

CRU-FLP DATA FORMAT

RAW DATA HEADER requirements:

- 1) GBT word (80 bit, TPC uses 112 bit, but it can't generate header from the FEE)
- 2) DDL word (32 bit)
- 3) One header every block of data sent from the FEE
- 4) The header should be in common between the 2 data transmission protocols, this is why a 64 bit word size has been selected. If not convenient in the future can be adjusted.
- 5) The data header should contain all the information needed to describe the payload:
 - a) Size
 - b) Payload identifier
 - c) Source identifier
 - d) Time stamps
 - e) status/error bit
 - f) Trigger information
- 6) A trailer can be added:
 - a) Final size
 - b) Extra time stamp
 - c) status/error bits
- 7) The header must be created in the FEE or in the CRU, before shipping the data in the memory of the FLP

CONTENT OF THE RAW DATA HEADER

Previous header used during RUN2 (known as CDH V3)

0	Block length ⁵ [0-31]				
1	Format version ⁵ [24-31]	MBZ [22-23]	L1 trigger message ¹ [14-21]	MBZ [12-13]	Event ID 1 (bunch crossing) ² [0-11]
2	PAR requests ⁵ [24-31]	Event ID 2 (orbit number) ² [0-23]			
3	Block attributes ⁵ [24-31]	Participating sub-detectors ² [0-23]			
4	Status & error bits ⁵ [12-31]				Mini-event ID (bunch crossing) ³ [0-11]
5	Trigger classes low ² [0-31]				
6	MBZ [18-31]		Trigger classes middle-low ² [0-17]		
7	Trigger classes middle-high ² [0-31]				
8	ROI low ⁴ [28-31]	MBZ [18-27]		Trigger classes high ² [0-17]	
9	ROI high ⁴ [0-31]				

Figure 1. Format of Common Data Header.

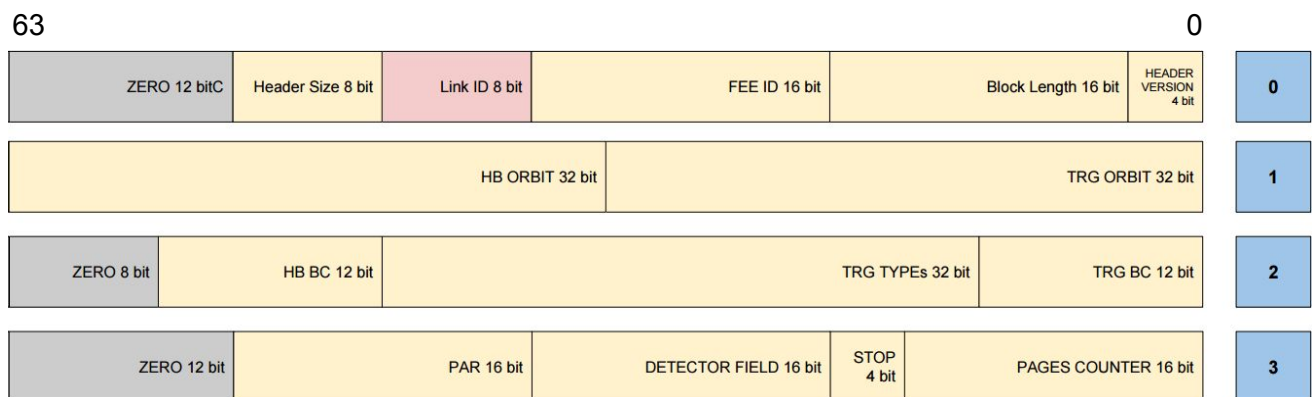
What do we want to keep?

- 1) Block length (32 bit)
- 2) Header version (4 bit)
- 3) TRG information (BC ORBIT) (60 bit ... not final)
- 4) Detector specific field (16 bit?)
- 5) PAR (16 bit?)

What do we want to add?

- 1) GBT link ID
- 2) FEE ID
- 3) HB ID
- 4) STOP
- 5) Pages counter
- 6) HEADER size in word

(Position of the fields not final)



	Reserved for future use
	Must be filled by det FEE (can be filled by CRU if FEE can't fill it properly)
	Must be filled by CRU

4 WORDS 64 bit.

That means

- 1) $4 * 128 \text{ bit for CRU (every 64 bit will be translated to 128 bit)} = 2 * 256 \text{ bit}$

The DMA page is 8 KB that means an overhead of 2 words (256 bit) out of 256 (0.78%)

FIELDS description

- **Block Length [16 bit]** : the length of the page (can be 0xFFFF if detector can't fill it properly).
- **FEE ID [16 bit]** : unique ID assigned to the FEE to know from which section of the detector data is coming (at least 18 bit).
- **The link ID [8 bit]** : used to identify from which GBT link in the CRU data is coming (0 - 23).
- **Header version [4 bit]** : version number to identify the header when new fields are added
- **Header Size [8 bit]** : number of 64 bit words composing the header
- **BC [12 bit] and Orbit [32 bit]** : the Trigger and HB identification.
- **TRG type [32 bit]** : trigger type set by CTP
- **Detector field [16 bit]** : detector specific field used by detector.
- **PAR [16 bit]** : field used by detector to trigger a configuration of the FEE

These 2 fields are needed only if the MEMORY DATA MODEL 1 is selected.

- **Pages counter [16 bit]** : counter to keep track of the different pages belonging to the same trigger.
- **STOP bit [4 bit]** : 1 bit to identify the last page (if there are more pages of 8KB belonging to the same trigger).

In usual run condition where the data of a detector fits in 8KB

- STOP bit always 0x1
- Pages Counter always 0x0000

The pages counter can be used to detect erroneous condition if it is not contiguous.

The link ID should be filled by the CRU (to be discussed)

The FEE ID should be filled by the FEE, if it is not possible the CRU should fill it.

The linkID is used to assign the data to the different SUPERPAGE.

MEMORY DATA MODELS

In the following sections, different data models will be presented.

Data model 1 sacrifice memory layout optimization, to have a simple firmware algorithm.

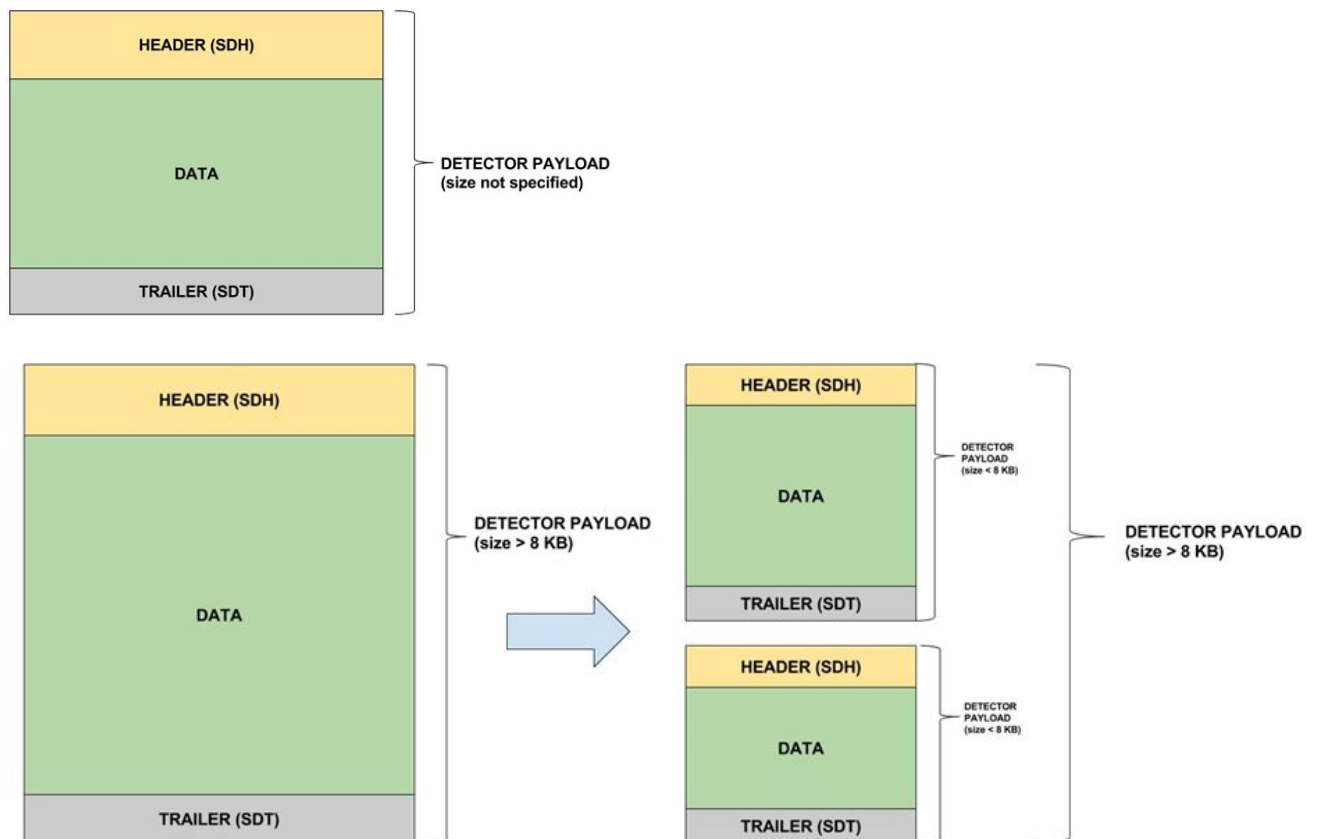
Data model 2 optimize the data format in memory with more requirements on the firmware.

Both model are compatible with the formatted data presented so far.

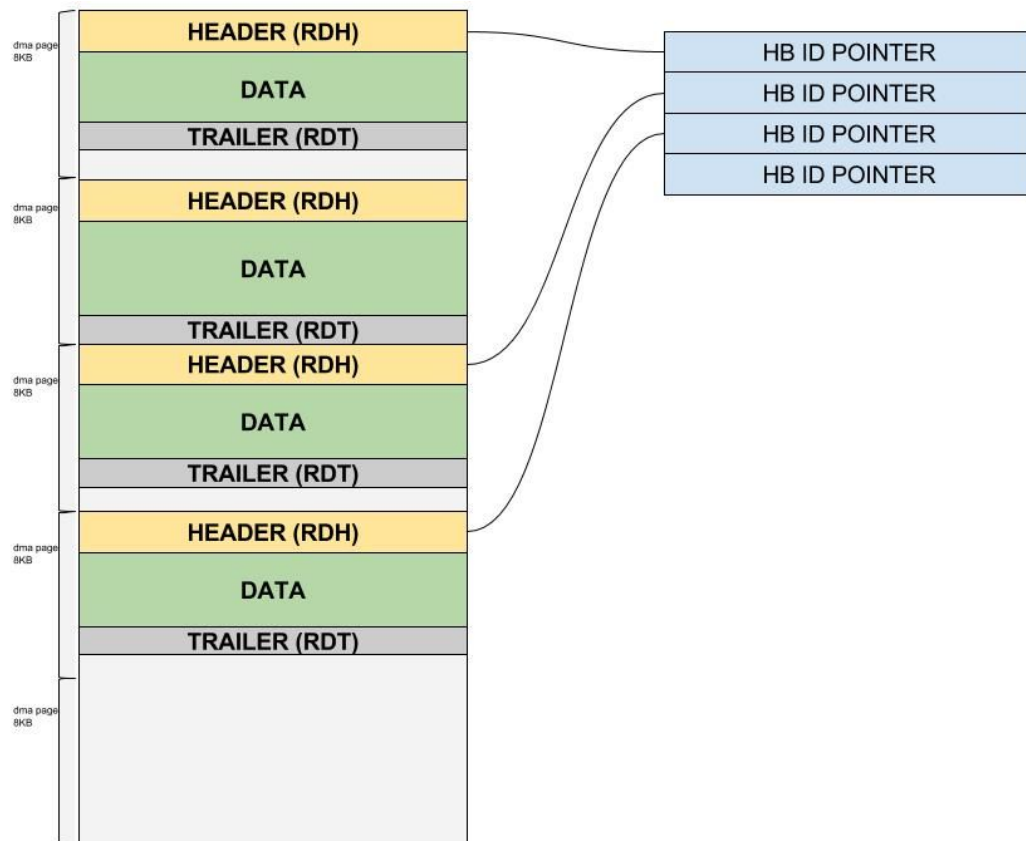
The detector formatted data presented here

<https://indico.cern.ch/event/633689/contributions/2571921/attachments/1450518/2237138/017.04.27.O2.WP1.FLPdataFormat.pdf>

Can be applied to both data models:

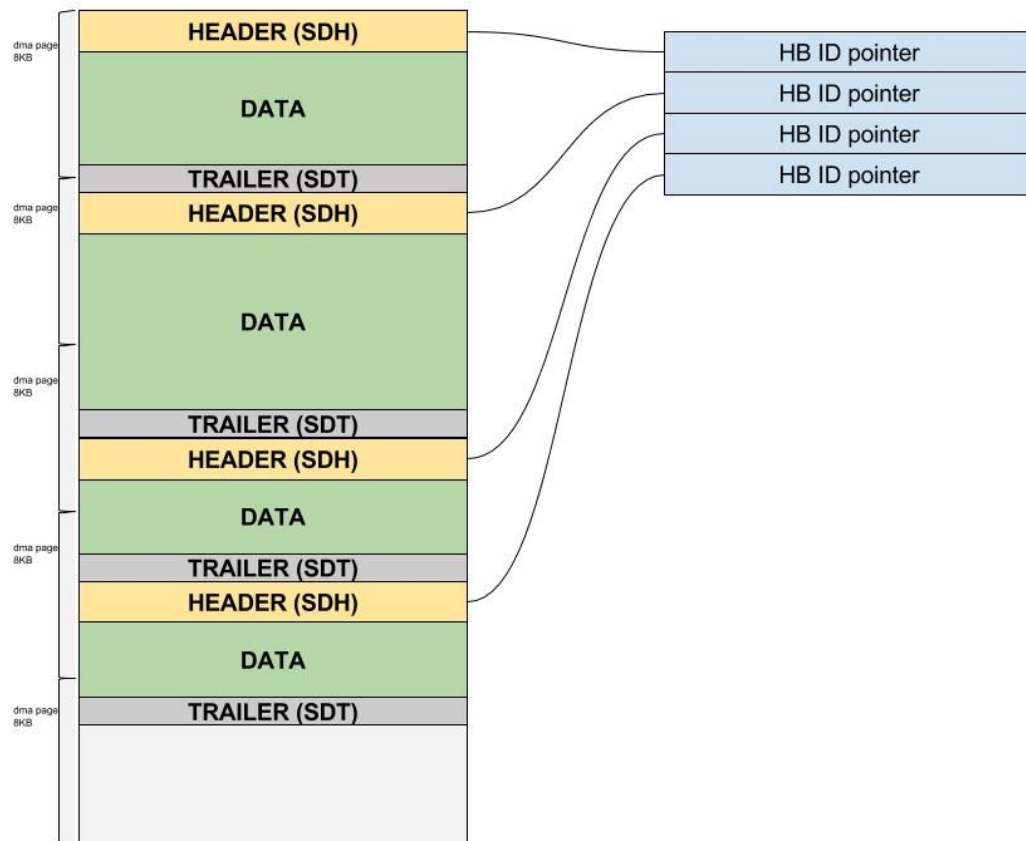


MEMORY DATA MODEL 1 (ACCEPTED)



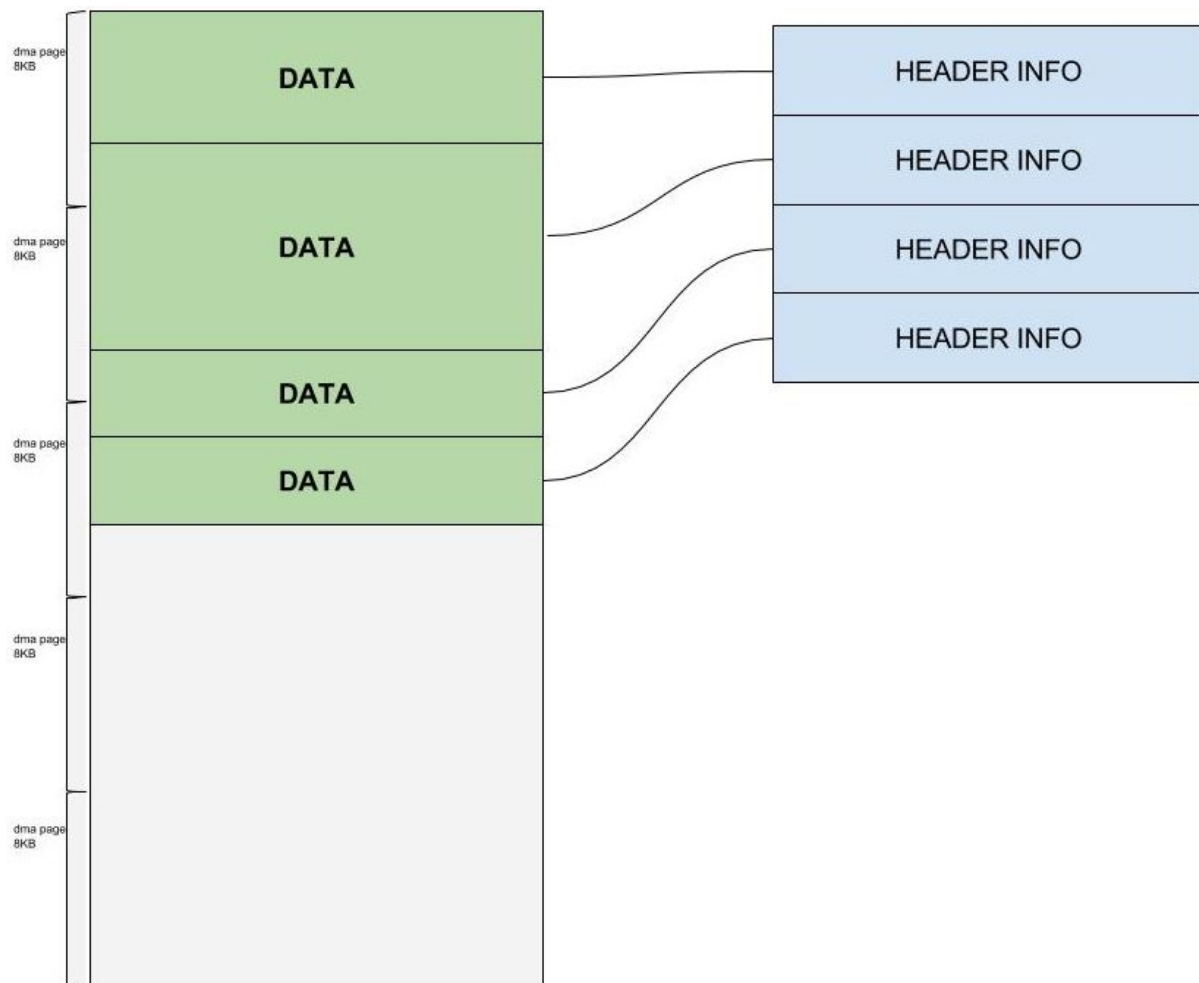
- 1) The SUPERPAGE is allocated by the SW and the start address + size is delivered to the CRU.
- 2) The CRU fills the SUPERPAGE with DMA pages of 8KB (HEADER + DATA)
- 3) The DMA PAGE format (HEADER + DATA) can be sent by the FEE ... in this case the CRU forward the information as it is
- 4) If the FEE has data for more than 8KB, it should generate 2 DMA pages with similar header.
 - a) 13 KB of data should be splitted in 8 + 5 KB ... but also 6 + 7 KB would be accepted.
 - b) Looking at the header of each DMA page it is possible for the sw to reconstruct all the information and to put together the different pieces.
- 5) The blue table in the picture should be created by the software and it contains the pointer in the memory of the beginning of each HB ID

MEMORY DATA MODEL 2 (under discussion)



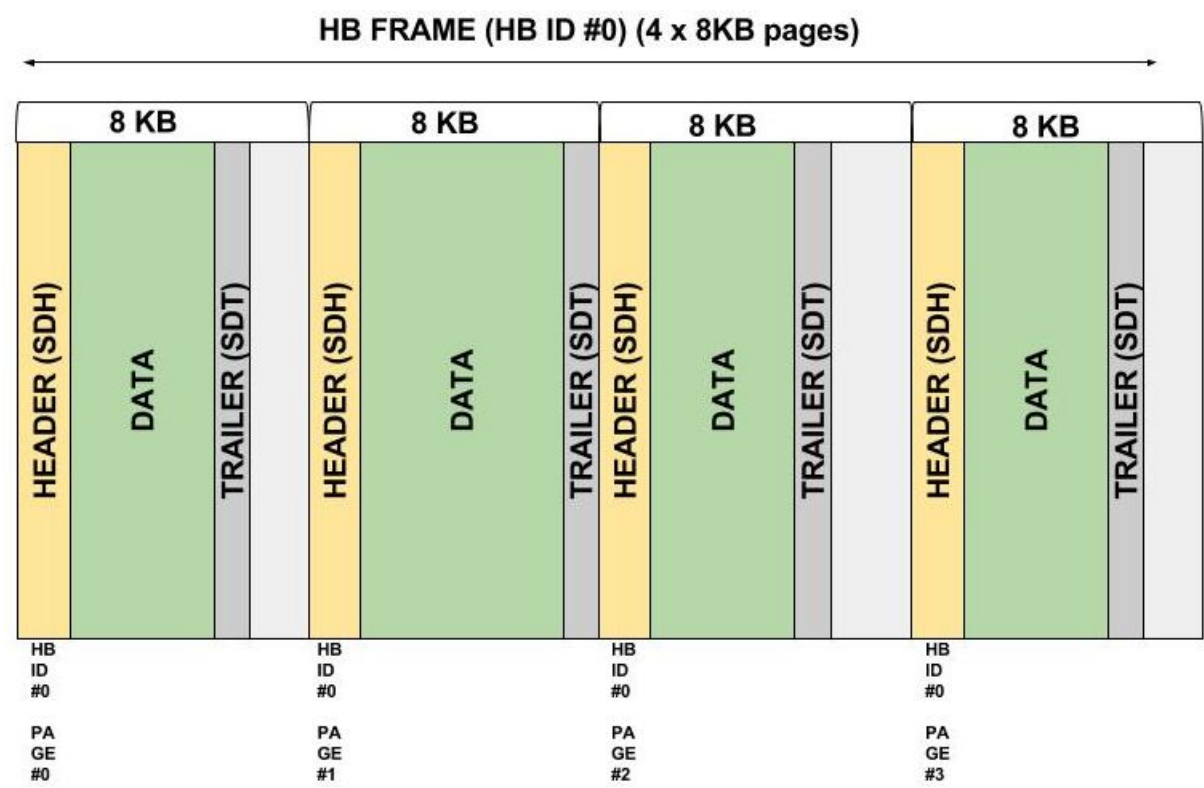
- 1) The SUPERPAGE is allocated by the SW and the start address + size is delivered to the CRU.
- 2) The CRU fills the SUPERPAGE with data coming from the detector and store the transfer information in a separate table
- 3) The max block size is still 8 KB, but data could cross 2 different DMA pages (as showed in the picture, the payload is between 2 different dma pages)
- 4) All the data in the superpage is contiguous and there are no holes in the data stream.
- 5) The blue table in the picture should be created by the software and it contains the pointer in the memory of the beginning of each HB ID

MEMORY DATA MODEL 3



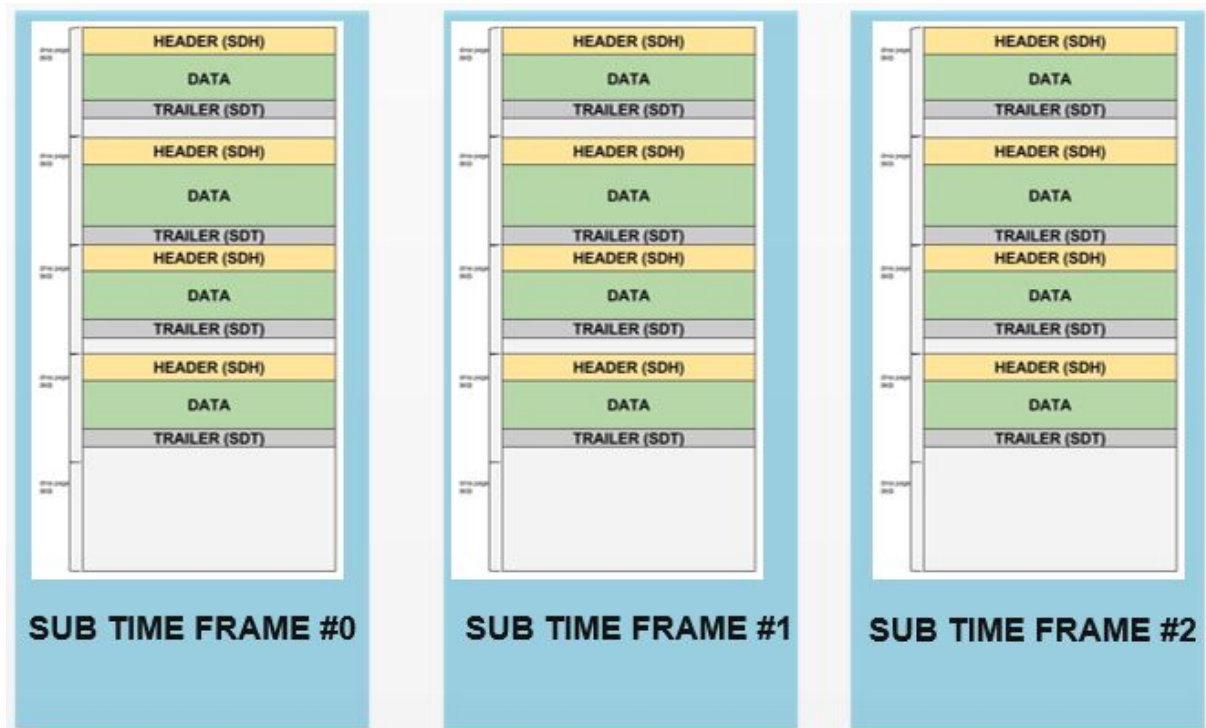
- 1) The SUPERPAGE is allocated by the SW and the start address + size is delivered to the CRU.
- 2) The CRU receives the data, remove the HEADER from the DATA (and stores it in a separate table), shift the DATA words at the beginning of the DMA PAGE and append the rest of the data to fill up the 8KB. This process is repeated until the DMA page size is reached and ready to be sent to the DMA engine
- 3) The CRU fills the SUPERPAGE with data coming from the detector and store the transfer+header information in a separate table
- 4) The superpage is a continuous block of DATA

HB FRAME



One HB frame consists of several 8KB DMA pages

SUB TIME FRAME

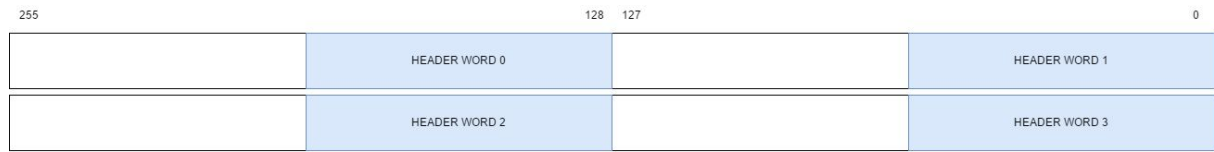


Superpage will be changed every new **SUB TIME FRAME**

WORDS IN MEMORY

In the CRU every GBT word is moved to 128 bit.

GBT word 80/112 bit + extra 16 bit



The DDL FEE will push several 32 bit words to match the same header position in the memory (to be discussed)

31

0

HEADER WORD # 0
HEADER WORD # 0
ZERO 0x00000000
ZERO 0x00000000
HEADER WORD # 1
HEADER WORD # 1
ZERO 0x00000000
ZERO 0x00000000
HEADER WORD # 2
HEADER WORD # 2
ZERO 0x00000000
ZERO 0x00000000
HEADER WORD # 3
HEADER WORD # 3
ZERO 0x00000000
ZERO 0x00000000

Questions:

- The field should be multiple of bytes? Keep it multiple of bytes or int
- What is the maximum number of words we should not cross (1% of the 8KB?)
- The DDL detectors and the GBT detectors must have the same type of header or some freedom is allowed? (different header size/field) ... the footprint should be the same
- Can we keep only TRIGGER ID instead HB ID?