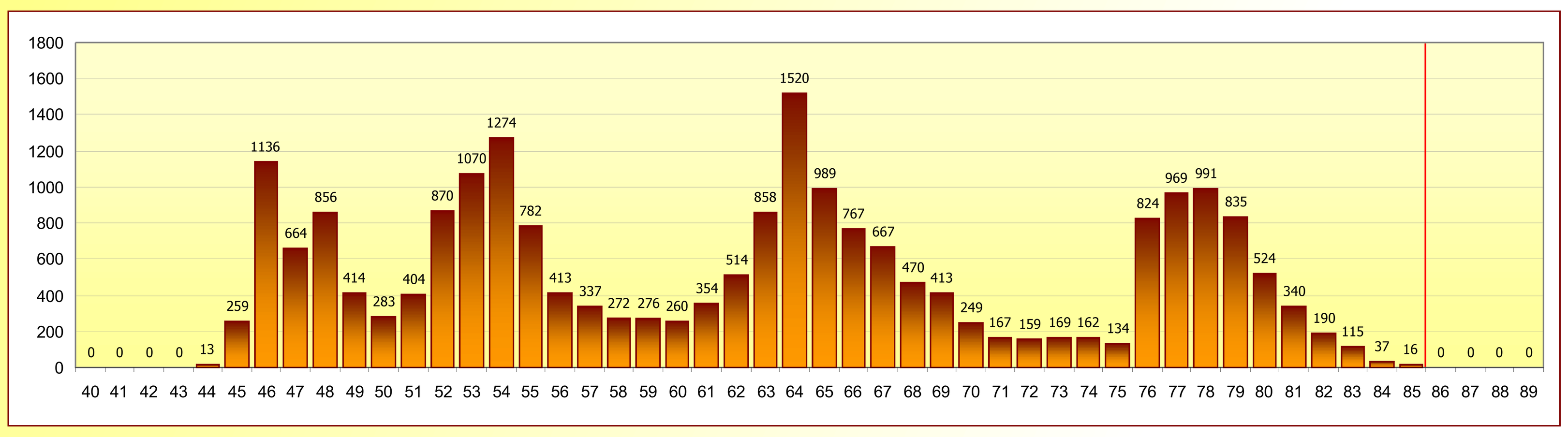
If we do not use any compression, this allows sending up to 37/4 = 9 channels of additional raw data on to the second stage of processing. This means that we have to choose quality bound q_0 large enough to avoid more than 9 channels with bad QF. Simple considerations show that when QF is bad due to the shape differences between standard pulse shape and current signal, all channels are likely to have bad QF. So, the possibility to send just 9 samples in FragType1 may sometimes appear insufficient.

Compression

Experiments show that standard compression tools such as RAR cannot successfully deal with this problem because they cannot take advantage of the smooth curved shape of the raw data and correlations between the channels. Note that if we have bad QF for some channels, we do not need to send Reco Data for them. So, we can free up a 32-bit word per channel for sending raw data.

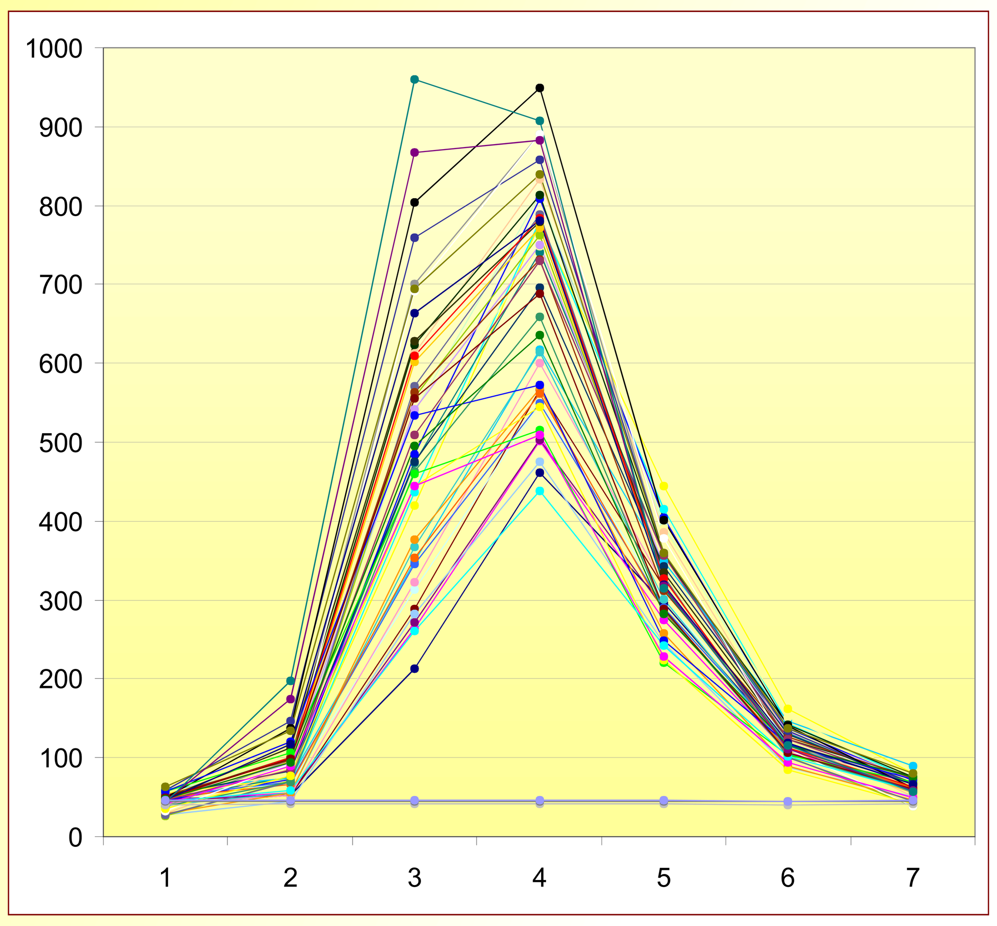
Statistics for run 87851

Histogram of compressed data lengths (fits within 85 words).



Example of successful compression of raw data with signal in ALL channels

The following example illustrates the case when QF in ALL channels is considered bad and all channels contain signal.




```
 0 :  2 => 48  53 213 462 296 116 68 | 9 | xxxxxxxxx
 1 :  5 => 46  55 264 501 275 106 68 | 7 | xxxxxxx
 2 :  4 => 56  74 419 778 444 162 80 | 6 | xxxxxx
 3 :  3 => 51  69 436 781 415 143 73 | 6 | xxxxxx
 4 :  6 => 46  53 272 503 289 106 57 | 6 | xxxxxx
 5 :  9 => 42  53 289 566 317 112 63 | 6 | xxxxxx
 6 :  8 => 48  75 473 741 357 131 66 | 7 | xxxxxxx
 7 :  7 => 47  73 485 808 405 138 71 | 6 | xxxxxx
 8 : 10 => 60  74 349 618 349 147 89 | 7 | xxxxxxx
 9 : 11 => 54  67 313 549 301 123 74 | 6 | xxxxxx
10 : 12 => 58  90 539 814 394 139 77 | 7 | xxxxxxx
11 : 13 => 56  91 501 749 390 151 75 | 6 | xxxxxx
12 : 16 => 28  44 282 476 246  88 51 | 7 | xxxxxxx
13 : 19 => 41  52 322 601 319 109 60 | 7 | xxxxxxx
14 : 18 => 50  93 542 750 342 133 73 | 9 | xxxxxxxxx
15 : 17 => 30  86 613 834 386 127 51 | 6 | xxxxxx
16 : 20 => 51  69 346 550 290 117 68 | 7 | xxxxxxx
17 : 21 => 40  55 367 614 301 110 64 | 6 | xxxxxx
18 : 22 => 26  71 557 762 331 102 38 | 7 | xxxxxxx
19 : 23 => 46  95 602 772 327 111 55 | 6 | xxxxxx
20 : 24 => 29  55 377 568 258  89 46 | 7 | xxxxxxx
21 : 25 => 50  67 354 562 285 110 62 | 6 | xxxxxx
22 : 30 => 28  69 571 789 330 100 45 | 7 | xxxxxxx
23 : 29 => 39  97 700 890 358 119 55 | 7 | xxxxxxx
24 : 28 => 52  83 475 696 342 130 68 | 7 | xxxxxxx
25 : 31 => 38  69 463 659 287 105 58 | 6 | xxxxxx
26 : 36 => 50 101 623 813 357 142 76 | 7 | xxxxxxx
27 : 35 => 49 113 628 779 335 123 76 | 6 | xxxxxx
28 : 34 => 53  97 564 732 311 126 74 | 6 | xxxxxx
29 : 37 => 47  85 510 730 356 124 68 | 7 | xxxxxxx
30 : 42 => 59 146 759 858 360 135 67 | 9 | xxxxxxxxx
31 : 41 => 49 138 804 949 401 140 62 | 7 | xxxxxxx
32 : 38 => 32  92 691 891 378 111 39 | 7 | xxxxxxx
33 : 39 => 48 100 610 784 327 112 64 | 6 | xxxxxx
34 : 46 => 57 106 460 515 220 104 70 | 7 | xxxxxxx
35 : 45 => 57 121 534 573 249 117 73 | 6 | xxxxxx
36 : 48 => 35  77 445 545 224  85 42 | 7 | xxxxxxx
37 : 47 => 41  89 444 509 229  94 50 | 6 | xxxxxx
38 :  1 => 44  59 261 439 243 102 58 | 7 | xxxxxxx
39 : 14 => 49  98 555 688 288 107 60 | 9 | xxxxxxxxx
40 : 15 => 48  94 495 636 283 116 75 | 6 | xxxxxx
41 : 26 => 47 117 664 781 320 119 67 | 7 | xxxxxxx
42 : 27 => 64 134 694 839 359 138 80 | 6 | xxxxxx
43 : 40 => 44 174 867 882 318 111 58 | 9 | xxxxxxxxx
44 : 43 => 43 198 960 907 315 115 57 | 7 | xxxxxxx
45 : 32 => 40  41  41  41  41  40 41 | 2 | xx
46 : 33 => 44  44  45  44  44  45 45 | 2 | xx
47 : 44 => 46  46  46  47  47  45 46 | 4 | xxxx
```

Time fitting

While the problem of fitting the bandwidth seems to be solvable, at the moment it remains an open problem whether the algorithm can fit the time restrictions of data processing at 100 kHz frequency. The answer to this question will affect the QF bound value q_0, which determines whether to use second stage algorithms or not. Obviously, the lower the value of q_0, the higher the required performance of the compression algorithm.

Further improvements

Note that the compression algorithm does not use any explicit information about the pulse shape function and one could get additional benefit by taking this shape into consideration. It also seems promising to use a tree driven ordering of cells instead of a linear one. There are also some other improvements that can increase the compression.

Conclusions

- A data compression algorithm is proposed for the Raw Data recording (in case of bad QF).
- The method used is especially suited to the existing detector design and hardware configuration.
- The algorithm has been tested for heavy input flow and proved to be capable to fit into the existing bandwidth bounds.
- The determination of the optimal value for Quality Factor bound q_0 determining switching to the second stage algorithms remains open and is under study.

Computing in High Energy and Nuclear Physics
Prague | Czech Republic | 21 - 27 March 2009

vakhtang.tsiskaridze@cern.ch