



EDMS NO.  
**2954073**

REV.  
**1.1**

VALIDITY  
**DRAFT**

REFERENCE

**CMS-CE-ES-0413**

Date: 2023-11-24

## Working document on Specification

# **HGCROC3b**

### ABSTRACT:

This document is a working document to detail the specification for the HGCROC3b.

- Description not differing from the HGCROC2 is in blue.
- New description for HGCROC3 is in black
- Think unclear or in process of change are in red

DOCUMENT PREPARED BY:	DOCUMENT TO BE REVIEWED BY:	DOCUMENT TO BE APPROVED BY:
C. De La Taille F. Dulucq D. Thienpont S. Extier M. El Berni	P. Aspell D. Barney P. Bloch P. Dauncey A. David C. De La Taille J. Hirschauer A. Lobanov M. Mannelli J. Manns M. Noy	P. Aspell P. Bloch C. De La Taille K. Gill J. Virdee

DOCUMENT SENT FOR INFORMATION TO:

**This document is uncontrolled when printed. Check the EDMS to verify that this is the correct version before use.**



REFERENCE EDMS NO. REV.

VALIDITY

**CMS-CE-ES-0413****2954073 1.1 DRAFT**

Page 2 of 144

Available to the CMS HGCal Community for internal use only. Not for distribution outside of the community.



REFERENCE EDMS NO. REV.

VALIDITY

**CMS-CE-ES-0413****2954073 1.1 DRAFT**

Page 3 of 144

## HISTORY OF CHANGES

REV. NO.	DATE	PAGES	DESCRIPTIONS OF THE CHANGES
1.0	2023-09-13	145	Working document with HGCROC3b updates
1.1	2023-11-24	145	Updates on align buffer behaviour



## TABLE OF CONTENTS

<b>1. HGCROC3: architectural overview</b>	<b>6</b>
1.1. Analog front-end	7
1.1.1. Preamplifier	8
1.1.2. Input DAC (leakage current compensation)	13
1.1.3. Shapers	16
<b>1.1.4. Discriminators</b>	<b>22</b>
<b>1.2. Mixed-signal blocks</b>	<b>26</b>
1.2.1. 10bits ADC and Align Buffer	26
1.2.2. TOT and TOA TDCs	28
1.3. Digital blocks	30
1.3.1. Data path	31
1.3.1.1. TOT compression	32
1.3.1.2. Data Path Content	32
1.3.1.3. Frame Description	34
1.3.2. Trigger path	36
1.3.2.1. Disabling channel	36
1.3.2.2. Trigger path content	37
1.3.2.3. Charge Linearization Block	37
1.3.2.4. ADC Charge Processing	37
1.3.2.5. TOT Charge Processing & Selection of ADC value or TOT.	38
1.3.3. Trigger Cell Sums	39
1.3.3.1. Frame description	40
1.3.4. Digital parameters	42
1.3.5. Description of the operating modes of the chip	43
1.3.5.1. Start-up sequence	43
1.3.5.2. RAM1 to RAM2 operation	45
1.3.5.3. RAM2 to SerDes State-Machine	45
1.3.5.4. Definition of the Time Tag Counters	46
1.3.5.5. Definition of the Resets	47
1.4. Interfaces blocks	48
<b>1.4.1. Fast command</b>	<b>48</b>
<b>1.4.2. I2C</b>	<b>53</b>
1.4.3. Output E-links	55
1.5. Ancillary blocks	56
1.5.1. PLL and clocks distribution	56



1.5.2. Bandgap and voltage references	58
<b>1.5.3. Calibration circuit</b>	<b>58</b>
1.5.3.1. Strobe pulse length	60
1.5.4. Monitoring	60
<b>2. Packaging, I/Os and powering scheme</b>	<b>62</b>
<b>    2.1. I/Os list</b>	<b>65</b>
2.2. Pin List	66
3. ASIC parameters	<b>70</b>
3.1. I2C Addressing	72
<b>4. Measurements of v2 for inputs for HGCROCv3 specifications</b>	<b>142</b>
4.1. Power consumption: RUN mode vs. SLEEP mode	142
4.2. Power consumption: HD vs. LD	142
<b>5. Production testing and in-situ calibration</b>	<b>143</b>
5.1. Production testing	143
5.2. In-situ testing	143

## 1. HGCROC3: architectural overview

The general block diagram for HGCROC3 is shown in Figure 1.

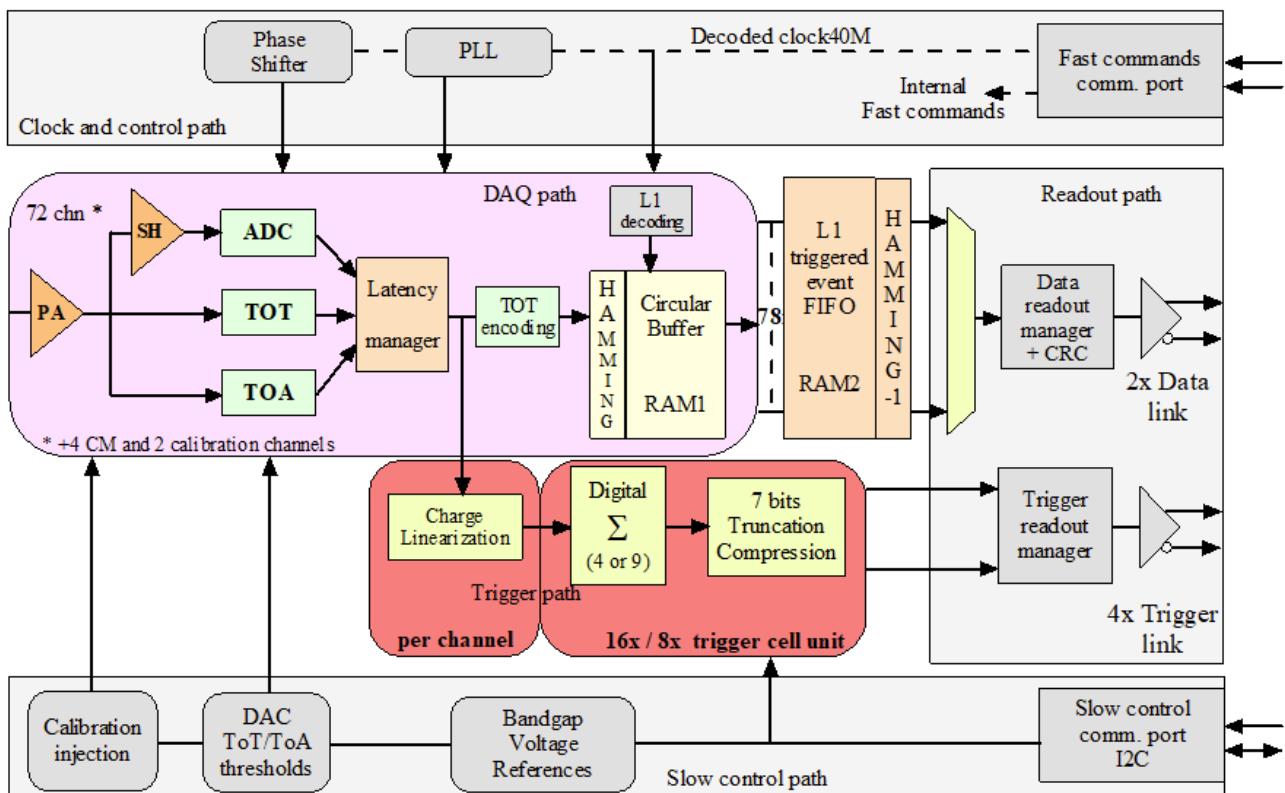


Figure 1: The block diagram for the HGCROC3 ASIC

Most of the blocks and functionality of V2 remains unchanged for V3. The main changes are regarding the digital processing of the DAQ path like the L1 triggered event FIFO, the hamming coding and encoding. For simplicity, the Circular Buffer in the figure above will be called RAM1 and the L1 triggered event FIFO will be called RAM2. Every Control and Command's Modules will be triplicated with TMRG tool. Data encoded inside RAM1 and RAM2 with Hamming. [The chip handles 72 channels + 4 channels for common mode subtraction + 2 special calibration channels.](#)

The figure below shows how the charge measurement is provided: in the preamplifier's linear region, the charge is given by the ADC, typically up to 100 MIP. When the preamplifier saturates, the Time-Over-Threshold (TOT) is used to give the charge. The figure below shows how ADC and TOT data combine to give the charge information.

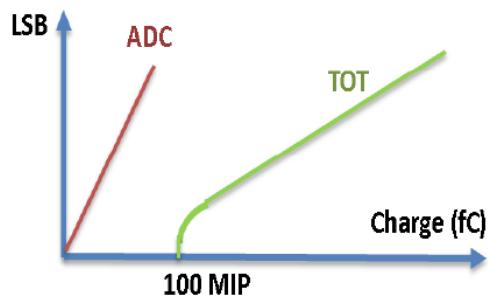


Figure 2: Graphical representation of the individual ADC and TOT charge components.

As the LSB are not equal for “ADC range” and “TOT range”, a linearization step is needed in the trigger path: in the non-linear region of the TOT, a plateau value is applied as shown in the figure below.

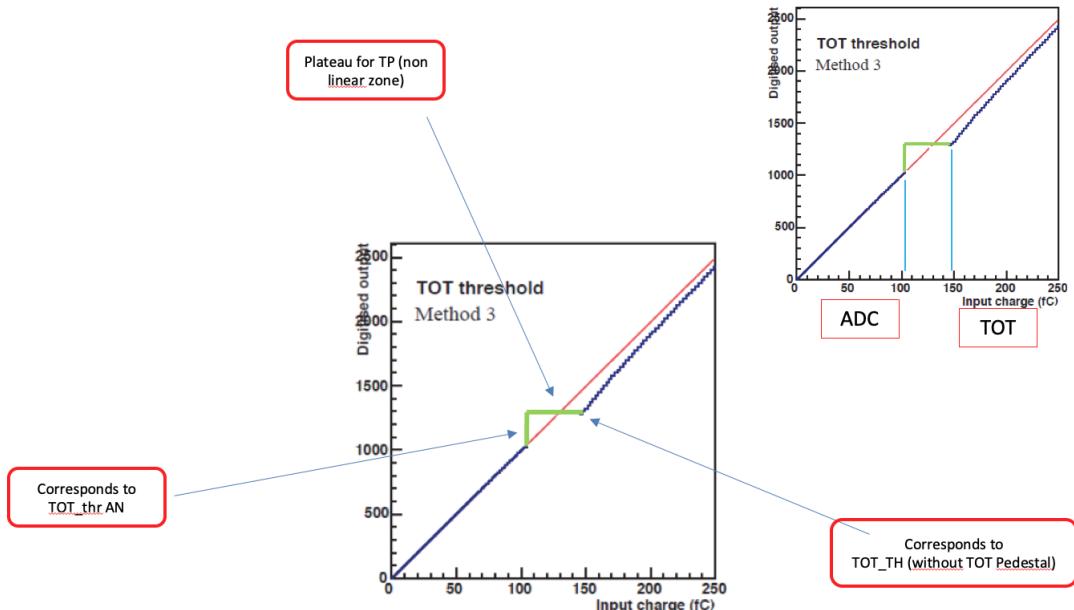


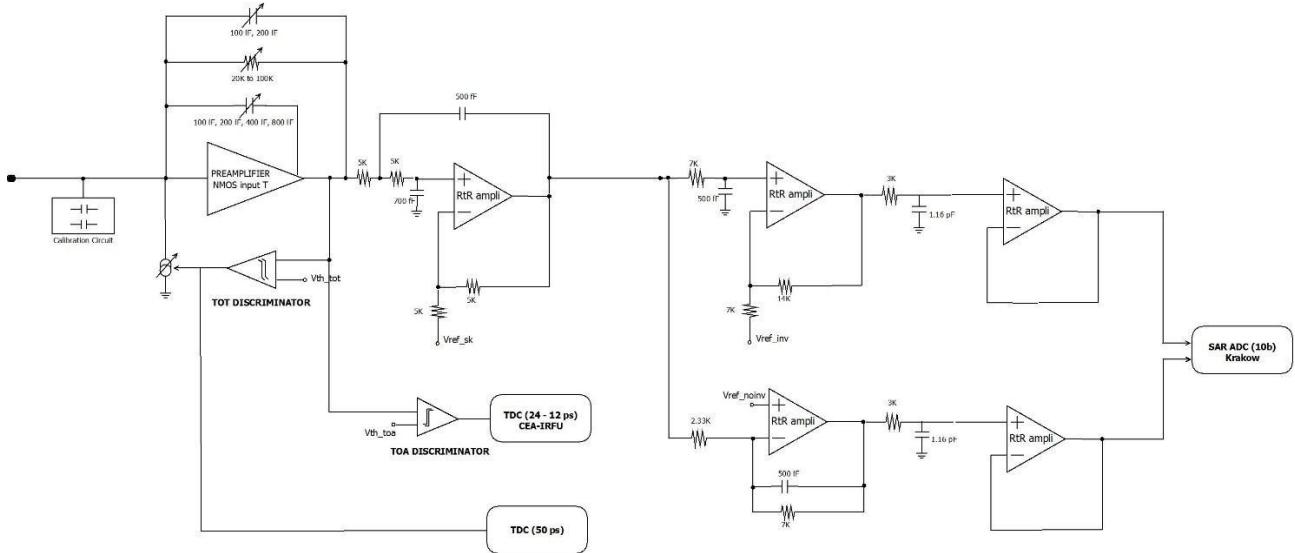
Figure 3: Graphical representation of how the ADC and TOT charge components combine in the TP.

### 1.1. Analog front-end

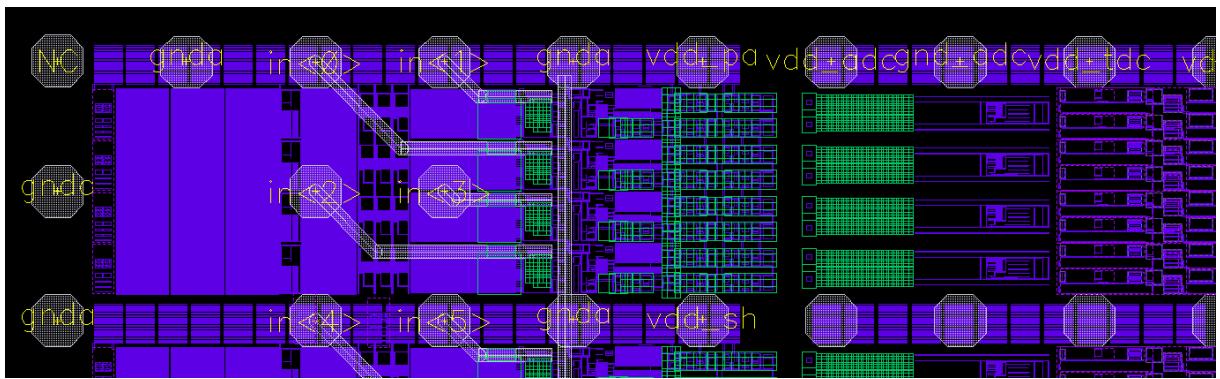
The front-end may be divided in three main sub-parts:

- The preamplifier which converts the input charge coming from the silicon diode to an output voltage. It must provide the first amplification of the signal with the best noise performance. In the linear part of the amplifier, the feedback capacitors and feedback resistors provide the gain and the shape of the output signal which is send to the shaper. From the saturation and above, the feedback discriminator triggers and provides the charge measurement by using the “Time Over Threshold” technique (TOT). Another discriminator allows to give the timing information. The preamplifier can be calibrated by injecting a voltage step through two channel-wise selectable capacitors (0.5pF and 8pF).
- The shaper part is composed of three stages: a Sallen-Key filter, a RC<sup>2</sup> filter and a unity gain amplifier to drive the ADC. The shaping time can be adjusted over +/- 20% to compensate for process variations and ensure out of time pileup below 20%.

- The two discriminators providing the TOT and TOA (Time of Arrival) pulses, each one sent to a dedicated TDC.



In order to accommodate the C4 bump bonding pattern, the layout was done in order to fit in 120  $\mu\text{m}$  height and avoid sensitive analog electronics below the bumps. Four channels fit between two rows of pads and the slow control, common to 4 channels, is placed below the pads.



In addition to the 72 readout channels, there are 4 channels for common mode subtraction and 2 channels for MIP calibration. The common mode channels are similar to the regular channels except they stop at the ADC (there are not TDCs). They do not enter the trigger path but are read out in the data path.

The calibration channels can be used in characterization mode or behave as regular ones, set by slow control.

In the following sections, more details are given for each block.

### 1.1.1. Preamplifier

The preamplifier is DC coupled to the input and provides two outputs:

- Outpa connected to the **shaper** and the **TOT discriminator**. Its DC operating point is the same as the preamplifier input (160 - 200 mV).
- outCf\_pa connected by default to the **TOA discriminator**. Its DC operating point is around outpa + Vgs ( $\sim 500$  mV).

A 8b-DAC is connected to the preamplifier input to absorb the leakage current coming from the sensor. More on this below.

The purpose of the preamplifier is to convert the input charge to a voltage output with the best signal-to-noise ratio and a gain adapted to the MIP signal. It must also provide a “short” signal duration to mitigate the Out-of-Time pileup effect at the shaper output. To meet all these requirements, the gain and the time constant must be adjustable (**these parameters are global, not channel-wise**). In the table below, all the possibilities are described:

Rf ( $\Omega$ )	25K, 50K, 66.66K, 100K	In parallel, these resistors provide 15 values to be adjusted with the Cf and Cf_comp values to get a decay time constant around 10 ns.
Cf (fF)	50, 100, 200, 400	Combined with the Cf_comp capacitors, provide the gain of the preamplifier.
Cf_comp (fF)	100, 200	Same purpose as Cf capacitors, but connected differently to improve the preamplifier stability. From gain point-of-view can be considered in parallel with Cf capacitors.

The following plot shows the preamplifier response to a 10 fC input charge for different choice of gain. The feedback resistor is adjusted so that the Rf\*Cf product is constant and so the “duration” of the signal. As can be seen in Fig. 4, an undershoot appears for the highest preamp gain.

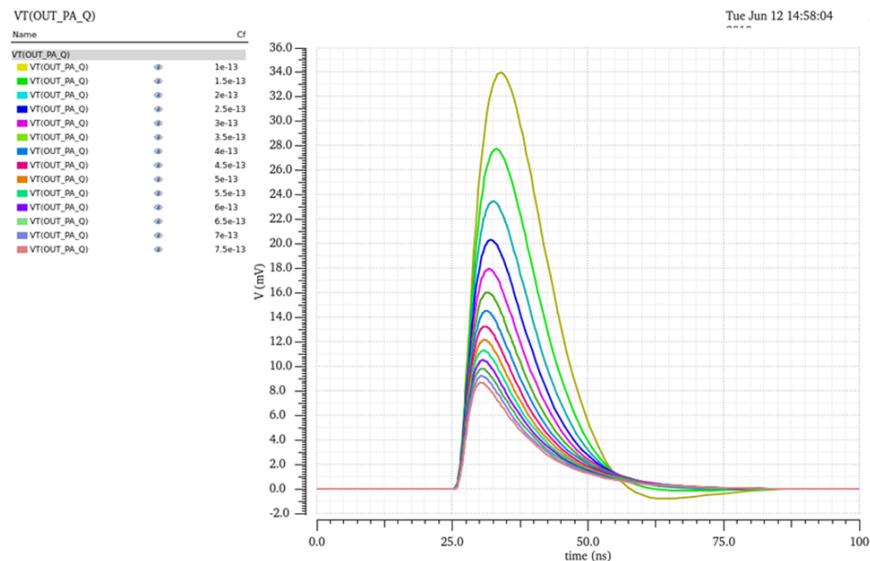


Figure 4 Preamplifier response to a 10 fC input charge for different choice of gain.

Changes in HGCROC3:

Remove unused timing part to increase the speed and the phase margin

1. Remove the common source transistor and resistor
2. Remove the current source to save 200  $\mu$ A
3. The decoupling capacitors are used to filter vbi\_pa (already in V2)
4. The level shifter for negative pulses is kept but not used

Minimum transistor size is M16 7/0.13 (negative polarity switch).



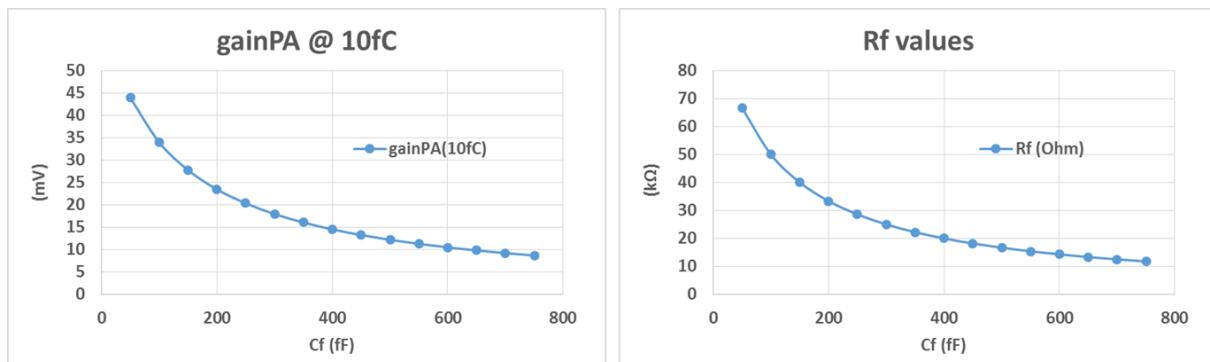
In the table below, all the parameters concerning the preamplifier are described.

### Preamplifier parameters

Name	# bits	I2C Sub-block / sub-address	Description
<b>Inputdacpol</b>	1	Channel-wise	Leakage current Input-DAC polarity
<b>Inputdac&lt;8:0&gt;</b>	5	Channel-wise	Leakage current Input-DAC Value (0 $\mu$ A default; 50 $\mu$ A max.)
<b>Probe_pa</b>	1	Channel-wise	Preamplifier output probe
<b>LowRange</b>	1	Channel-wise	0.5pF injection cap
<b>HighRange</b>	1	Channel-wise	8pF injection cap
<b>Channel_off</b>	1	Channel-wise	"1" = preamplifier input tied to ground
<b>Trim_vbi_pa&lt;5:0&gt;</b>	6	Global-analog	6b-DAC for preamp input stage current tuning
<b>Trim_vbo_pa&lt;5:0&gt;</b>	6	Global-analog	6b-DAC for preamp output stage current tuning
<b>ON_pa</b>	1	Global-analog	"1" = enable preamplifier bias
<b>Cf&lt;3:0&gt;</b>	4	Global-analog	Preamp feedback cap. <b>&lt;0&gt; = 50fF, &lt;1&gt; = 100fF, &lt;2&gt; = 200fF, &lt;3&gt; = 400fF</b> In //
<b>Cf_comp&lt;1:0&gt;</b>	2	Global-analog	Preamp feedback comp. cap. <b>&lt;0&gt; = 100fF, &lt;1&gt; = 200fF</b> In //
<b>Rf&lt;3:0&gt;</b>	4	Global-analog	Preamp feedback Res. <b>&lt;3&gt; = 25K, &lt;2&gt; = 50K, &lt;1&gt; = 66.66K, &lt;0&gt; = 100K</b> In //
<b>Neg</b>	1	Global-analog	"1" = negative input polarity (Default)
<b>Calib_dac&lt;11:0&gt;</b>	12	Voltage references	Calibration DAC value

<b>IntCtest</b>	1	Voltage references	Selection of the Calibration DAC
<b>ExtCtest</b>	1	Voltage references	Selection of the external pulse test

The two following plots show respectively the preamp gain as a function of the feedback capacitor (with feedback R adjusted so that  $R \cdot C$  is constant) and the nominal feedback resistor value for a given Cf value.



As the MIP value is depending on the sensor thickness and the irradiation, the preamplifier gain must be programmable in order to adjust the ADC range to 100 MIPs. The table below summarises the gain specifications.

thickness	MIP		Noise		LSB	ADC range	ADC range	MIP	Noise
	e-	fC	e-	fC	fC	fC	MIP	ch above ped	LSB
120um	9000	1,4	2000	0,3	0,4	320	222	3,6	0,8
120um	9000	1,4	2000	0,3	0,2	160	111	7,2	1,6
120 irrad (worse 3000 fb-1)	6000	1,0	2600	0,4	0,1	80	83	9,6	4,2
200um	15000	2,4	2500	0,4	0,4	320	133	6	1,0
200 um	15000	2,4	2500	0,4	0,4	320	133	6	1
200 irrad (worse)	6000	1,0	3000	0,5	0,1	80	83	9,6	4,8
300 um	20000	3,2	2000	0,3	0,4	320	100	8	0,8
300 irrad (worse 3000fb-1)	10000	1,6	2200	0,4	0,2	160	100	8	1,8
300 irrad (worse 3000 fb-1) high gain	10000	1,6	2200	0,4	0,1	80	50	16	3,5

The table below gives the nominal Cf and Rf values for a given ADC range.

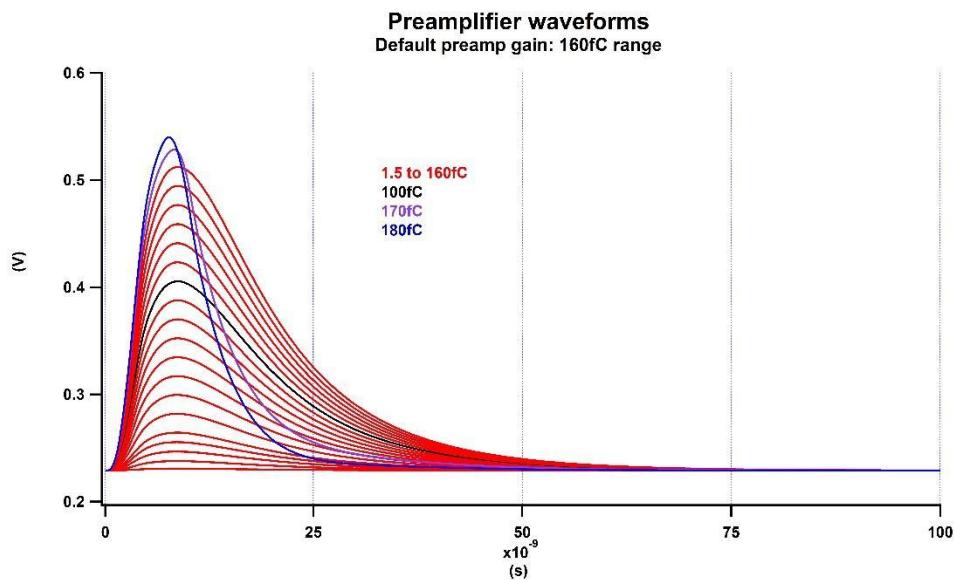
ADC range	Cf	Cf_comp	Rf	Comment
80 fc	200 fF “0100”	0 “00”	50K “0100”	Typical value! Can be more or less adjusted in practice!
160 fc	200 fF “0100”	200 fF “10”	33.33K “0101”	Default slow control values
320 fc	400 fF “1000”	300 fF “11”	16.66K “1100”	Typical value! Can be more or less adjusted in practice!

The preamp gain settings showed above are typical to get the specified ADC range, but since the feedback capacitors can be set from 50 fF to 750 fF and more and the feedback resistor from 11.7K to

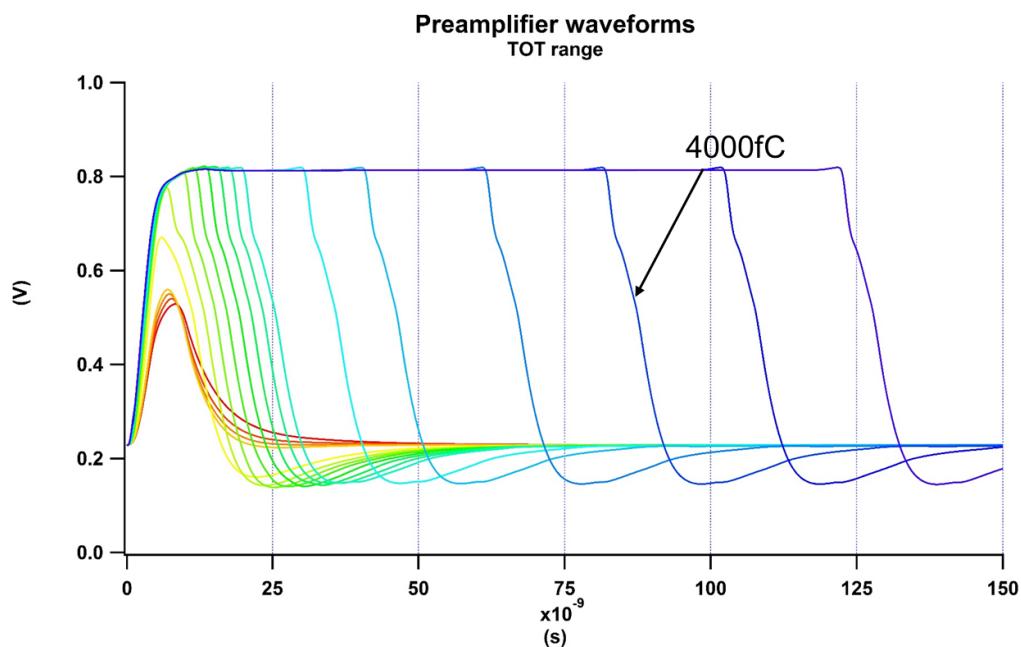
100K, there are more possible combinations reaching the specifications. **The Annexe B describes all the combinations of R<sub>f</sub> and C<sub>f</sub> (TBD, for now cf V2 datasheet).**

Since the preamplifier converts an input charge to an output voltage, its behaviour over the entire charge dynamic must be well known and characterized. That can be divided in three steps:

- The linear mode: the preamplifier provides an output amplitude proportional to the input charge. It is able to provide linear amplitude over ~300mV dynamic range<sup>1</sup> (see red curves in the following plot). The ADC is used to measure the charge in this region which is named ADC range.
- The non-linear mode: this mode occurs in-between the linear and the saturated modes when the preamplifier is no longer linear but not still fully saturated. It is in this region that the TOT threshold has to be set in order to optimize the ADC range linearity. The preamplifier non-linear mode leads to the non-linearity of the TOT in the beginning of the TOT range, but the pile-up limitation is expected to be the best in this region (see violet and blue curves in the following plot and red curves in the next). The non-linear mode occurs for an output amplitude between 500 and 600mV.
- The saturated mode: it occurs for an output amplitude above 600mV (see yellow and greens curves in following plot). In this region, the preamplifier pulse width is proportional to the input charge and so well suited to use the Time-over-Threshold technics. The drawback is the undershoot of the preamplifier signal leading to incorrect charge measurement in the next bunch crossing. The undershoot is due to the fact that in saturation the preamplifier is in open loop and consequently slower to recover its normal behaviour.



<sup>1</sup> The linear dynamic range is limited to 300mV because the first stage of the preamplifier is composed of four serial transistors needed to get a large open loop gain (90dB) what implies the first stage output amplitude is limited by a cascode transistor's  $V_{DS} - V_{DS\_SAT}$  voltage.

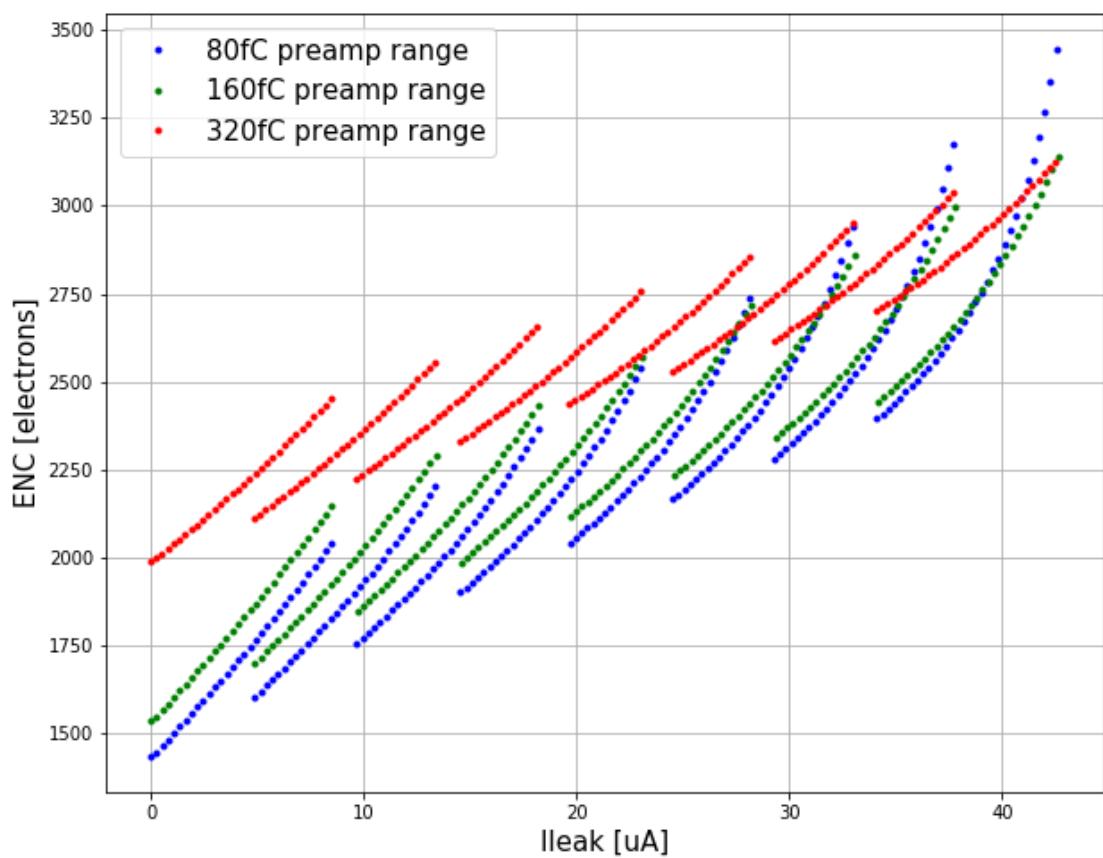
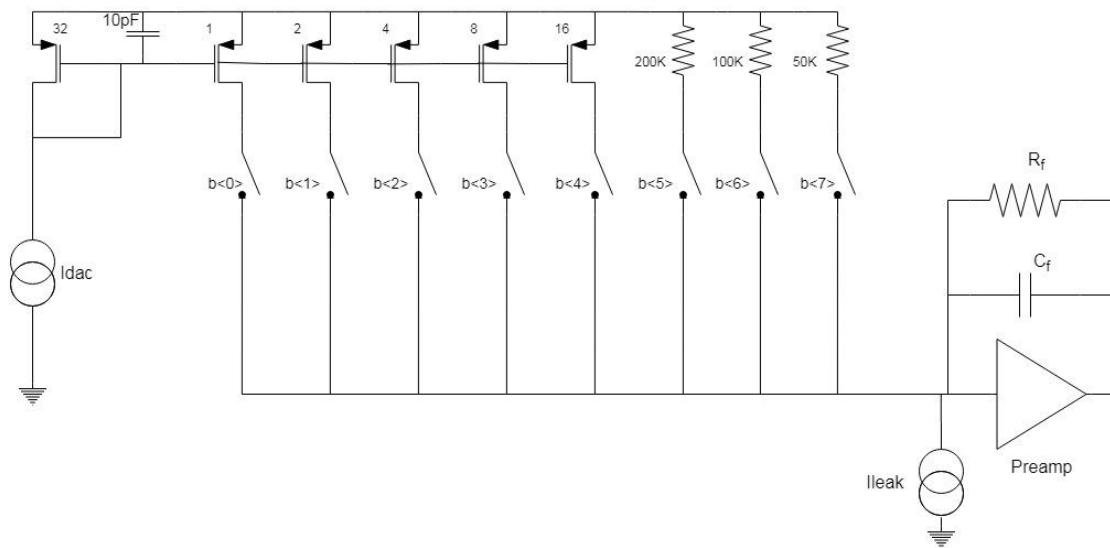


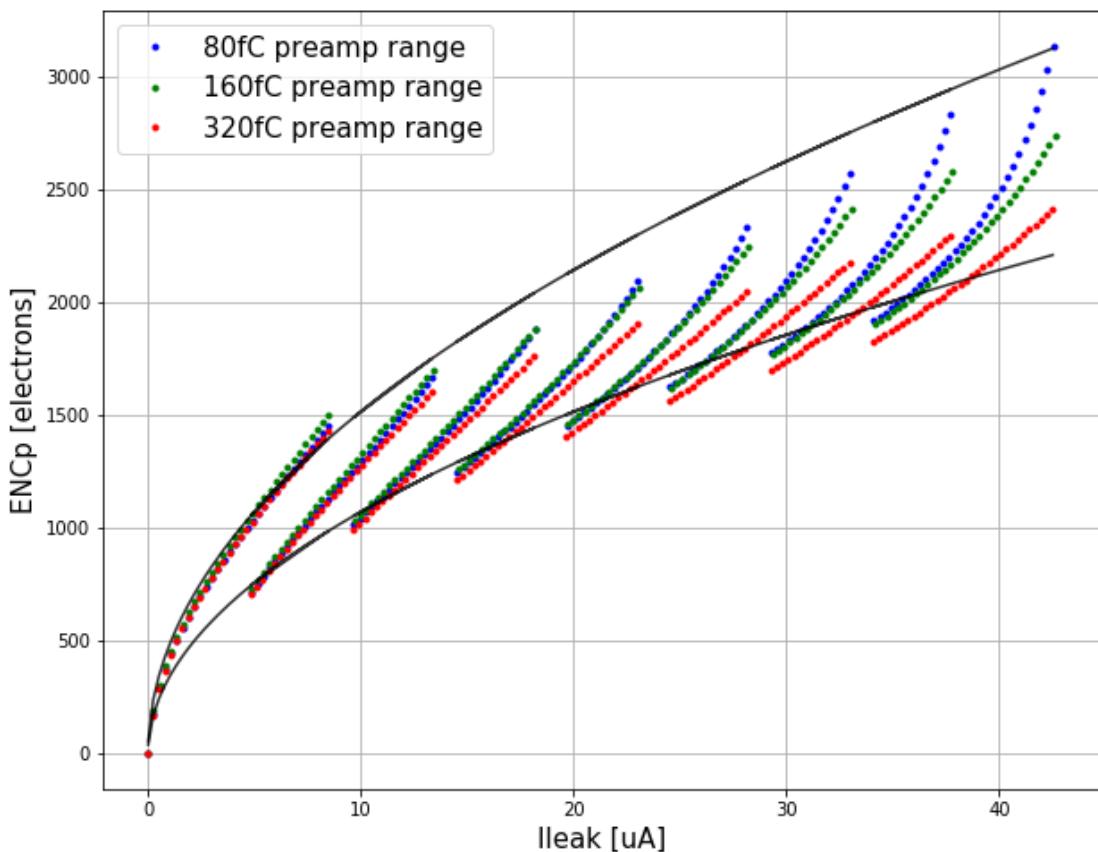
### 1.1.2. Input DAC (leakage current compensation)

The parallel noise can be reduced by  $\sqrt{2}$  by filtering the current mirror. A 10 pF is taken from vbi\_pa to vb\_inputdac to filter the noise. The maximum leakage compensation is increased from 10 uA to 50 uA. The DAC is raised from 5 bits to 8 bits by adding a low noise DAC for 5, 10 and 20 uA made with resistors.

- 1- Replace all 60/0.13 CMOS switches by 3/0.13 NMOS to reduce leakage
- 2- Replace 50/0.5um current mirror for leakage current inversion by 5/0.5 to reduce capacitance on input
- 3- Removes 3/0.15 NMOS cascodes
- 4- Add 3 bits connected to respectively 50k, 100k and 200k to vdd\_dac
- 5- The dac polarity is now common to all channels

The figure of the new architecture is shown below. The main changes are the 10pF to filter the noise from the master current source, and the 3 resistors to achieve better noise performance at higher leakage current as shown in simulations below.



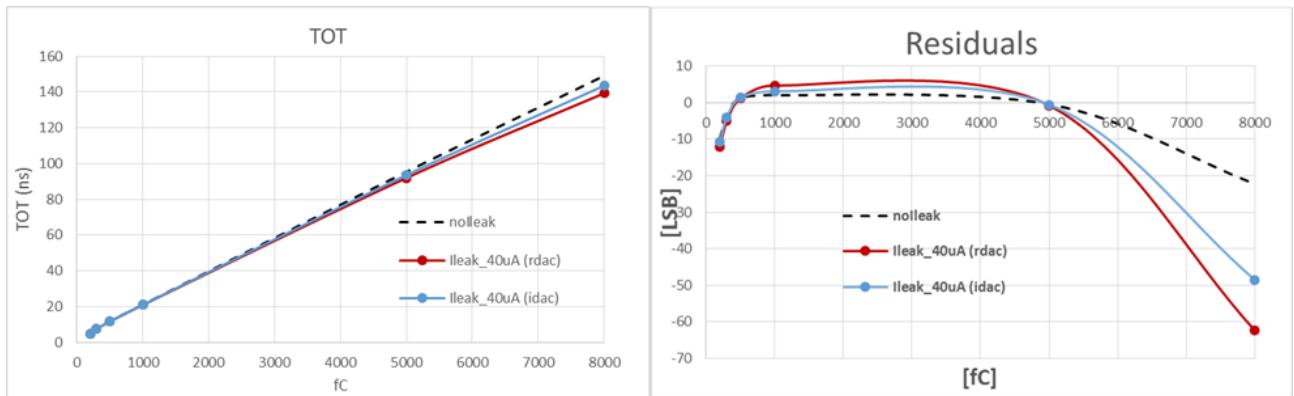


The leakage noise in HGCROC-v2 is higher than the simulations above: by filtering the input DAC with the 10pF, it follows the N=2 curve (upper black curve). By adding the R-based 3-bits DAC, the leakage noise does not follow the N=2 shape over the full leakage current range but is lowered in-between N=2 and N=1 shapes.

When a large input signal occurs the preamplifier saturates and moves to an open-loop circuit: the channel enters in the TOT mode. In this situation, the preamplifier is no longer a low-impedance circuit and therefore the input voltage is decreasing by  $I/C_{\text{total}}$ , with  $I$  the input current due to the charge and  $C_{\text{total}}$ , the total capacitance at the preamplifier input. The TOT gain is tuned by a constant current discharging the preamplifier feedback capacitor.

Now the input DAC used to compensate for some leakage current needs to provide relatively constant current whatever the preamplifier input voltage, but for the R-DAC, an additional current equal to 200mV/R will increase the TOT current leading to a faster TOT (the maximum value of this additional current can reach 20% of the TOT current). This effect is expected to be less pronounced for the i-DAC as the transistors are in their strong saturation region.

Briefly, as shown in the previous and following plots, the R-DAC achieves better noise optimization ( $\sim 1000$  electrons at high leakage) but degrades a little bit at high charge a linearity which must be calibrated in any case.



### 1.1.3. Shapers

The shaper is divided in three stages:

- A Sallen-Key (S-K) shaper, gain 2
- A  $RC^2$  shaper, gain 2
- Then a buffer to drive the ADC

It is a 4<sup>th</sup> order RC shaper with the peaking time typically set around 23ns (shaping time  $\sim$  5ns). The purpose is to optimise the signal-to-noise ratio and use the full available dynamic range ( $\sim$  1 V). The signal must be short enough to keep the signal below 20% after 25ns (for the next bunch crossing, to limit the out of time pileup). As the gain and the decay time are given by the preamplifier feedback, the shaper has to ensure the optimal shaping time, between 20 and 25 ns. The shaping time is adjustable over 2 bits to mitigate process variations.

The user has to set the inv\_vref and noinv\_vref 10b-DACs to globally set the DC levels of respectively the inverter and non\_inverter shapers, and so set the ADC pedestal. These 10b-DACs have typically 1mV LSB. In order to reduce the dispersion per channel, the user can play with a channel-wise trimming 6b-DAC: these 6b-DAC have typically 2 mV LSB.

- The inverter shaper output's DC level is equal to  $3 * (\text{inv\_vref} < 9:0> - \text{trim\_dac} < 4:0>) - 2 * V_{\text{inpa}}$
- The non\_inverter shaper output's DC level is equal to  $2 * V_{\text{inpa}} - \text{noinv\_vref} < 9:0>$
- With  $V_{\text{inpa}}$  the preamplifier input's DC level ( $\sim$  200 mV).
- The ADC converts the differential voltage ( $S_H_{\text{noinv}} - S_H_{\text{inv}}$ ).

#### Optimisation of the « ADC range »

We have introduced a simple circuit to automatically find out the best combination. The principle is to force one of the ADC input to 0.6V, perform a scan of the Vref DAC of the other branch and set it to the value giving the code 256 (266 in fact to have some margin), and then redo the same operation but for the other branch. By construction, this way will optimise the pedestal as well as the dynamic range.

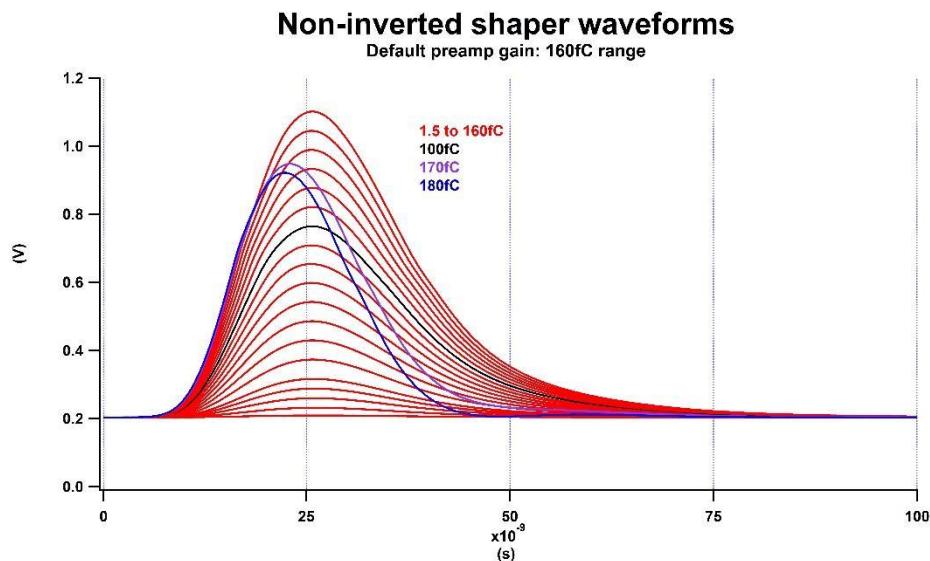
The channel-wise pedestal trimming DAC will be raised from 5b to 6b to better improve the channel-to-channel uniformity.

## Shaper parameters

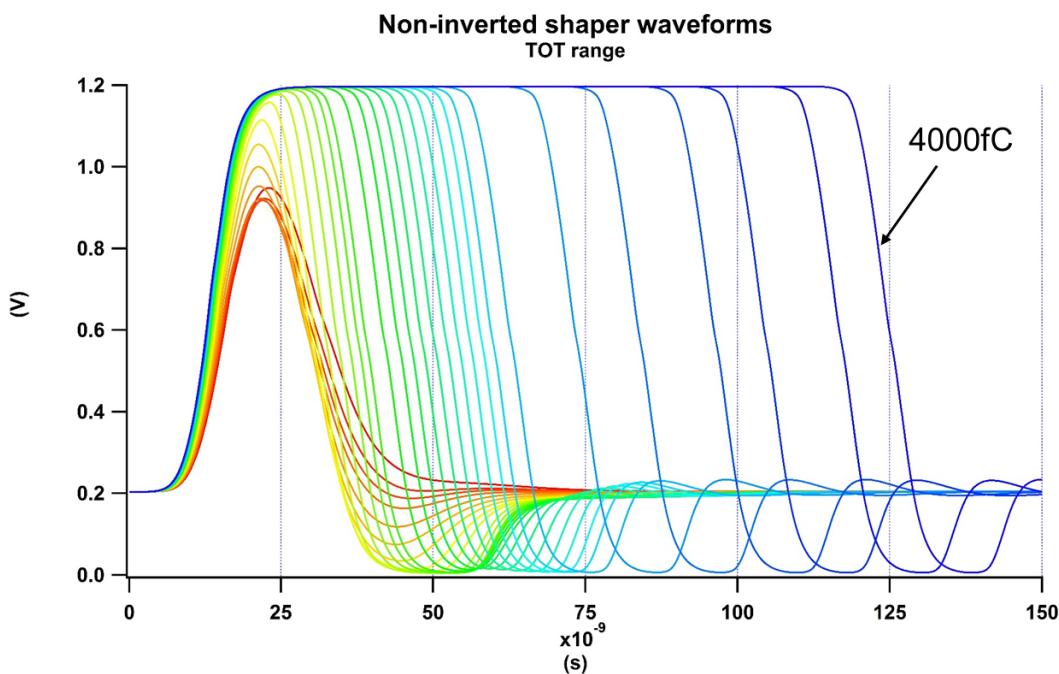


Name	# bits	I2C Sub-block / sub-address	Description
Probe_noinv	1	Channel-wise	Non inverter shaper output probe ("1" = selected)
Probe_inv	1	Channel-wise	Inverter shaper output probe ("1" = selected)
trim_dac_inv<5:0>	6	Channel-wise	Local 6b-TrimDAC for ADC pedestal tuning
ON_rtr	1	Global-analog	"1" = enable shaper amplifiers bias
Ibi_sk<1:0>	2	Global-analog	S-K amplifier input stage current
Ibo_sk<5:0>	6	Global-analog	S-K amplifier output stage current
S_sk<2:0>	3	Global-analog	S-K amp Miller cap. <0> = 50fF, <1> = 100fF, <2> = 200fF
Ibi_inv<1:0>	2	Global-analog	Inverter amplifier input stage current
Ibo_inv<5:0>	6	Global-analog	Inverter amplifier output stage current
S_inv<2:0>	3	Global-analog	Inverter amp Miller cap. <0> = 50fF, <1> = 100fF, <2> = 200fF
Ibi_noinv<1:0>	2	Global-analog	Non Inverter amplifier input stage current
Ibo_noinv<5:0>	6	Global-analog	Non Inverter amplifier output stage current
S_noinv<2:0>	3	Global-analog	Non Inverter amp Miller cap. <0> = 50fF, <1> = 100fF, <2> = 200fF
Ibi_inv_buf<1:0>	2	Global-analog	Inverter buffer input stage current
Ibo_inv_buf<5:0>	6	Global-analog	Inverter buffer output stage current
S_inv_buf<2:0>	3	Global-analog	Inverter buffer Miller cap. <0> = 100fF, <1> = 200fF, <2> = 400fF
Ibi_noinv_buf<1:0>	2	Global-analog	Non Inverter buffer input stage current
Ibo_noinv_buf<5:0>	6	Global-analog	Non Inverter buffer output stage current
S_noinv_buf<2:0>	3	Global-analog	Non Inverter buffer Miller cap. <0> = 100fF, <1> = 200fF, <2> = 400fF
Rc<1:0>	2	Global-analog	Shaping time adjustment
Inv_vref<9:0>	10	Voltage references	Inverter shaper global reference
Noinv_vref<9:0>	10	Voltage references	Non Inverter shaper global reference

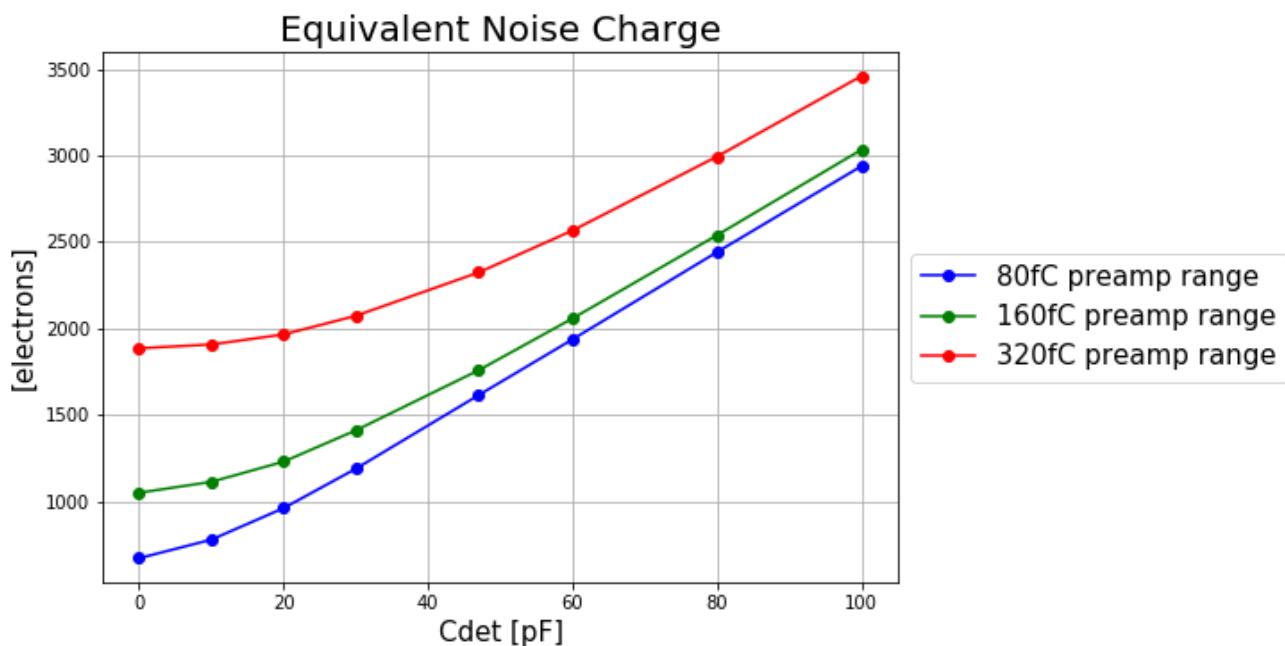
The following plot shows the non-inverted shaper output. The red curves correspond to the signal in the ADC range. The violet and blue curves show the response to small "TOT charges". It can be noticed the duration of the signals. Whether it remains 20% of signal in BX+1 in the ADC range (preamplifier linear mode), in the preamplifier non-linear mode, it remains less than 5%.



The following plot shows the non-inverted shaper output in TOT range. This TOT range is using the non-linear mode (red curves) and the saturated mode of the preamplifier (yellow and green curves). It can be noticed the undershoot in the saturated region limiting the pile-up efficiency.



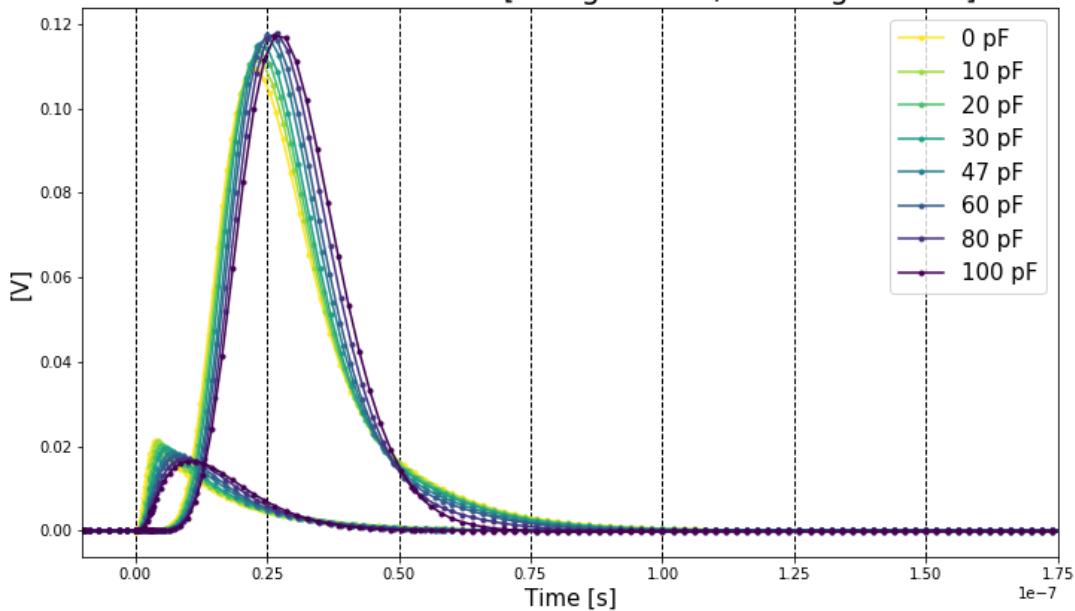
The next figure presents the Equivalent Noise Charge (ENC) as a function of the detector capacitance. It may be noticed the lower performance in the low gain mode where the contributions of the parallel noise and of the second stage are more visible.



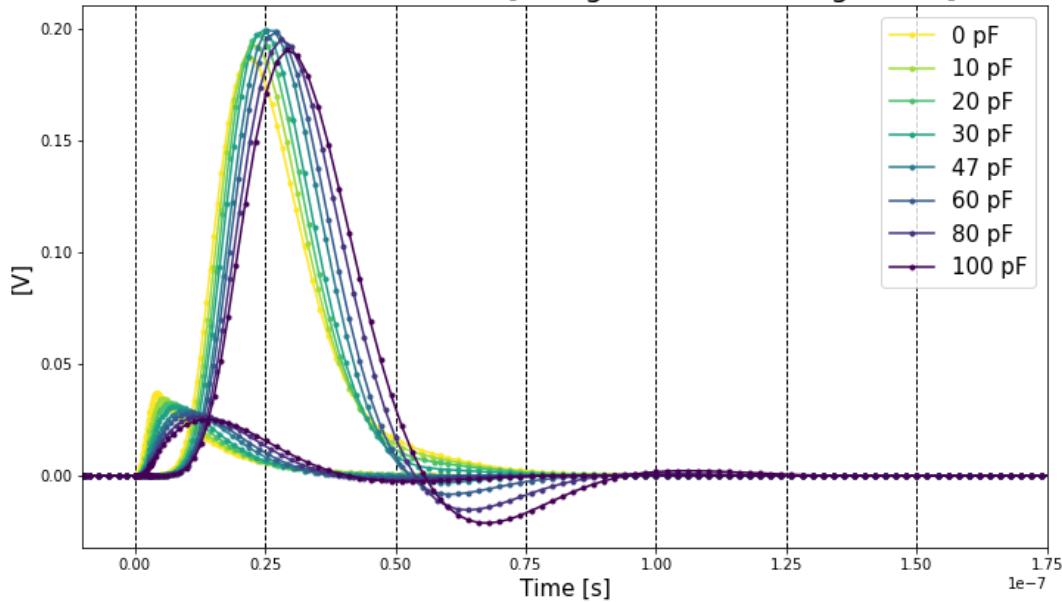
The gain, the peaking time, the OOT pileup vary with corners (PVT variations) and with the sensor capacitance. Therefore some parameters are available to tune these values: the shaping time over 2 bits, the bandwidth of the RtR amplifier.

The waveforms below show the preamplifier and the shaper outputs as a function of the sensor capacitance Cdet.

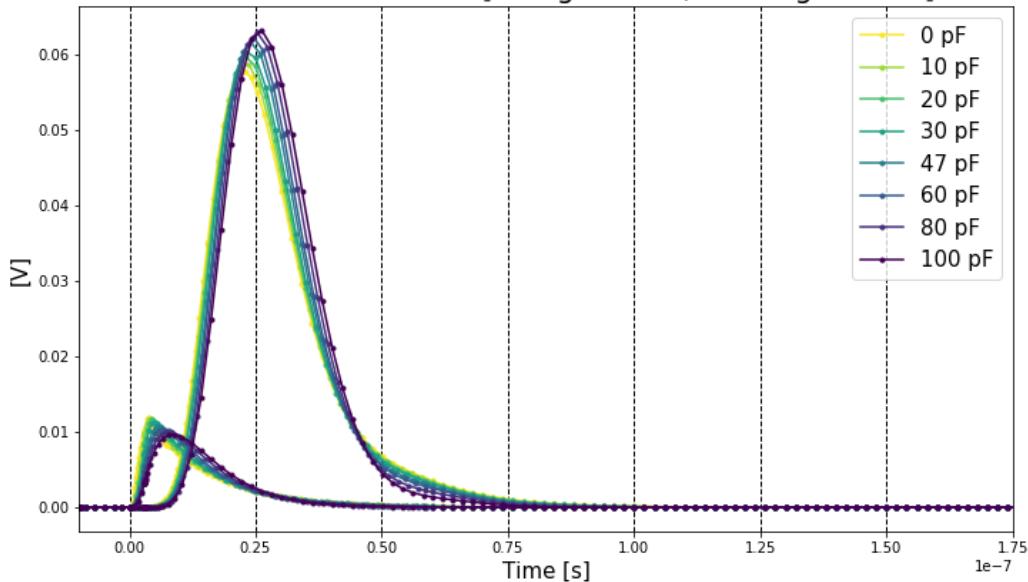
PA and SH waveforms [charge=10fC, PA range 160fC]



PA and SH waveforms [charge=10fC, PA range 80fC]

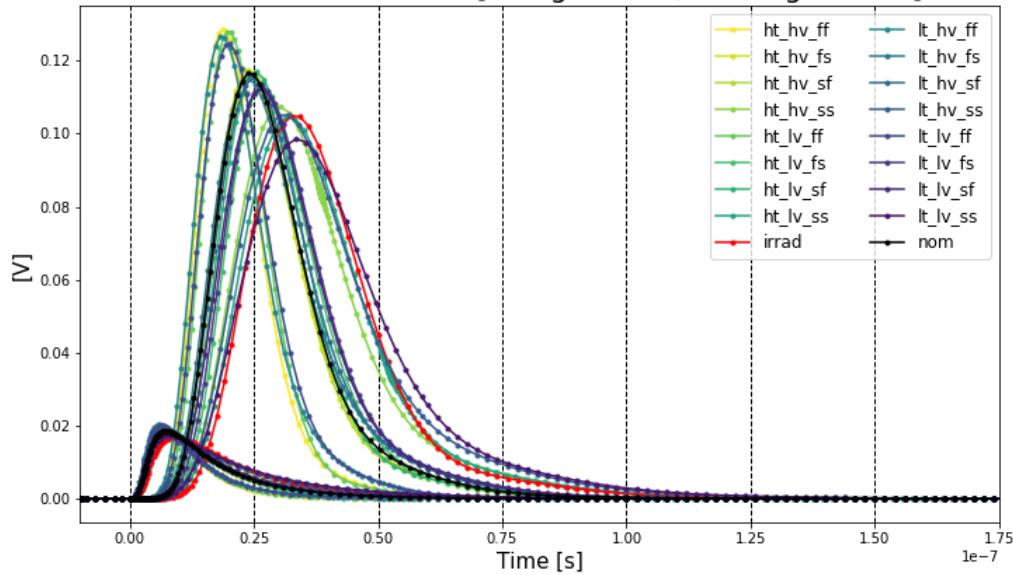


PA and SH waveforms [charge=10fC, PA range 320fC]

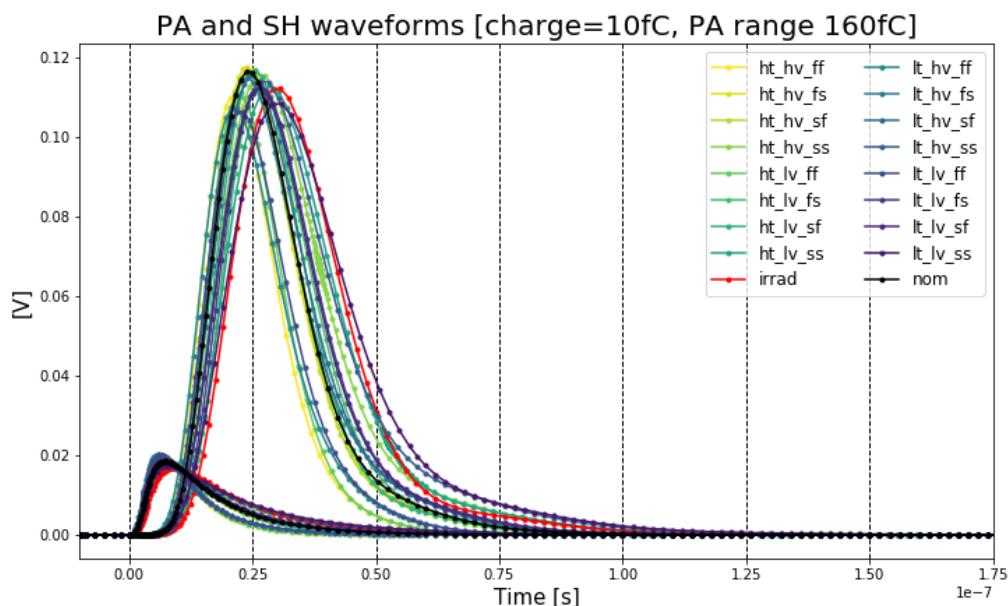


The waveforms below show the preamplifier and the shaper outputs as a function of the PVT variations.

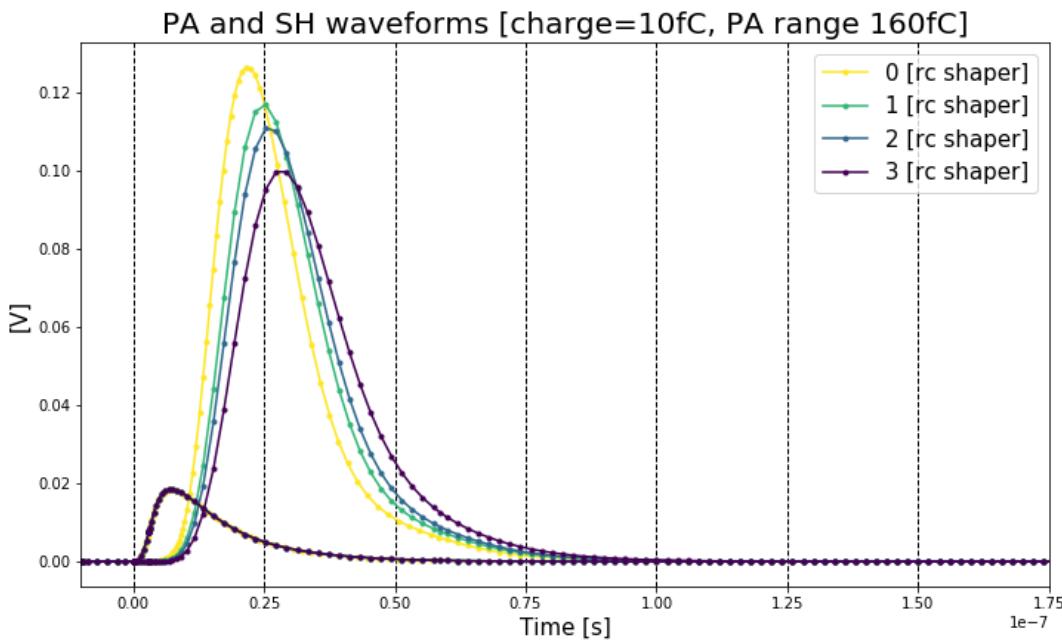
PA and SH waveforms [charge=10fC, PA range 160fC]



And now with trimmed shaping time.



The shaping time parameter allow the user to tune the peaking time to set it around 25 ns.



If those parameters are not sufficient, the user can tune as well the bandwidth of the RTR amplifiers (bias of the input differential stage).

#### 1.1.4. Discriminators

There are two discriminators per channel, one for the TOT measurement, the other for the TOA.

The TOT discriminator is connected to the outpa output of the preamplifier (160 – 200 mV). The TOA discriminator is connected to the outCf\_pa output of the preamplifier (~ 500 mV). Two global 10b-DACs

allow the user to adjust the thresholds of the discriminators and two local trimming 6b-DACs allow to reduce the dispersion per channel. The 10b-DAC have typically 1 mV LSB, the trimming 6b-DAC have typically 0.25 mV LSB.

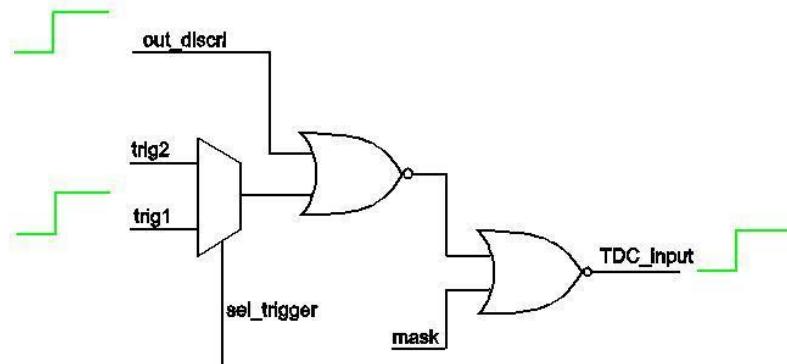
- $\text{Toa\_Threshold} = \text{Toa\_vref<9:0>} - \text{Trim\_dac\_toa<4:0>}$
- $\text{Tot\_Threshold} = \text{Tot\_vref<9:0>} - \text{Trim\_dac\_tot<4:0>}$

There are two external trigger inputs available; typically, in the case when the user wants to calibrate the TOT and TOA, he can send a trigger for the TOA discriminator and the other for the TOT discriminator. He can also send a trigger for a channel, and the other for a neighbouring channel.

The two discriminators outputs can be masked per channel as well.

The following figures shows the principle of the external trigger usage. The Trig1/2 are without effect when they are tied to 0. To send a trigger to a chosen channel, all the others must be masked.

Pull-down resistors are put on the Trig1/2 ports (comparing to V2 where this was not done).



Regarding the TOT discriminator, when it is triggered (output at 0), it enables a constant current source which discharges the preamplifier so that the duration of the preamp signal is proportional to the input charge. This current source can be adjusted over 6 bits in order to adjust the width of the TOT (nominal specification is 200ns for 10 pC).

The two discriminators outputs can be probed and looked at on a scope.

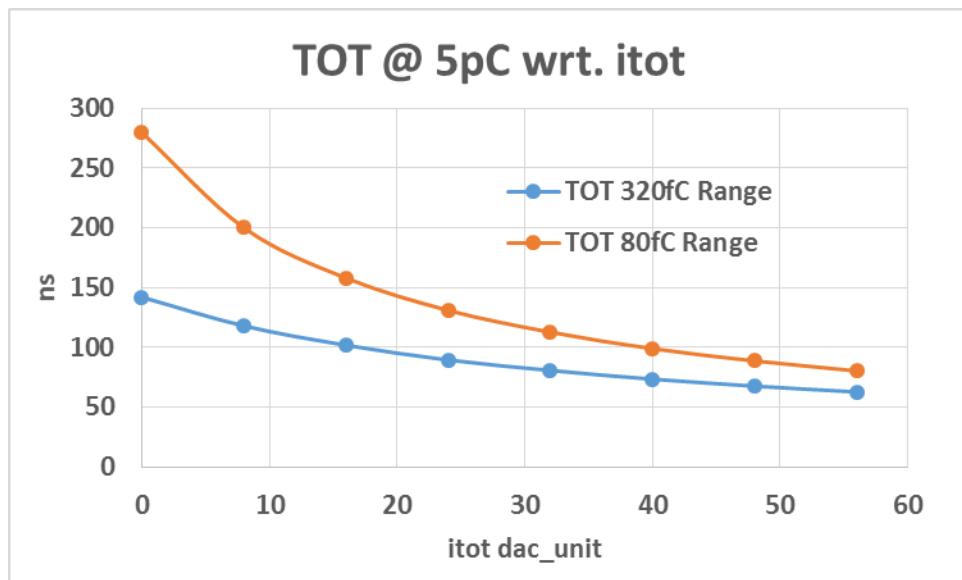
### *Discriminators parameters*

Name	# bits	I2C Sub-block / sub-address	Description
<b>Trim_dac_toa&lt;4:0&gt;</b>	5	Channel-wise	Local 5b-DAC for TOA threshold tuning
<b>Trim_dac_tot&lt;4:0&gt;</b>	5	Channel-wise	Local 5b-DAC for TOT threshold tuning
<b>Mask_toa</b>	1	Channel-wise	TOA discri output mask ("1" = masked)
<b>Sel_trigger_toa</b>	1	Channel-wise	External trigger selection for TOA ("0" = Ext Trig1; "1" = Ext Trig2)
<b>Sel_trigger_tot</b>	1	Channel-wise	External trigger selection for TOT ("0" = Ext Trig1; "1" = Ext Trig2)
<b>Mask_tot</b>	1	Channel-wise	TOT discri output mask ("1" = masked)

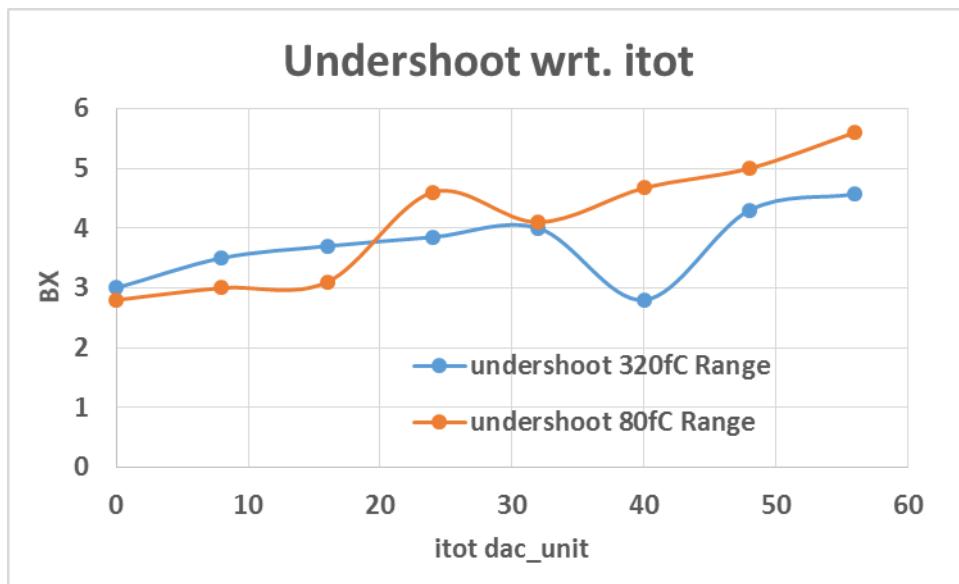
<b>Probe_tot</b>	1	Channel-wise	TOT discri output probe
<b>Probe_toa</b>	1	Channel-wise	TOA discri output probe
<b>ON_toa</b>	1	Global-analog	"1" = enable TOA discri bias
<b>ON_tot</b>	1	Global-analog	"1" = enable TOT discri bias
<b>Dac_itot&lt;5:0&gt;</b>	6	Global-analog	6b-DAC for TOT gain tuning ("000000" = no feedback current as the original TOT architecture)
<b>En_hyst_tot</b>	1	Global-analog	"1" = enable the TOT discri hysteresis
<b>Pol_trig_toa</b>	1	Global-analog	Polarity of the TOA discri output
<b>Tot_vref&lt;9:0&gt;</b>	10	Voltage references	TOT threshold global value
<b>Toa_vref&lt;9:0&gt;</b>	10	Voltage references	TOA threshold global value

The following figure shows the TOT duration at 5 pC injected charge as a function of the constant current itot. (the specification gives 100ns TOT at 5pC)

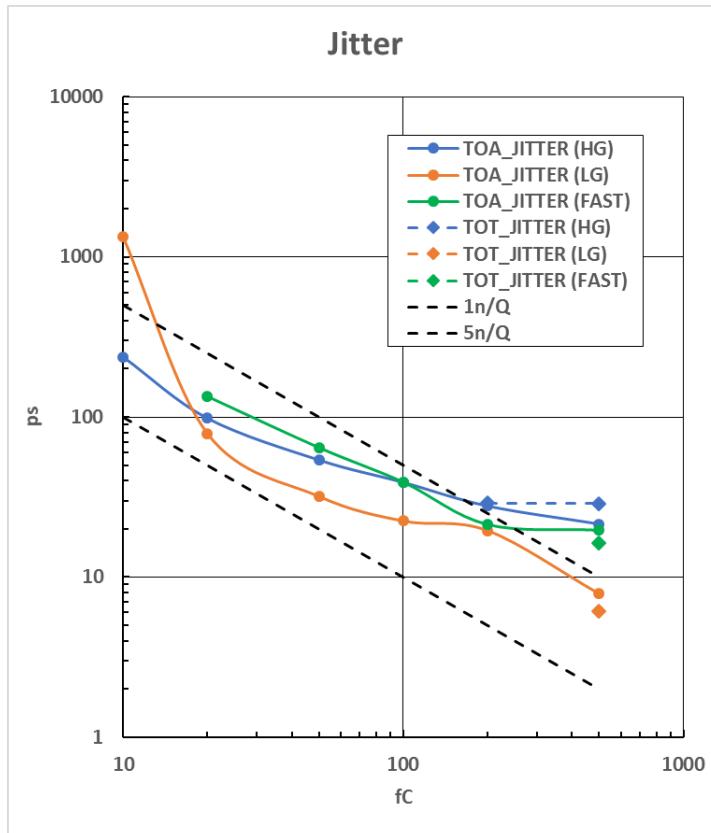
Clarify/update plots for HGCROC3 if needed

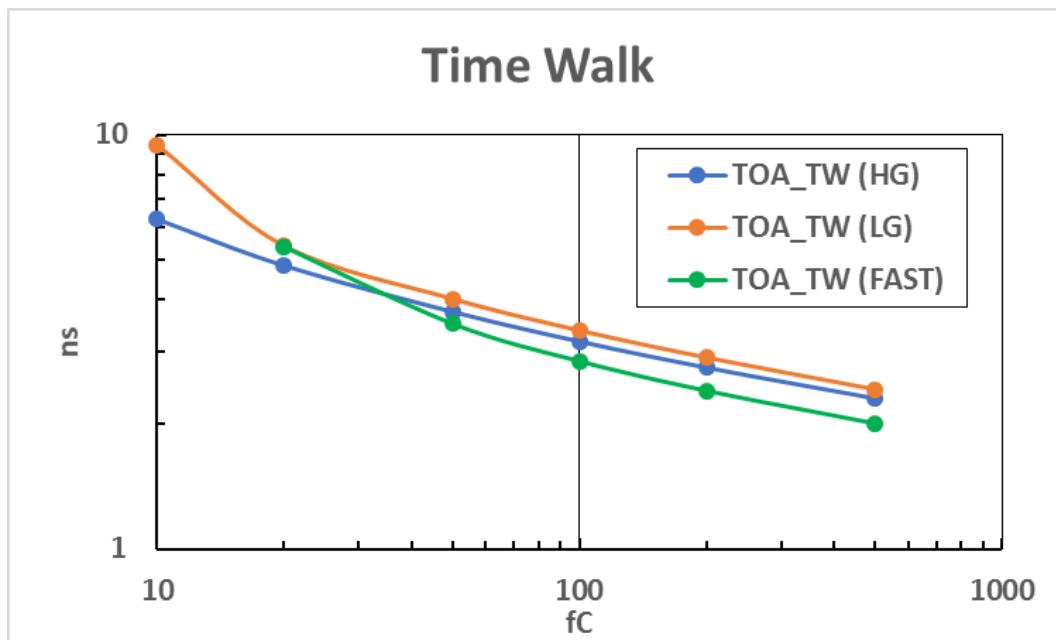


The following figure shows the duration of the "TOT dead time" expressed in Bunch Crossing count, namely the duration from the end of the TOT pulse up to the moment where the ADC gets the pedestal back. (to be noticed that itot=0 corresponds to the very first design)



The two next figures give respectively the jitter and time walk curve as a function of the input charge.

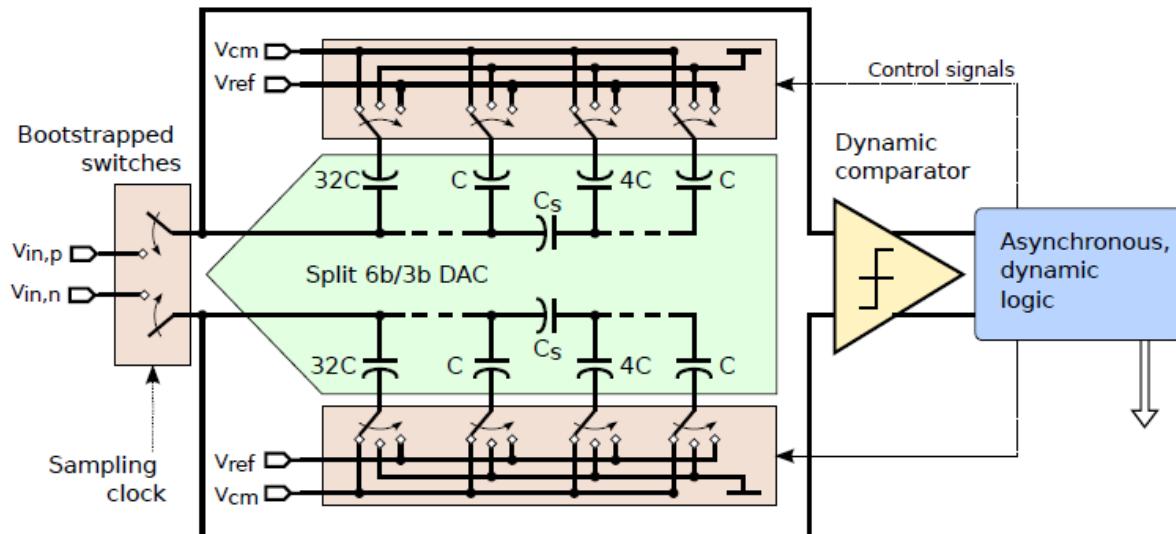




## 1.2. Mixed-signal blocks

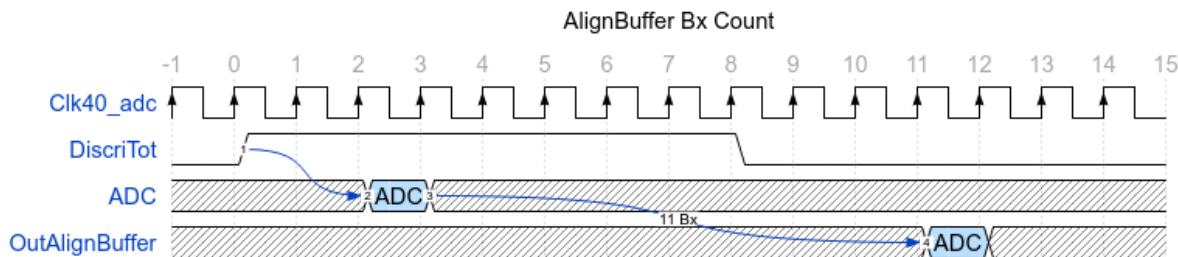
### 1.2.1. 10bits ADC and Align Buffer

The HGCROC3 contains a 10b SAR ADC, designed by AGH in Krakow. The ADC's vrefm reference voltage is tied to ground.



The 10 bits data provided by the ADC are sent to an Align Buffer to align them to the TOT and TOA data. Indeed, the TDCs providing the TOT measurement introduce a latency due to the duration of the TOT itself (this latency is tuneable in the TDCs). The ADC + Align Buffer have a fixed latency of 11 bunch

crossings: sampling at BC=1, ADC data available at BC=2, data at the Align Buffer outputs at BC=11 (see next chronogram).



### ADC parameters

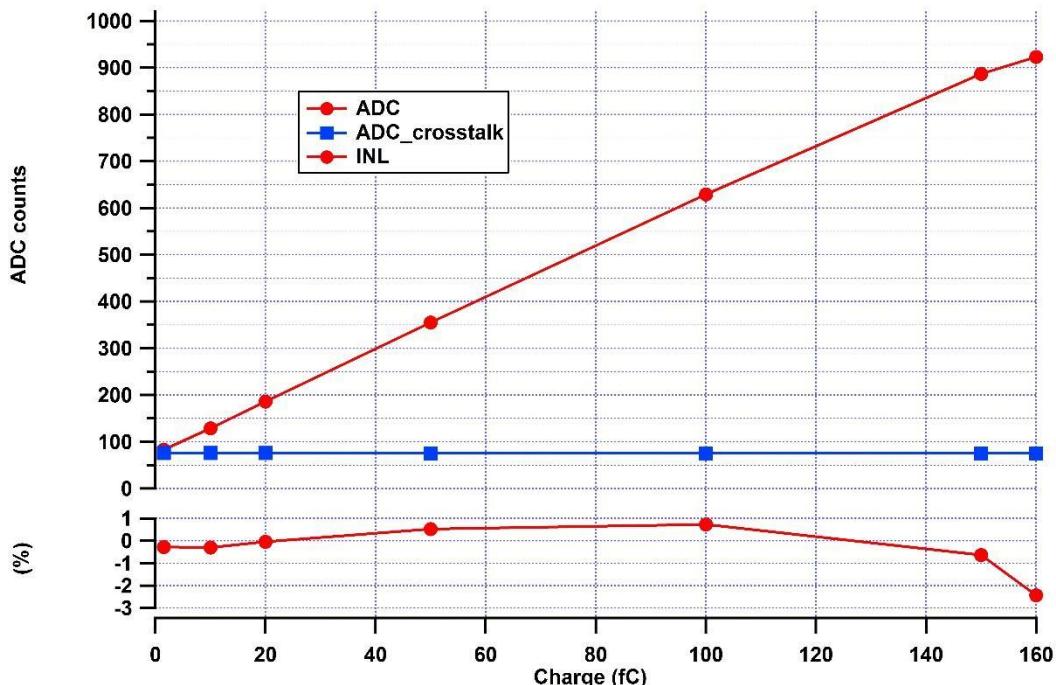
Name	# bits	I2C Sub-block / sub-address	Description
<b>Mask_adc</b>	1	Channel-wise	"1" = ADC clock off
<b>maskAlignBuffer</b>	1	Channel-wise	"1" = AlignBuffer clock off
<b>Adc_pedestal&lt;7:0&gt;</b>	8	Channel-wise	ADC pedestal value (use in Trigger path)
<b>ExtData&lt;9:0&gt;</b>	10	Channel-wise	Forced ADC data (enable ExtData by SelExtADC parameter: bit 3, register 10, "Global Analog")
<b>Clr_ShaperTail</b>	1	Global-analog	Force ADC to 0 for 2 BXs after the TOT to remove the undershoot
<b>SelRisingEdge</b>	1	Global-analog	"1" = AlignBuffer provides data on rising edge
<b>SelExtADC</b>	1	Global-analog	"1" = Forced ADC data send to the DRAM
<b>Clr_ADC</b>	1	Global-analog	Force ADC to 0 when TOT signal @ 1
<b>Ref_adc&lt;1:0&gt;</b>	2	Global-analog	Input stage current of the Ref ADC OTA
<b>Delay40&lt;2:0&gt;</b>	3	Global-analog	Delay tuning for bits <4:0> "000" = faster conversion
<b>Delay65&lt;2:0&gt;</b>	3	Global-analog	Delay tuning for bits <6:5> "000" = faster conversion
<b>Delay87&lt;2:0&gt;</b>	3	Global-analog	Delay tuning for bits <8:7> "000" = faster conversion
<b>Delay9&lt;2:0&gt;</b>	3	Global-analog	Delay tuning for bit <9> "000" = faster conversion
<b>ON_ref_adc</b>	1	Global-analog	"1" = enable ADC ref OTA
<b>Pol_adc</b>	1	Global-analog	ADC input swap

This Align buffer outputs ADC raw data for the DAQ path and ADC with pedestal subtraction for the trigger path.

Only for the trigger path, the user can choose to force ADC value to 0 during TOT integration by setting the parameter **Clr\_ADC** to 1. Moreover, to let the analog part return to its baseline after a TOT, the user can choose to force ADC to 0 for the next two bunch crossing by setting **Clr\_ShaperTail** to 1 (undershoot < 2 Bx after a TOT integration).

The following plot shows the linearity and the INL expressed in % of the ADC range in the default parameters setting.

### ADC gain: channel & crosstalk



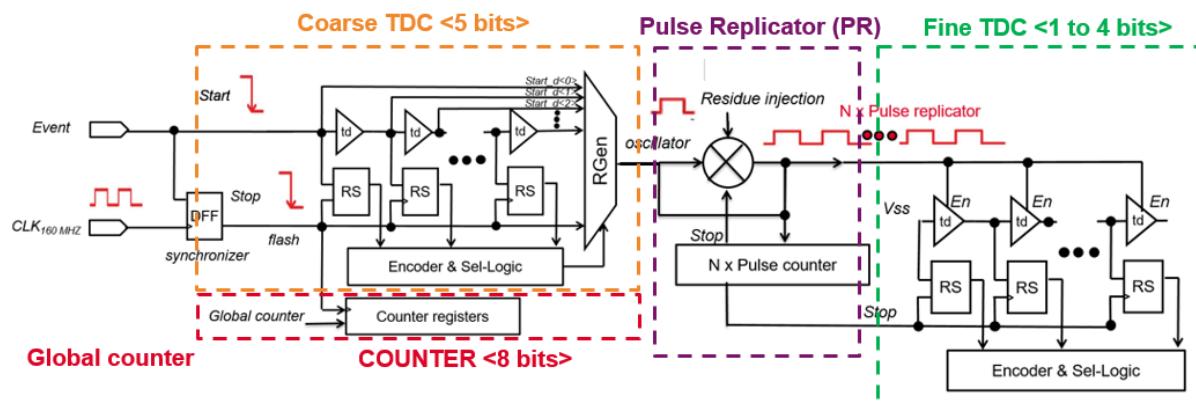
#### 1.2.2. TOT and TOA TDCs

One TDC block handles the TOA and TOT measurements. It was designed by the CEA IRFU group in Saclay. The two following tables give the specifications respectively for the TOA and the TOT.

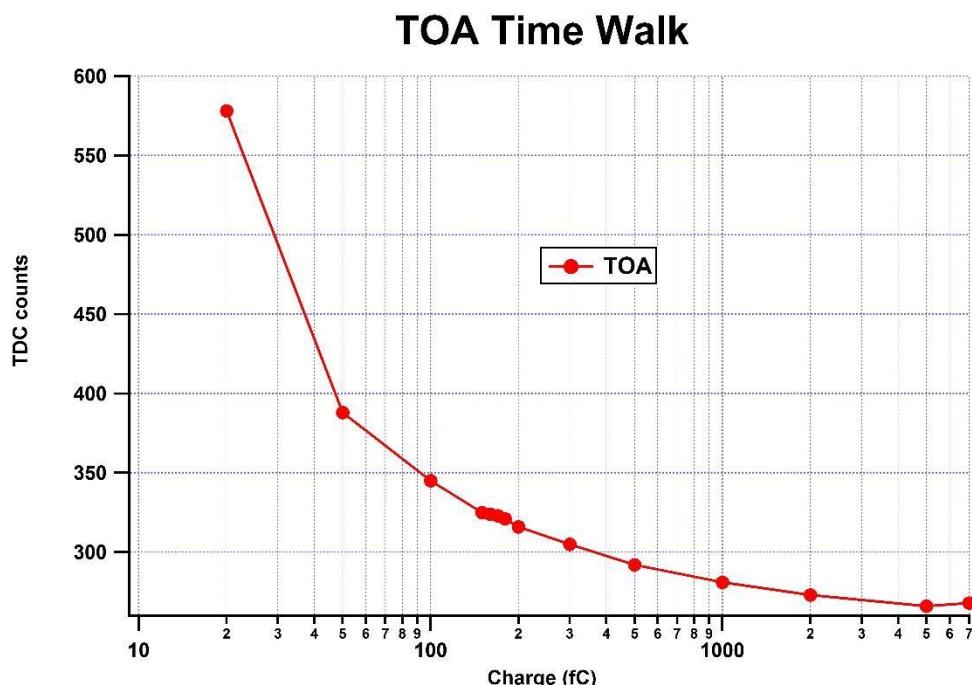
TDC ToA specifications	
Resolution	about 25 ps RMS
Range	10 bits over 25 ns
Conversion rate	> 40 MHz (bunch clock)
Power consumption	< 2 mW / channel
Area	Pitch 120 µm
Technology	TSMC 130 nm
Temperature	-30 °C

TDC ToT specifications	
Resolution	< 50 ps RMS
Range	12 bits over 2-200 ns
Min time between hits	25 ns
Power consumption	< 2 mW / channel
Fixed latency	12 clock periods
Technology	TSMC 130 nm
Area	Pitch 120 µm
Temperature	-30 °C

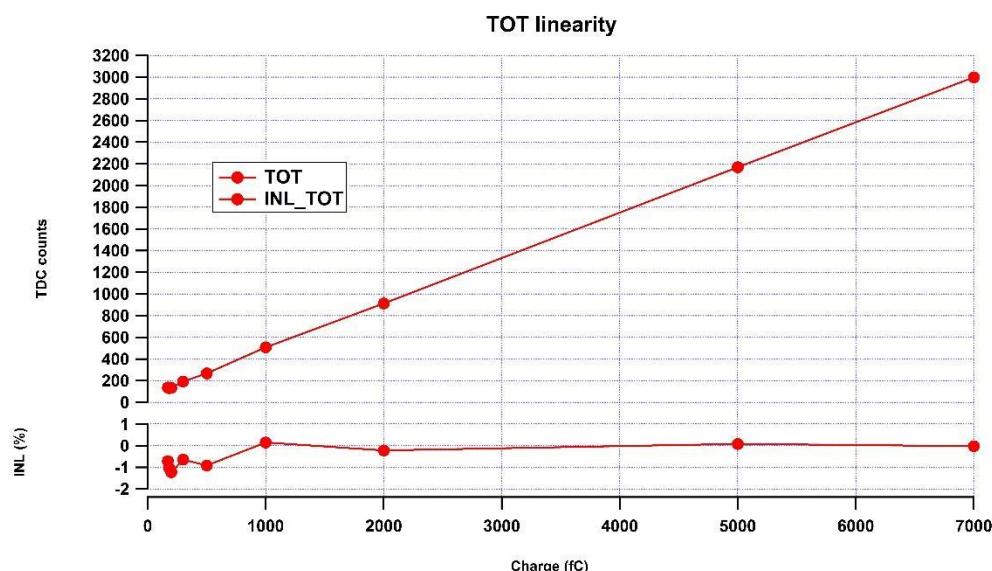
The schematic below gives an overview of the TDC circuitry.



The following plot shows the digitized TOA time walk of HGCROC2.



The following plot shows the TOT linearity and the INL expressed in % of the dynamic range.



### 1.3. Digital blocks

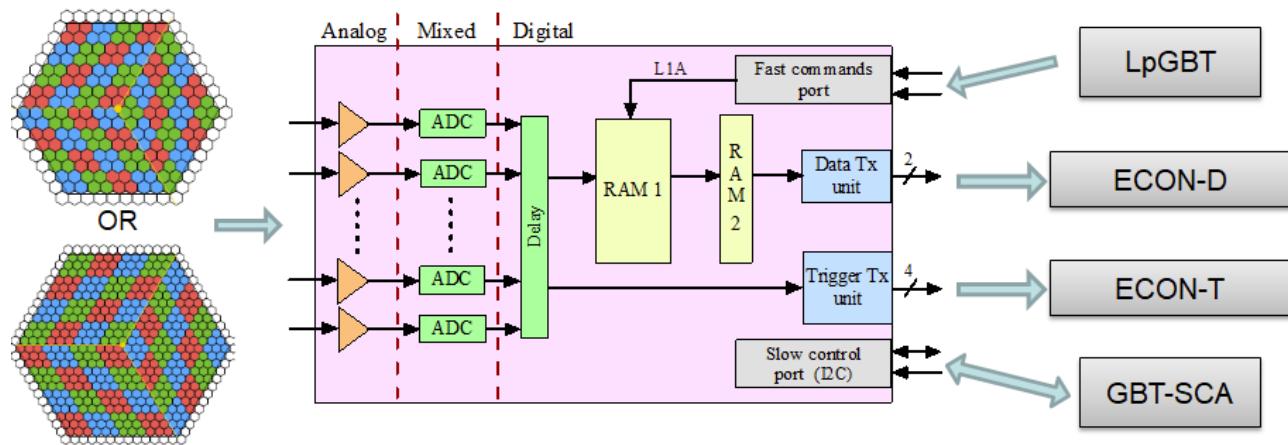
The main specifications of the digital part of the ASIC are described in the table below.

Analog channels	1. 72 analog “normal” channels 2. 4 for common mode channels 3. 2 for calibration
Acquisition mode	Continuous @ Bx rate (40 MHz)
L1 functionality	With derandomizer
L1-Trigger rate	1 Mhz (max 32 consecutive)
DAQ readout mode	<ul style="list-style-type: none"><li>Triggered by L1A</li><li>Normal or Tests modes</li></ul>
Trigger data type	7-bit sums (4 or 9 channels)
DAQ data type	32 bits / channels (40 words/frame)
Trigger output port	4 x CLPS @ 1280 Mbps (2 used for 9ch sums)
DAQ output port	2 x CLPS @ 1280 MBps

HGCROC2 integrates 72 channels to readout

- 192 channels sensor with a 64-ch configuration

- 432 channels sensor with a 72-ch configuration

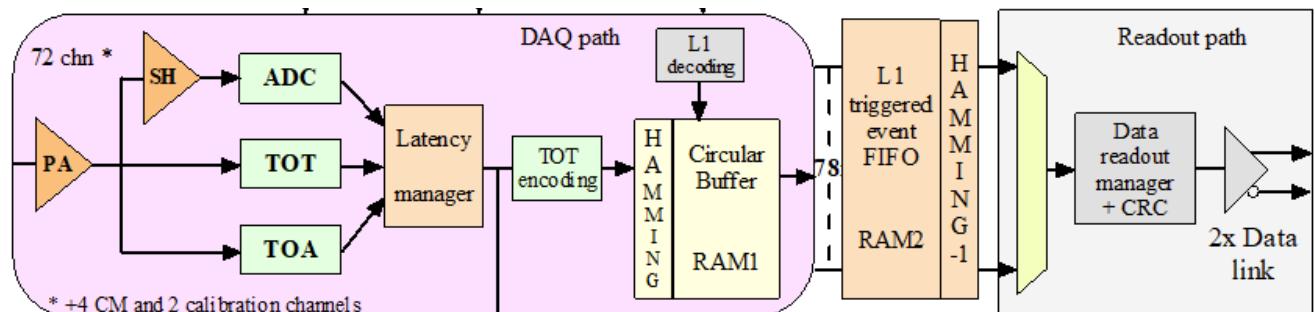


### 1.3.1. Data path

Following the Latency Manager block, the 3 pieces of channel information (ADC, TOT and TOA) are fed into the data path. The ADC values in the data path are without pedestal subtraction.

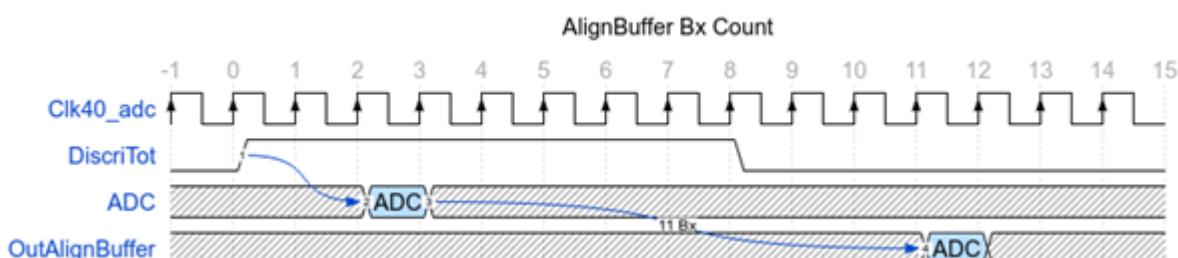
Common mode channels provide only the ADC data.

The figure below describes the functionality of the data path.



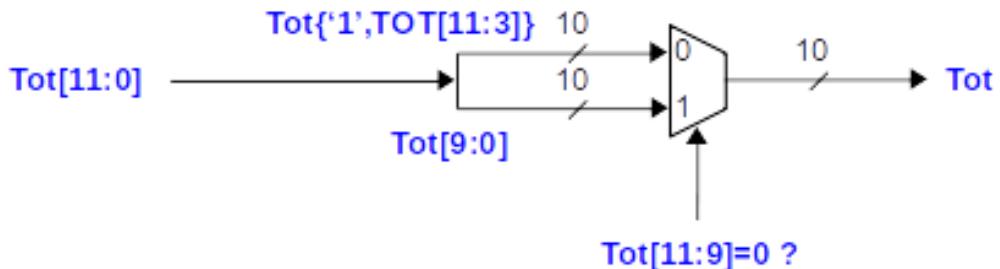
By slow control, the user can force a programmed pattern to be sent into RAM1. Otherwise, the raw ADC data is always sent to the RAM1.

The figure below shows the chronogram of the AlignBuffer.



### 1.3.1.1. TOT compression

The TOT data are compressed from 12 bits to 10 bits. This operation is done in the digital block.



### 1.3.1.2. Data Path Content

For each channel, the 30 bits of the Data path content is populated with values shown in Table 1. Three digitized values are selected depending on the crossed threshold for the charge. The ADC values in the data path are without pedestal subtraction.

The previous “calibration mode” is now renamed “characterization mode”: it is selected by slow control. In this mode, we always have ADC, TOT, TOA of the event readout: this mode is dedicated for characterization and debugging.

	ADC (t-1)	ADC (t)	TOT	TOA	Charge collection	Data type
1	x	x		x (=0)	Q < TOA_thr AN	Normal
2	x	x		x	Q < TOT_thr AN	Normal
3	x		x	x	Q > TOT_thr AN	Normal
4		x	x	x		Characterization

Table 1: The Data Content

Only for regular and calibration channels, two flags are added (at MSB positions) to form a 32 bits:

1. TOT-Complete, Tc (MSB i.e. bit 31), (applicable in cases 3 and 4): indicates a TOT value in the current 32-bit word.
2. TOT-In-Progress, Tp, (applicable in all cases): it indicates a TOT operation in progress (integration or undershoot). During this phase, the ADC value is valid but may be saturated (TOT integration).

#### Tc and Tp Interpretation:

- 2b00: The TOT is not in operation (not busy), normal behaviour for ADC data
- 2b01: The TOT is busy (integration or undershoot phase), Tp also highlights the fact that provided ADC correspond to saturation (during integration) or undershoot
- 2b11: The TOT value is output, normal behaviour with TOT data (ADC value is between saturation and undershoot)
- 2b10: should only appear when the TOT value is near the threshold (TOT value is valid)

**Tc and Tp inside 32 bits Output Data Frame related to table 1 are detailed above:**

0	Tp	10b ADC-1	10b ADC	10b TOA
---	----	--------------	------------	------------

**Case 1 and 2**

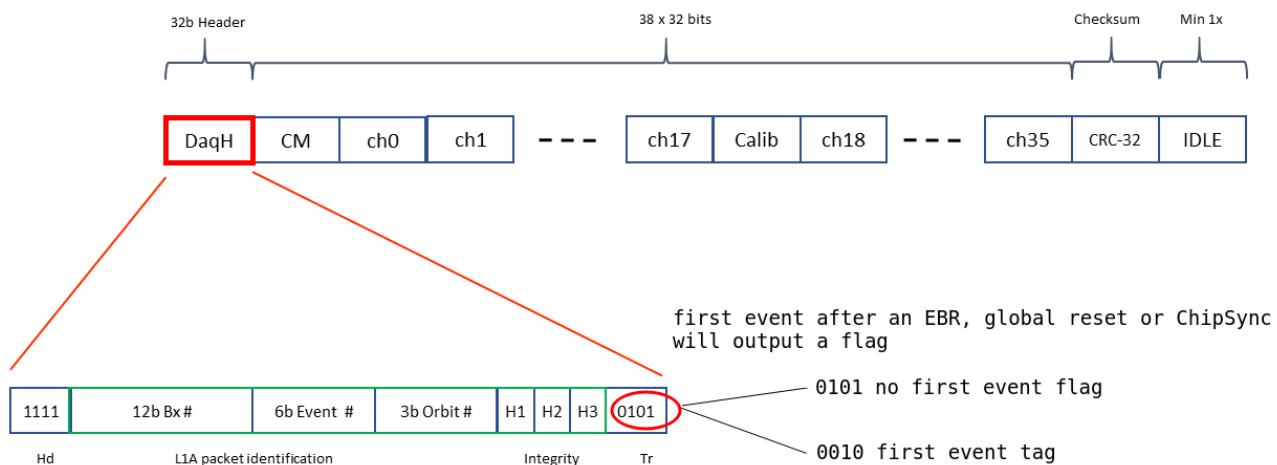
1	Tp	10b ADC-1	10b TOT	10b TOA
---	----	--------------	------------	------------

**Case 3**

Tc	Tp	10b ADC	10b TOT	10b TOA
----	----	------------	------------	------------

**Case 4**

### 1.3.1.3. Frame Description



- Bx#: Value of BC on 12 bits.
- Event#: Value of EC on 6 bits, to detect if a L1 trigger has been sent but related data wasn't saved (RAM2 full)
- Orbit#: Value of Orbit Counter OB on 3 bits.
- H1: Error during hamming decoding in counters (Event and Bx/Orbit).
- H2: Error during hamming decoding in CM to CH17 (1st Quarter).
- H3: Error during hamming decoding in Calib to CH35 (2nd Quarter).

The Calib channel's data content is the same as a regular channel. But in HGCROC3, the data content was forced to case 4 (as described in the previous section) with ADC, TOT and TOA values. The Tc Tp bits are valid.

The 32-bit common mode word (CM) is given below (only ADCs so no Tc Tp bits here):

12'b0	ADC CM0	ADC CM1
-------	---------	---------

The packet integrity is checked by the CRC (Cyclic Redundancy Check) on 32 bits data width. This module takes 39 packets of datas in input (Header, CM, Chn0, ..., Chn17, Calib, Chn18, ..., Chn35) and sends the CRC packet at the end of the frame (packet 40th) to the serializer.



The polynomial to apply is 0x04C11DB7:

$$X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$$

The verification of CRC coding is done by 2 different online CRC coder:

### 1st Example

[http://www.sunshine2k.de/coding/javascript/crc/crc\\_js.html](http://www.sunshine2k.de/coding/javascript/crc/crc_js.html)

CRC width: Bit length:  CRC-8  CRC-16  CRC-32  CRC-64

CRC parametrization:  Predefined  Custom

CRC detailed parameters: Input reflected:  Result reflected:   
Polynomial: **0x04C11DB7**  
Initial Value: **0x00000000**  
Final Xor Value: **0x00000000**

CRC Input Data:  String  Bytes  Binary string  
0x02

Show reflected lookup table:  (This option does not affect the CRC calculation, only the display)

**Calculate CRC!**

Result CRC value: **0x09823B6E**

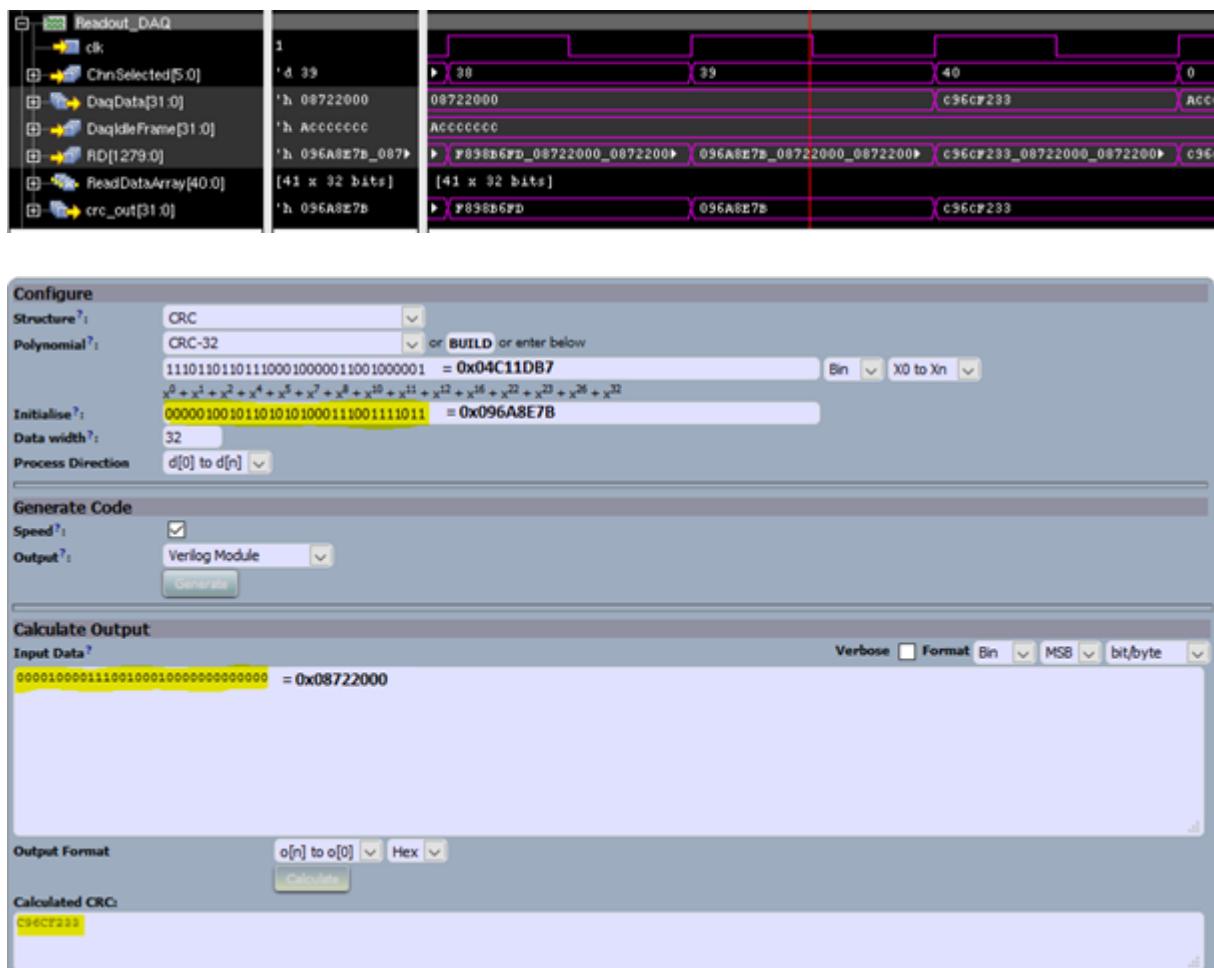
#### Lookup Table:

0x00000000	0x04C11DB7	0x09823B6E	0x0D4326D9	0x130476DC	0x17C56B6B	0x1A864DB2	0x1E475005
0x2608EDB8	0x22C9F00F	0x2F8AD6D6	0x2B4BCB61	0x350C9B64	0x31CD86D3	0x3C8EA00A	0x384FBDBD
0x4C11DB70	0x48D0C6C7	0x4593E01E	0x4152FDAA	0x5F15ADAC	0x5BD4B01B	0x569796C2	0x52568B75
0x6A1936C8	0x6ED82B7F	0x639B0DA6	0x675A1011	0x791D4014	0x7DDC5DA3	0x709FTB7A	0x745E66CD
0x9823B6E0	0x9CE2AB57	0x91A18D8E	0x95609039	0xB27C03C	0x8FE6DD8B	0x82A5FB52	0x8664E6E5
0xBE2B5B58	0xBAEA46EF	0xB7A96036	0xB3687D81	0xAD2F2D84	0xA9EE3033	0xA4AD16EA	0xA06COB5D

We can observe the validation of CRC-32 Algorithm with a simulation of CRC module alone. We write in the input an incremental value (0x00 to 0x07), and verify the output value with an online checksum calculator.

### 2nd Example

<https://leventozturk.com/engineering/crc>



In this test, we ran a full digital design's simulation. The CRC already coded consecutively **every 38<sup>th</sup>** 32bits packets. The CRC value is **0x096A8E7B** and the input is the **39<sup>th</sup>** (last with data) packet with the value **0x08722000**. The output is written in the **40<sup>th</sup>** packet and has the value **0xC96CF233** and the online CRC calculator gives the same result (value crosscheck).

One IDLE is attached at the end of the frame.

An idle packet is continuously sent out when no L1 trigger is activated. This Idle packet is configurable by slow control, the default idle word is **CCCCCCC**.

### 1.3.2. Trigger path

The data processing for the trigger path is composed as per below:

- Charge linearization over ADC/TOT range
- Sum of 4 or 9 channels depending on the sensor
- Charge compression to fit the bandwidth

By slow control, the user can choose to force ADC to 0 when TOT pulse = 1 and/or three more BX (for the undershoot). Then the pedestal subtraction is achieved. By slow control, the user can force to send a programmed ADC value (global parameter).

### 1.3.2.1. Disabling channel

By slow control, ADC and TOT data can be forced to 0 per channel. This is made in the digital block after the AlignBuffer.

### 1.3.2.2. Trigger path content

Following the AlignBuffer block, the 2 pieces of channel information (ADC and TOT) are fed into the trigger path. There follows a Charge Linearization block and a summing block to create Trigger Sum cells.

### 1.3.2.3. Charge Linearization Block

The Charge linearization block treats the ADC and TOT charge in different ways. The two paths are shown below. It is a scheme of principle as actually the pedestal subtraction is made in the AlignBuffer.

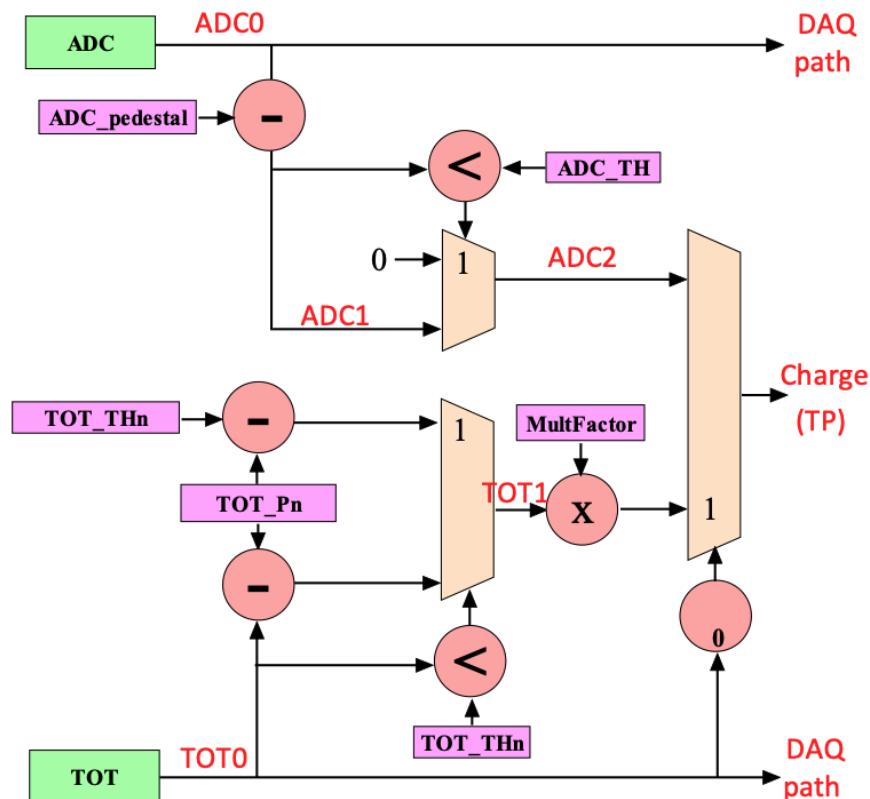


Figure: Charge Linearization Block

### 1.3.2.4. ADC Charge Processing

Referring to figure above, the ADC charge processing has 2 main steps

Step	Process name	Parameter	Action



1	Pedestal subtraction	ADC_pedestal (8 bits per channel)	If ADC0 > (ADC_pedestal + ADC_TH) then ADC2 = ADC0 - ADC_pedestal Else ADC2 = 0
2	Noise cancellation	ADC_TH (5 bits global per ROC)	

Note:

The constants are to be measured during a calibration phase and loaded via slow control. ADC\_Pedestal is a dc offset per channel and ADC\_TH is used for a noise cut.

#### 1.3.2.5. TOT Charge Processing & Selection of ADC value or TOT.

The TOT charge processing and selection has 4 main steps. For step 1, a positive value of TOT pedestal should be fed for calculation (which corresponds to Tot\_P\_Add parameter set to 0)

Step	Process name	Parameter	Actions
1	TOT pedestal subtraction	TOT_Pn (7 bits per group of 9 channels)	If TOT0 > TOT_THn then TOT1 = TOT0 - TOT_Pn Else TOT1 = TOT_THn - TOT_Pn
2	Plateau or linear selection	TOT_THn (8 bits per group of 9 channels)	
3	Convert time to charge (ADC units)	Multifactor (5 bits global, default value = 25)	
4	Selection between ADC and TOT		If TOT0 ≠ 0 then Charge = TOT1 x Multifactor Else Charge = ADC2

Notes:

TOT\_pedestal, TOT\_threshold and Multiplication factor are parameters to be measured during a calibration phase and loaded via slow control.

TOT0 is a 12 bit value obtained from the TOT block.

TOT pedestal, defines the TOT linear fit offset (should be positive so Tot\_P\_Add parameter must be 0).

TOT threshold defines the lower limit of the TOT linear part

Multiplication factor = Ratio between the TOT and the ADC LSB

Finally there is the selection between charge from the ADC or TOT.

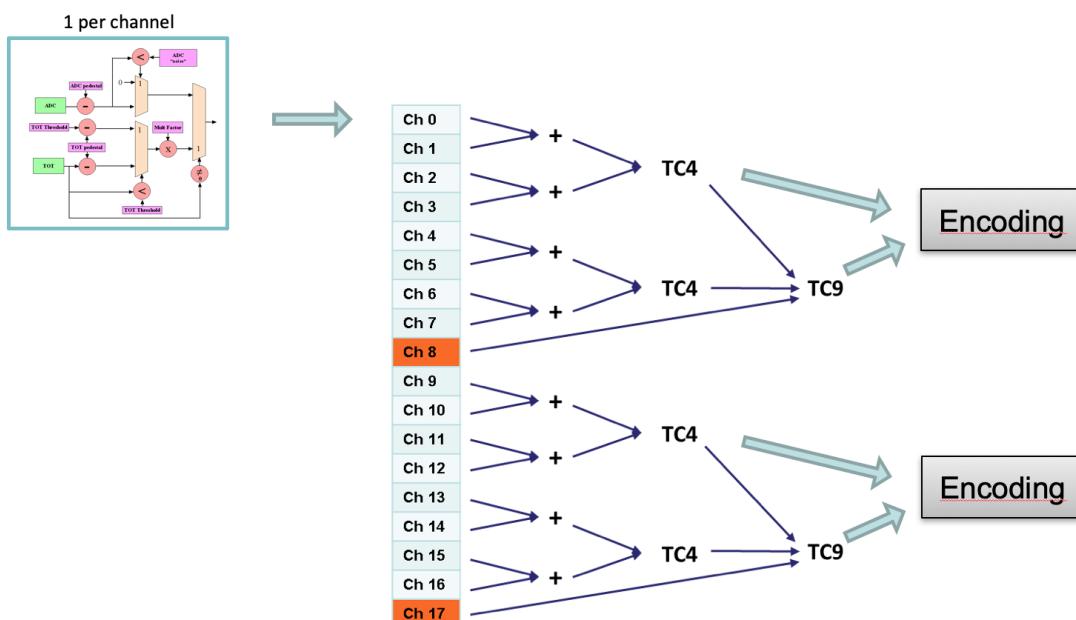
The value of the Multiplication factor allows to linearize the ADC range and the TOT range; it is coded over 5 bits in order to cope with the three typical gains:

- 31 for the 80fC ADC range
- 15.6 for the 160fC ADC range
- 7.8 for the 320fC ADC range

(The default slow control value is 25 for a theoretical 100fC ADC range.)

### 1.3.3. Trigger Cell Sums

The user can define the chip to sum charge in groups of 4 or 9 channels to obtain trigger sums. The selection between the sum by 4 (TC4) or 9 (TC9) is done by the ASIC parameter “SelTC4”.



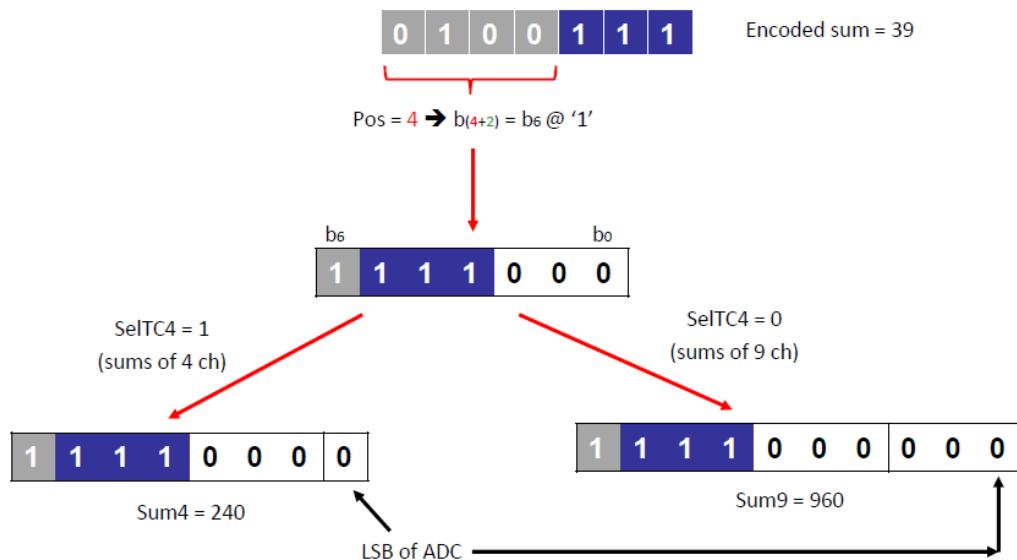
\* "orange" channels only when sum of 9 channels

Figure 5: Channel Mapping for Trigger Sums

Internally, the sum by 4 gives 19b and 21 bits for the sum by 9. We are using the encoding over 7 bits with 4 bits for the position of the MSB and the 3 next bits after the MSB. This maximum number of bits is limited to 18 ( $b_{17}$  to  $b_0$ ). So the sums are truncated by 1 LSB and 3 LSB before the encoding. Finally, we have a word over 18 bits and the encoding follows this logic:

- If sum  $\leq 7$  Position = “0000” and next 3 bits =  $b_2 b_1 b_0$
- Else if MSB @1 is  $b_n$  Pos =  $(n-2)_2$  and next 3 bits =  $b[n-1:n-3]$ 
  - o example, if  $b_{17}$  is 1 then Pos = “1111”

And so the decoding can be described as per below.



### 1.3.3.1. Frame description

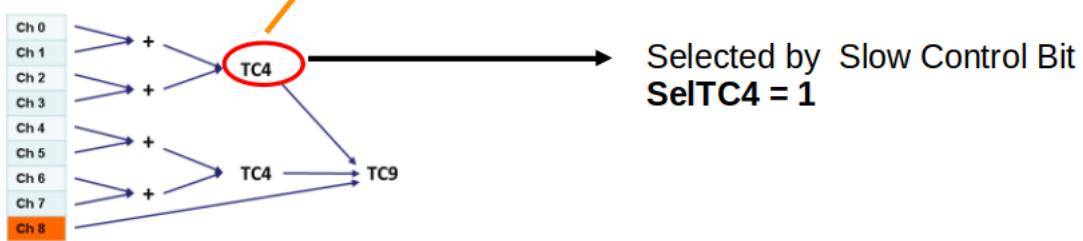
The data transmission is MSB first.

The sum by 4 or 9 is set by slow control.

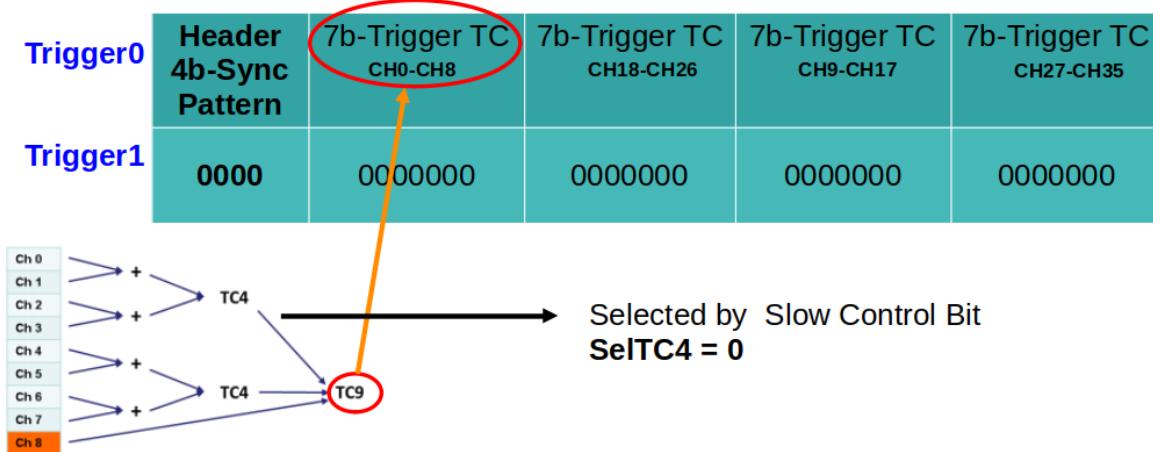
The Calibration and common mode channels are not in Trigger data.

The figure below shows the dataframe for the sum by 4 (half chip).

<b>Trigger0</b>	<b>Header 4b-Sync Pattern</b>	<b>7b-Trigger TC CH0-CH3</b>	<b>7b-Trigger TC CH4-CH7</b>	<b>7b-Trigger TC CH9-CH12</b>	<b>7b-Trigger TC CH13-CH16</b>
<b>Trigger1</b>	<b>Header 4b-Sync Pattern</b>	<b>7b-Trigger TC CH19-CH22</b>	<b>7b-Trigger TC CH23-CH26</b>	<b>7b-Trigger TC CH28-CH31</b>	<b>7b-Trigger TC CH32-CH35</b>

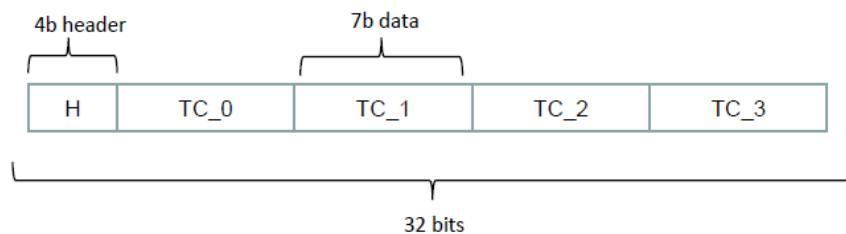


The figure below shows the dataframe for the sum by 9 (except during link reset T procedure and startup).



The table below gives an overview of the trigger path output format.

# Trig-link	2 or 4 (configurable)
Link electrical standard	CLPS
Link bitrate	1280 Mbps
Possible to switch-off unused link	Yes (partially)
Serialisation factor	32 (i.e. 32 bits per 25 ns)
Bits order	MSB first
Packet composition	header (4 bit) + payload (28 bits) 4 header = packet [31:28] 28 payload = packet[27:0]
Header	<ul style="list-style-type: none"> <li>- During startup : 4'b1010 on all links (enabled or disabled)</li> <li>- During run phase : <ul style="list-style-type: none"> <li>- LINKRESETROCT procedure: 4'h9 @ BCT else 4'hA on all links (enabled or disabled)</li> <li>- else : 4'h9 @ BCT else 4'hA for enabled links, 4'h0 for disabled links</li> </ul> </li> </ul>
Payload	<ul style="list-style-type: none"> <li>- During startup: configurable idle word (default 28'hCCC_CCCC) on all links (enabled or disabled)</li> <li>- During run phase: <ul style="list-style-type: none"> <li>- LINKRESETROCT procedure: configurable idle word (default 28'hCCC_CCCC) on all links (enabled or disabled)</li> <li>- else: 4 consecutive Trigger Cells (TC) for enabled links, 28'h000_0000 for disabled links</li> </ul> </li> </ul>
Trigger cell (TC) encoding	Floating point with 4 bits exponent and 3 bits mantissa Exponent= TC[6:3] Mantissa= TC[2:0]



The content of the trigger data is depending on the SelTC4 setting.

	H	TC0_0	TC0_1	TC0_2	TC0_3	72 ch ASIC Sums of 9 ch	64 ch ASIC Sums of 4 ch
Trig-link #0						TC0_0 Sum ch → 0 to 8	Sum ch → 0 to 3
						TC0_1 Sum ch → 18 to 26	Sum ch → 4 to 7
						TC0_2 Sum ch → 9 to 17	Sum ch → 9 to 12
						TC0_3 Sum ch → 27 to 35	Sum ch → 13 to 16
Trig-link #1	H	TC1_0	TC1_1	TC1_2	TC1_3	TC1_0 0	Sum ch → 19 to 22
						TC1_1 0	Sum ch → 23 to 26
						TC1_2 0	Sum ch → 28 to 31
						TC1_3 0	Sum ch → 32 to 35
Trig-link #2	H	TC2_0	TC2_1	TC2_2	TC2_3	TC2_0 Sum ch → 36 to 44	Sum ch → 36 to 39
						TC2_1 Sum ch → 54 to 62	Sum ch → 40 to 43
						TC2_2 Sum ch → 45 to 53	Sum ch → 45 to 48
						TC2_3 Sum ch → 63 to 71	Sum ch → 49 to 52
Trig-link #3	H	TC3_0	TC3_1	TC3_2	TC3_3	TC3_0 0	Sum ch → 55 to 58
						TC3_1 0	Sum ch → 59 to 62
						TC3_2 0	Sum ch → 64 to 67
						TC3_3 0	Sum ch → 68 to 71

H is always a 4-bit “1010” except when crossing a trigger value (BXCpt=BCT) where it is “1001”.

#### 1.3.4. Digital parameters

The table below gives all the parameters of the digital block for the data and trigger paths, thus there are two of them in the chip as there are two digital blocks for the both sides of the chip.



name	# bits	comment
PllLocked	1	If available, status of PLL locked
CmdSelEdge	1	0 : select fall edge for fast commands sampling (default)
SelTC4	1	1: sum by 4 / 0 : sum by 9
ClrAdcTot_Trig	36	1 : forcing 0 on ADC and TOT on each channel
IdleFrame	28	Default 28 LSB « hCCCCCCC » of idle DAQ/T frame
L1Offset	9	L1 offset corresponding to L1 latency
Bx_offset	12	BxCounter reset value
Bx_trigger	12	Generate Special header @ Bx = Bx_trigger
Adc_TH	5	Threshold corresponding to noise in ADC count
MultFactor	5	TOT vs ADC ratio for linearization (default ~25)
Tot_P0,Tot_P1,Tot_P2,Tot_P3	7	TOT pedestal used in TP (common to 9 channels)
Tot_TH0,Tot_TH1,Tot_TH2,Tot_TH3	8	TOT threshold used in TP (common to 9 channels)

SelTC4, namely Select Trigger Cell of 4 channels, allows the user to select the sum mode: set to 1 to sum 4 channels, otherwise sum by 9.

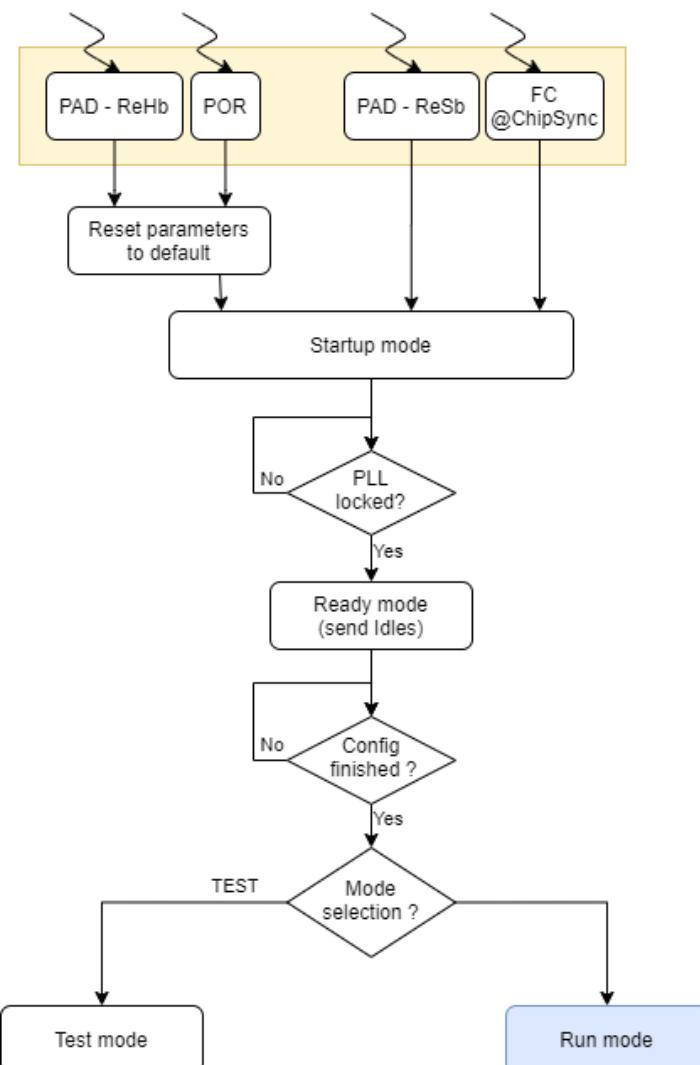
SelTC4	# channels used	# ch in each TC	# Daq-link @ 1,28G	# Trig-link @ 1,28G	unused channels
0	72	9	2	2	-
1 (default)	64	4	2	4	(8, 17, 18, 27) (44, 53, 54, 63)

### 1.3.5. Description of the operating modes of the chip

The hard reset pin ReHb (CMOS input, active low) resets everything inside the ASIC (PLL, I2C, FSM, counters, parameters). The soft reset pin ReSb (CMOS input, active low) resets everything except the slow-control parameters.

#### 1.3.5.1. Start-up sequence

This section describes the start-up sequence of HGCROC3.



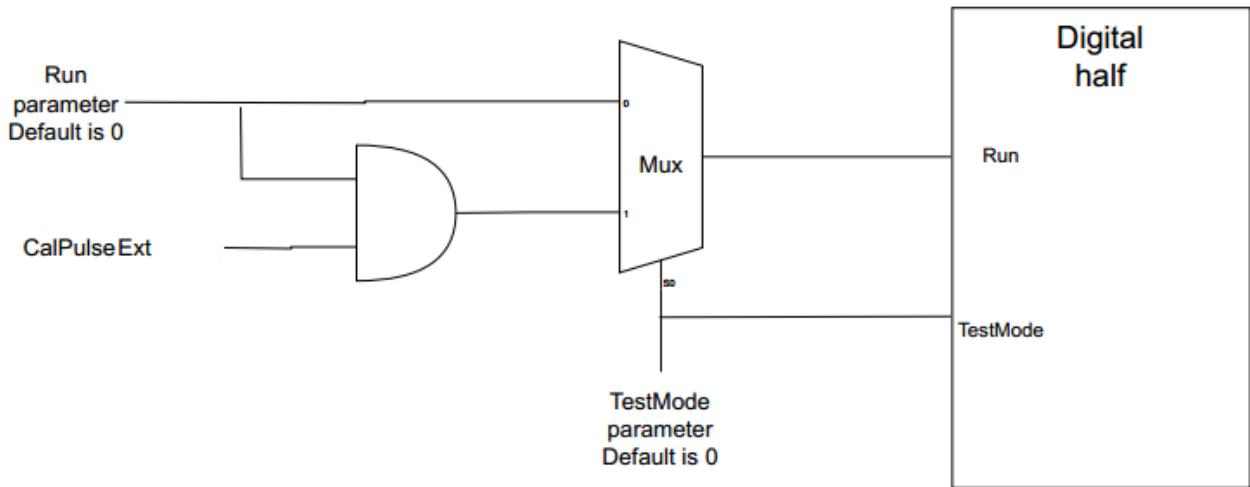
When the chip is powering on, the Power-On-Reset (POR) block applies a reset (like the ReHb). The POR block can be disabled by a pin (Power-On disable).

There are 4 main states in this sequence:

- Startup mode: This is the state just after a reset and the ASIC state is defined by the default parameters. It waits there until the PLL is locked
- Ready mode: In this state the PLL is locked and the user should load the ASIC parameters through I2C. To get out of this state, a special “config\_OK” parameter should be written. When all the parameters related to the serial links are loaded, the ASIC starts sending a forced 32b IDLE pattern through the 4 trigger links (4b header forced to 1010).
- Then depending of the chosen mode (ASIC parameter) loaded in the previous state, we enter in:
  - Run mode: this is the normal operational mode where the ASIC starts writing into DRAM physics data.
  - Test mode: this mode enables an extensive test of the DRAM (retention time). When tests are finished, the user should apply an hard reset to re-initiate the startup flow and go to the “Run mode”.

If TestMode = 0 ==> normal acquisition for HL-LHC = no change compared to ROC3a

If TestMode =1 ==> DRAM retention tests = need Run parameter + CalPulseExt fastcmd to start



Low Power mode disables the analog part of the chip: preamplifier, shaper, discriminators. But since the PLL is ON, the ADC, TDC, AlignBuffer will be ON.

The Power-On-Reset is a very critical block of the chip. It has been simulated with the special gate-leakage models and furthermore with additional 1 uA leakage current on the sensitive transistors. Moreover, its output is accessible on a spare pad.

### 1.3.5.2. RAM1 to RAM2 operation

For the DAQ path, the ASIC integrates 2 memories. The first one, RAM1, handles the storage of the data until the ASIC receives a L1A command. The second one, RAM2, is managed like a FIFO memory and stores the data extracted from RAM1 by the L1A command.

When the chip receives this L1A command, the data of the event which occurred at a programmed number of bunch-crossing earlier (value set by slow control: typically 12.5  $\mu$ s) is copied into the RAM2 with the value of the time tag and event counters. Then, the latter one (EC) is incremented.

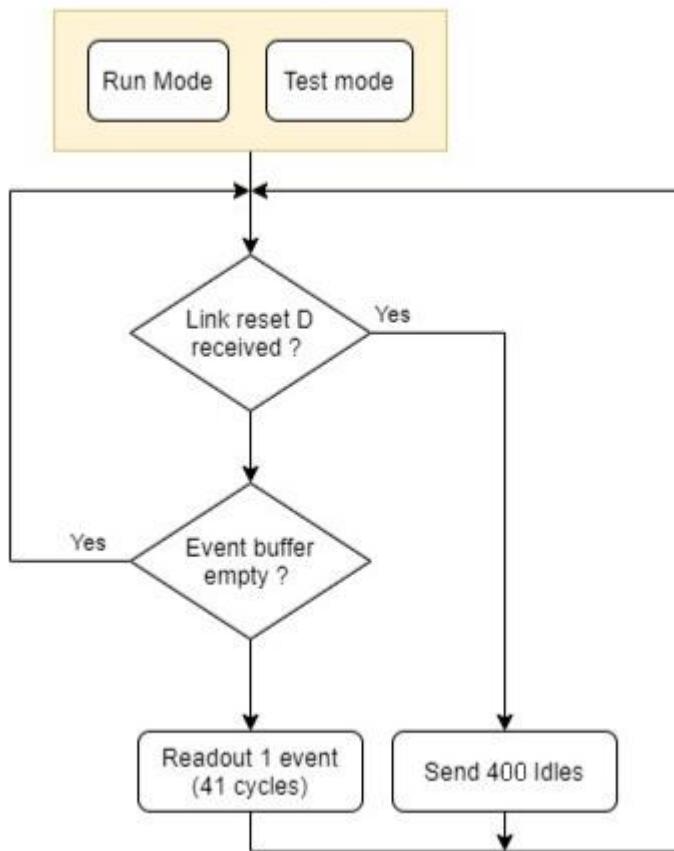
As the internal architecture is pipelined, the system is able to handle consecutive L1A.

### 1.3.5.3. RAM2 to SerDes State-Machine

This section describes the read-out operations from RAM2 (Event Buffer) to the DAQ serial links. As long as RAM2 is not empty, the chip sends out the data else it sends out the IDLE pattern. The RAM2 is handled like a FIFO circular buffer.

As described below, when the chip receives a Link-Reset-ROC-D fast command, it starts the link reset procedure. If an event readout is in progress, the link reset procedure is delayed after the end of the event readout.

If another Link-Reset-ROC-D command is received during the execution of a previous one, it's skipped.

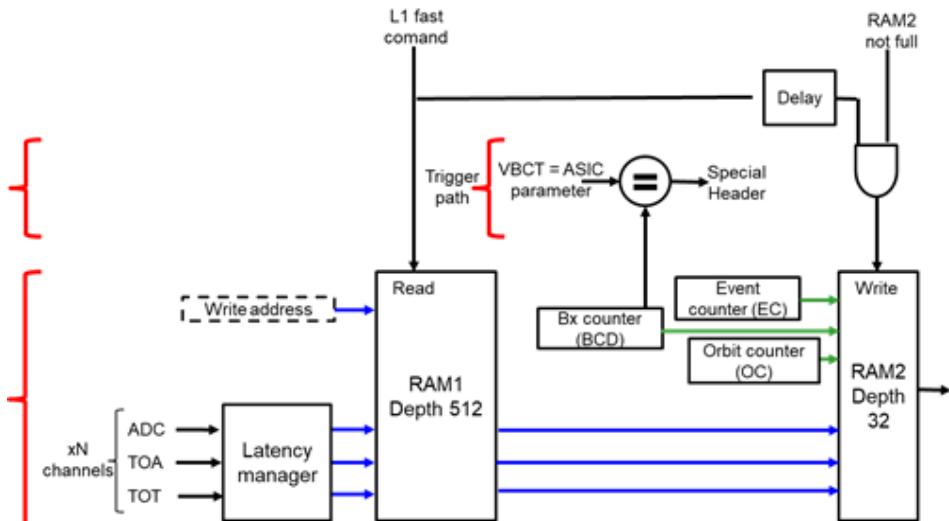


#### 1.3.5.4. Definition of the Time Tag Counters

The HGCROC chip has 3 internal counters to tag the data:

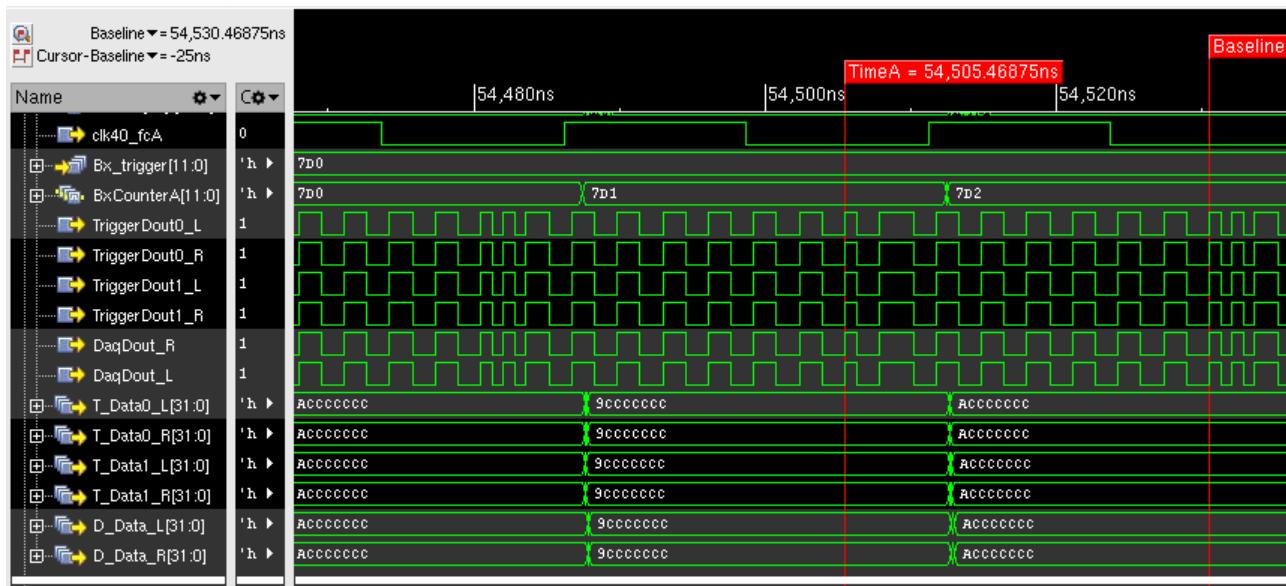
- Bunch crossing counter (BCD)
- Orbit counter (OC)
- Event counter (EC)

They are attached to the data extracted from RAM1 and copied to RAM2. BCD is also used to generate the special header in the trigger path when its value crosses a programmed one (slow control parameter on 12 bits). A global scheme of these counter is shown below:



The 3 counters are detailed below:

- BCD counter (BCD): 12-bit counter incremented every cycle (25 ns). Its reset value is given by a 12-bit programmable parameter (Bx\_offset: value 1 by default). Its range is from 1 to 3564 with a wrapping when the upper limit is reached. This counter is also used to generate the special header (binary 4 bits: 1001) when a programmed value (VBCT/BxTrigger) is crossed. This header occurs on output at the same time in all serial links (trigger or data), as shown below (post layout simulation):



- Event Counter (EC): 6-bit counter incremented every L1A (after data processing). Its reset value is 1 with a range from 0 to 63 (wrapping when the upper limit is reached). After a reset, the first event readout will have an event counter tag equal to 1.
- Orbit Counter (OC): 3-bit counter incremented every BCD wrap (i.e BCD equals 3564). Its reset value is 0 with a range from 0 to 7 (wrapping when the upper limit is reached).

#### 1.3.5.5. Definition of the Resets

- Bunch Crossing counter reset (BCR): from the fast command receiver, it resets the Bunch Crossing counter to its default value (parameter)
  - If we have a BCR followed by a L1A the event on link will be tagged bx=bx\_offset
- Event Counter reset (ECR): from the fast command receiver, it resets the counter to 1
  - If we have an ECR followed by a L1A the event on link will be tagged EC=1
  - If we have at same time an ECR and a L1A the event on link will be tagged EC=1
- Orbit Counter reset (OCR): from the fast command receiver, it resets the counter to 0
  - If we have an OCR followed by a L1A the event on link will be tagged EC=1 and OC=0
- ChipSync: This command is equivalent to an external soft reset (but only affects the digital part), it does not reset the Fast Command unit or analog part (like PLL). A BCR is needed after a ChipSync to resynchronize the time tagging counters with the whole system.



- Event Buffer Reset (EBR): from the fast command receiver, it resets only the pointers of the Event Buffer (DRAM2). The ASIC is able to accept a new L1A on the next clock cycle. If there is an event readout in progress, it finishes normally. An EBR also does an ECR.  
On the other hand, L1A prior to EBR (or at the same time) will be cancelled and will not be readout (due to the fact that EBR is suppressing L1A in the pipe). L1A coming 4 Bx (or more) before the EBR are processed normally (sequence is: L1A, 3 or more Idles, EBR).

During the DAQ readout, the first event after an EBR, a global reset or a ChipSync will be flagged on the 4 LSB of the 32b Header word:

- 0 "101" by default
- 0 "010" first event tag

- ReHb: it comes from an external pin and resets everything (analog + digital)
- ReSb: Same ReHb but leaves the SC parameters programmed.



## 1.4. Interfaces blocks

### 1.4.1. Fast command

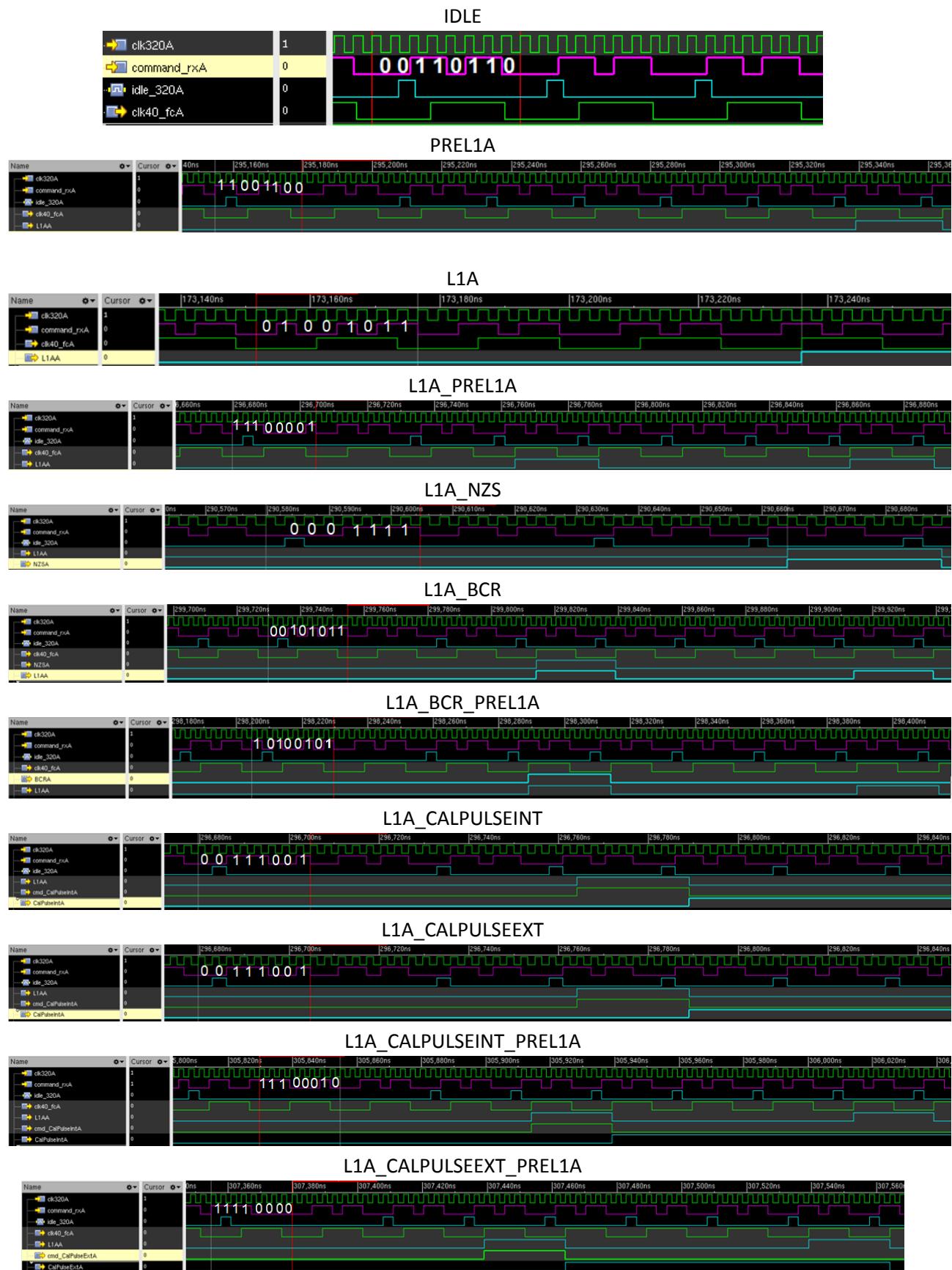
The Fast Command unit (FCU) is used to generate internal fast commands and to create 2 triplicated 40 MHz clocks (1 free running + 1 synchronised with the incoming bitstream). The following table describes each command and their latencies. The Latency is defined from the last bit received in the 320Mbps bitstream to the associated actions taken inside the ASIC. The Latency is given related to the synchronised 40 MHz clock (clk\_fc). The latency of the FCU, common to all fast commands, is 3 clock cycles ( $= 59.375\text{ns} \pm 3.125\text{ns}$ ) to decode a command since the arrival of the last serial bit of command.

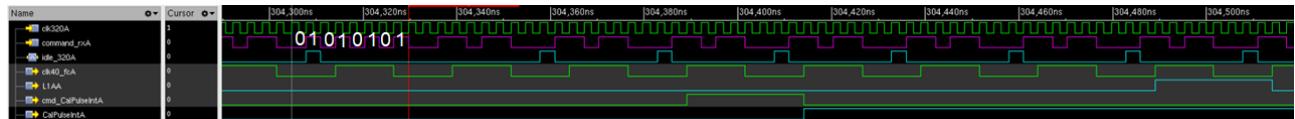
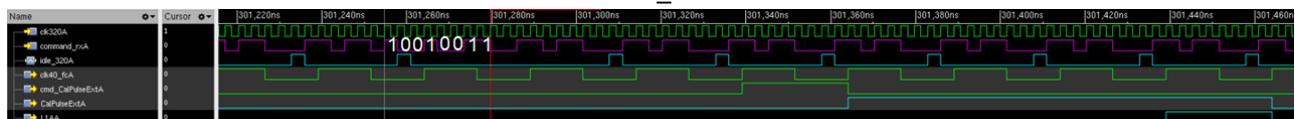
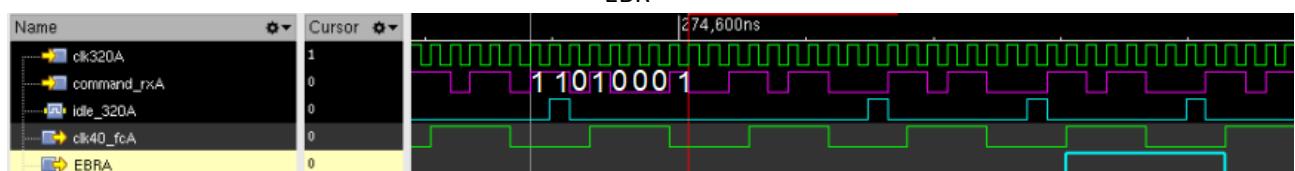
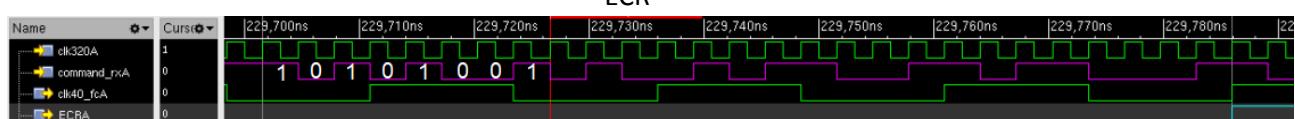
Command	Code	Description	Total latency (including FCU)
CMD_IDLE	00110110	Permit to lock the phase to generate the main clock 40MHz of digital.	
CMD_PREL1A	11001100	Send a delayed L1A (configured by the PreL1AOffset parameter). PREL1A is disabled if PreL1AOffset=0	1 additional clock cycle @40 per PreL1AOffset
CMD_L1A	01001011	The L1A reads an event in RAM1 (circular buffer). This event is copied in RAM2 and the readout.	- if EB empty, L1A to serial link latency (from last bit of fast cmd to first bit of Header): 10 clock cycle at 40MHz. - if EB not empty, from last bit of fast cmd to EB Writing: 4 clock cycle
CMD_L1A_PREL1A	11100001	Send a L1A then a delayed L1A.	L1A and PreL1A latency.
CMD_L1A_NZS	00001111	Same behaviour as L1A	L1A latency.
CMD_L1A_NZS_PREL1A	00101011	Send a NZS and L1A signal, then a delayed L1A.	L1A and PreL1A latency.
CMD_L1A_BCR	01110001	Send a L1A and reset the Bx counter.	L1A and BCR latency.
CMD_L1A_BCR_PREL1A	10100101	Send a L1A and reset the Bx counter, then a delayed L1A.	L1A, BCR and PreL1A latency.
CMD_L1A_CALPULSEINT	00111001	Send a L1A and a Calibration Pulse of 32 Bx (800ns)	CalPulseInt and L1A latency.
CMD_L1A_CALPULSEEXT	10000111	Send a L1A and a Calibration Pulse of 4 Bx (100ns)	CalPulseExt and L1A latency.
CMD_L1A_CALPULSEINT_PREL1A	11100010	Send a L1A and a Calibration Pulse of 32 Bx, then a delayed L1A.	CalPulseInt L1A and PreL1A latency.
CMD_L1A_CALPULSEEXT_PREL1A	11110000	Send a L1A and a Calibration Pulse of 4 Bx, then a delayed L1A.	CalPulseExt L1A and PreL1A latency.



<b>CMD_BCR</b>	00011101	Reset the Bx counter to its default value (parameter). Do not affect other counters.	3 clock cycle at 40MHz.
<b>CMD_BCR_PREL1A</b>	10100011	Reset the Bx counter, then a delayed L1A.	BCR and PreL1A latency.
<b>CMD_BCR_OCR</b>	10010101	Reset the Bx and Orbit counter.	BCR and OCR latency.
<b>CMD_CALPULSEINT</b>	00101101	Send a Calibration Pulse of 32 Bx.	1 additional clock cycle at 40MHz to latch the pulse.
<b>CMD_CALPULSEEXT</b>	01111000	Send a Calibration Pulse of 4 Bx.	1 additional clock cycle at 40MHz to latch the pulse.
<b>CMD_CALPULSEINT PREL1A</b>	01010101	Send a Calibration Pulse of 32 Bx and a delayed L1A.	CalPulseInt and PreL1A latency.
<b>CMD_CALPULSEEXT PREL1A</b>	10010011	Send a Calibration Pulse of 4 Bx and a delayed L1A.	CalPulseExt and PreL1A latency.
<b>CMD CHIPSYNC</b>	11010010	Like a ReSb but only affects the digital part. Does not reset analog part (PLL) and the FCU	4 clock cycles at 40 MHz to reset Startup FSM to idle state.
<b>CMD_EBR</b>	11010001	Reset the buffer pointer of RAM2 (FIFO). Also do an ECR	4 clock cycle at 40MHz.
<b>CMD_ECR</b>	10101001	Reset the Event counter.	5 clock cycles 40MHz.
<b>CMD_LINKRESETROCT</b>	10011001	Force links to send 400 IDLES.	5 clock cycles at 40MHz.
<b>CMD_LINKRESETROCD</b>	10011010	Force links to send 400 IDLES after the current sending packet ends.	The link is reset immediately if the readout is finished.
<b>CMD_ROC-Serializer-Reset</b>	10011100	Reset all Serializers, it does a bit alignment for all of them. The chip loses links during 14 clock cycles at 40MHz.	1 clock cycle at 40MHz to reset and 14 clock cycle at 40MHz to recover links.

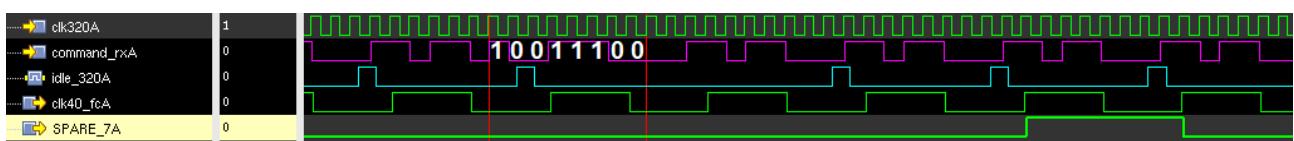
- We tested each Fast Commands in simulation:



**BCR****BCR\_PREL1A****BCR\_OCR****CALPULSEEXT****CALPULSEINT****CALPULSEINT\_PREL1A****CALPULSEEXT\_PREL1A****CHIPSYNC****EBR****ECR****LINK\_RESET\_T****LINK\_RESET\_D**

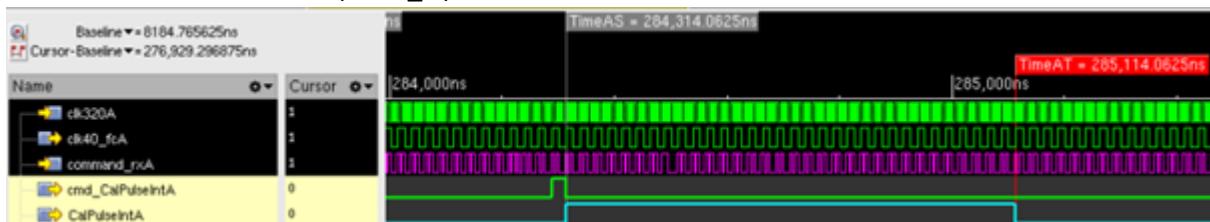


ROC-Serializer-Reset

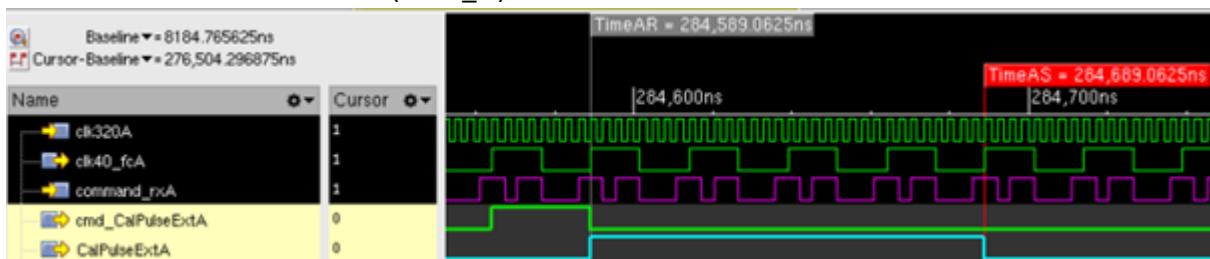


Some details:

- CalPulseInt: send STROBE pulse to the internal calibration DAC. The STROBE pulse length is 32 Bx at 40 MHz = 800ns. The STROBE pulse phase depends on the main digital clock from the Fast Commands module (clk40\_fc).



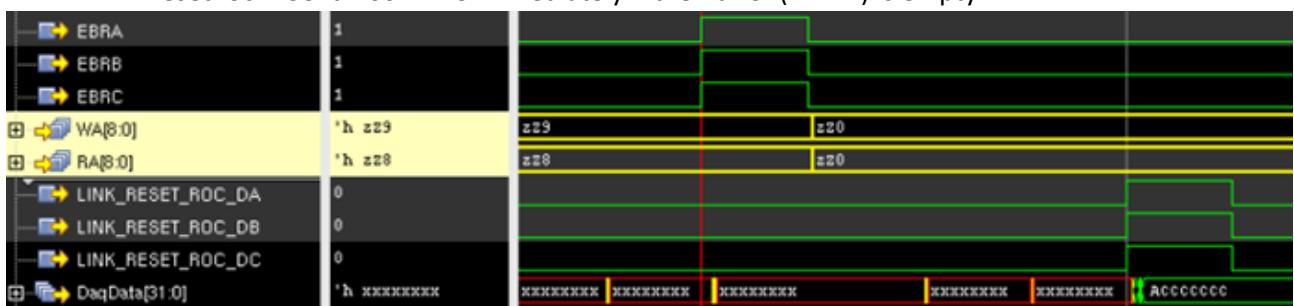
- CalPulseExt: send STROBE pulse to the external pin SiPM\_calibration. The STROBE pulse length is 4 Bx at 40 MHz = 100ns. The STROBE pulse phase depends on the main digital clock from the Fast Commands module (clk40\_fc).



The length and the phase of the STROBE pulse are set by slow-control.

Sending Fast Command CalPulse{Int/Ext} does not change the data content mode.

- LinkResetRocD: Send 400 IDLES immediately if the Buffer (RAM2) is empty:





- Two L1A sent spaced by offset: L1A at Bx N , second L1A at Bx N + Offset  
The number of Idle words between event data in the ROC output are like in the following table

N	N + Offset	Number of Idles between event data
0	0 + 40	1
0	0 + 41	1
0	0 + 42	3
0	0 + 43	3
0	0 + 44	4
0	0 + 45	5

#### 1.4.2. I2C

I2C protocol is used to access ASIC parameters.

A detailed document explaining I2C usage with HGCROC3 is available.

Main features are given in the table below:

I2C	Detail	Comments
Chip addressing	4 bits	MSB of I2C first byte
Number of Direct-access registers	8	Reg0 to Reg7
Direct-access register adressing	3 bits	LSB of I2C first byte
Indirect-access register adressing	16 bits	
“Burst” writing / reading	Yes	Accessible through R3 (Reg3)
Multi-Byte writing / reading*	Yes	*Multi-Byte reading in HGCROC3A /3B
I2C speed	100k - 1M	
General call	No	
Broadcast write	No	

As general call and broadcast write features are not implemented, we can access the asic only with its chipid.

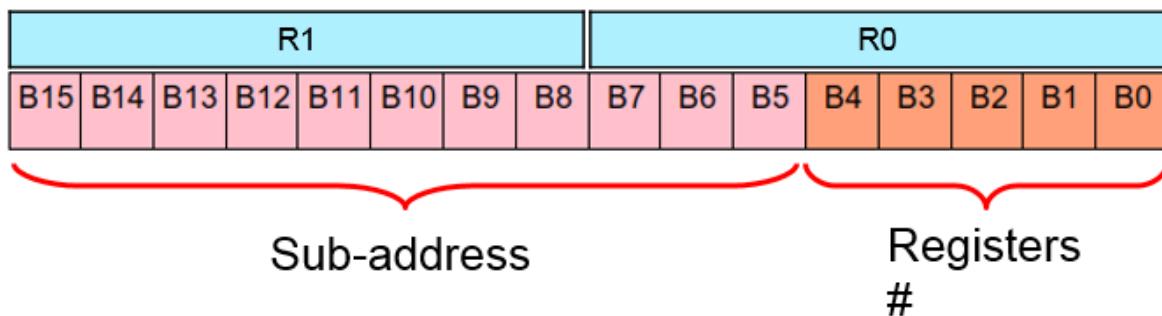
The I2C circuits of the chip has 8 internal registers whose the use is described in the table below:



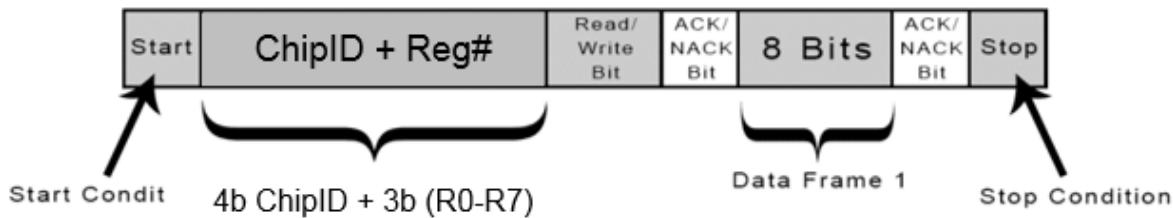
I2C @	Register	Comments
R0	ASIC parameter address (LSB)	Indirect @
R1	ASIC parameter address (MSB)	Indirect @
R2	Data	
R3	Data with auto @++	Increment indirect @ after each access
R4-R5-R6	Direct access SC-register	
R7	Status register (error, parity)	Read-only

To cope with the large number of parameters, extended addressing is used:

- 512 sub\_address can be addressed (B15, B14 not used and have to be set to 0)
  - Each sub\_address has max 32 configuration parameters
  - Extended addressing realised through 2 direct access registers: R0 and R1



The frame of the I<sup>2</sup>C protocol is always the same:



To write, set R/W bit to 0 and to read set R/W to 1.

For instance, to set a specific 8b word of the chip, the user has to write into the R0 register then R1 register to select the good parameters register address, and then write the data into the R2 register.

Start	<b>ChipID + R0</b>	Read/ Write Bit	ACK/ NACK Bit	<b>8 Bits</b>	ACK/ NACK Bit	Stop
Start	<b>ChipID + R1</b>	Read/ Write Bit	ACK/ NACK Bit	<b>8 Bits</b>	ACK/ NACK Bit	Stop
Start	<b>ChipID + R2</b>	Read/ Write Bit	ACK/ NACK Bit	<b>8 Bits</b>	ACK/ NACK Bit	Stop

The user can also write into consecutive parameters register addresses: rather to write the parameters into the R2 register, he has to write successively into the R3 register.

Start	<b>ChipID + R0</b>	Read/ Write Bit	ACK/ NACK Bit	<b>8 Bits</b>	ACK/ NACK Bit	Stop
Start	<b>ChipID + R1</b>	Read/ Write Bit	ACK/ NACK Bit	<b>8 Bits</b>	ACK/ NACK Bit	Stop
Start	<b>ChipID + R3</b>	Read/ Write Bit	ACK/ NACK Bit	<b>8 Bits</b>	ACK/ NACK Bit	Stop
Start	<b>ChipID + R3</b>	Read/ Write Bit	ACK/ NACK Bit	<b>8 Bits</b>	ACK/ NACK Bit	Stop
Start	<b>ChipID + R3</b>	Read/ Write Bit	ACK/ NACK Bit	<b>8 Bits</b>	ACK/ NACK Bit	Stop

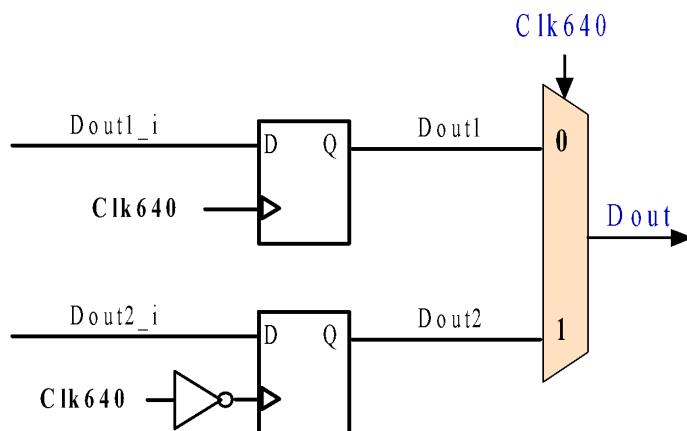
The I2C circuitry has been fully checked:

- digital verification of the I2C block
- analog verification of the I2C block
  - Write and read transactions from two tri-states inputs, with the bi-directionnal pads and the i2c block
- analog verification of the chip's full registers : write and read of dedicated pattern in chosen registers on both halves.

#### 1.4.3. Output E-links

The output differential links are composed of a serializer and a driver compatible with the LpGBT protocol (CLPS). The serializer converts parallel 32 bits words at 40 MHz to a serial train of bits send out at 1280 MHz.

New serializers without ReSyncLoad wrt HGCROCv2. ReSyncLoad will be removed and the SerDes will be autonomous. The internal logic will be partially triplicated.



In the table below, the electrical specifications of the driver are given.

Specification description	Value
Vcm (common voltage)	0,6 V
Vdiff (differential voltage)	100 to 200 mV
Pre-emphasis current	0,5 to 4 mA
Termination load	100 Ω

The termination load resistor must be placed outside the chip.

The register 5 of the “TOP” I2C sub-address allows to configure the current and the pre-emphasis of the driver.

Bit	Name	Default	Description
0	EN1	“1”	Current value of the CLPS drivers (Data/Trigger)
1	EN2	“1”	
2	EN3	“0”	
3	EN-pE0	“0”	Current value of the CLPS pre-emphasis driver (Data/Trigger)
4	EN-pE1	“0”	
5	EN-pE2	“0”	
6	S0	“0”	Delay value of the CLPS pre-emphasis driver (Data/Trigger)
7	S1	“0”	

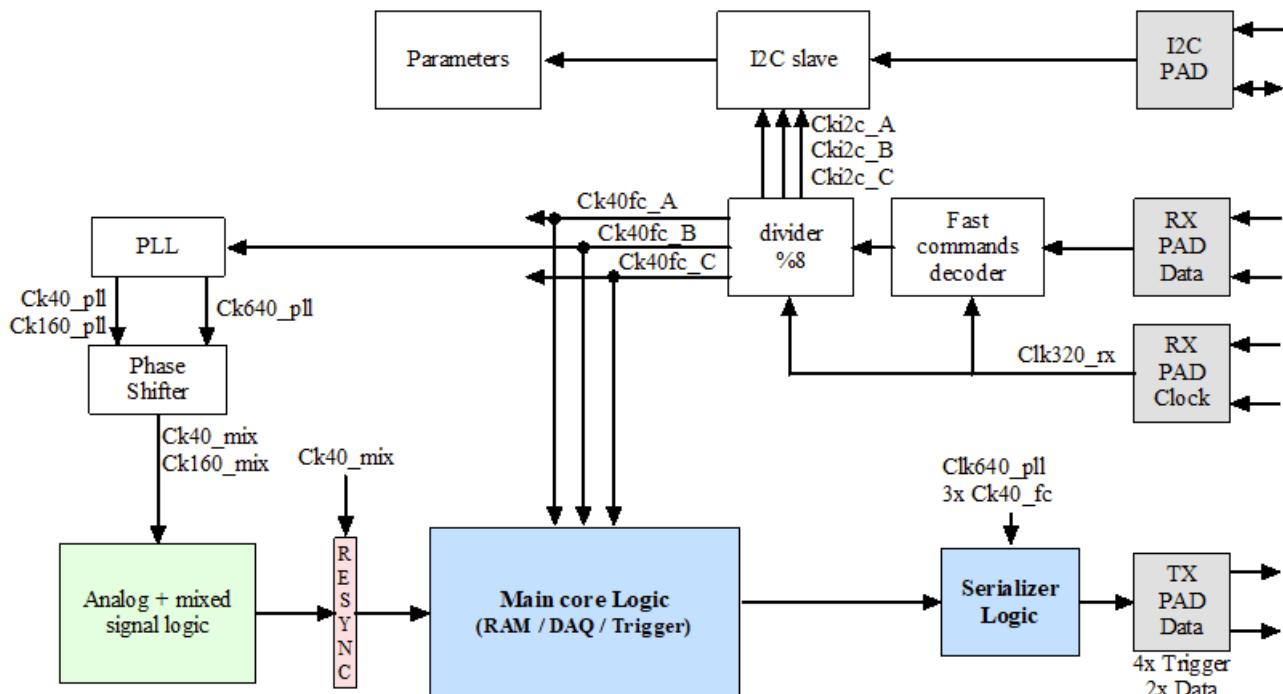
## 1.5. Ancillary blocks

### 1.5.1. PLL and clocks distribution

The main specifications of the PLL are described in the following table:

<b>PLL specifications</b>	
<b>Input frequency</b>	40 MHz (LHC bunch clock)
<b>Output frequencies</b>	1.28 GHz and 640, 320, 160 MHz
<b>Jitter cleaner</b>	Low jitter < 15 ps RMS (for an input jitter of 30 ps RMS)
<b>Power consumption</b>	< 2 mW
<b>Area</b>	Pitch 200 $\mu$ m
<b>Technology</b>	TSMC 130 nm
<b>Temperature</b>	-30 °C

The chip only receives the Fast-Command link made of the 320 MHz fast command and the 320 MHz clock. The FastCommand block decodes the fast command and the 40 MHz clock in phase with the LHC. This 40 MHz clock is used for the I2C block, all the FSMs, the counters, the two RAMs, the wr/rd pointers and is the PLL reference clock. Only hard resets (ReHb and RESb) allow to stop or restart this clock by resetting the FastCommand block.



The PLL takes as reference the 40 MHz clock provided by the FastCommand block. The PLL can only be reset by the hard resets (ReHb and ReSb). The PLL generates three clocks: clk\_40M\_pll, clk\_160M\_pll, clk\_640M\_pll.

- Clk\_40M\_pll: this clock comes in a Phase Shift block to achieve the ADCs clock
- Clk\_160M\_pll: this clock comes in a Phase Shift block to achieve the TDCs clock.
- Clk\_640M\_pll: this clock comes in the SerDes of the E-links.

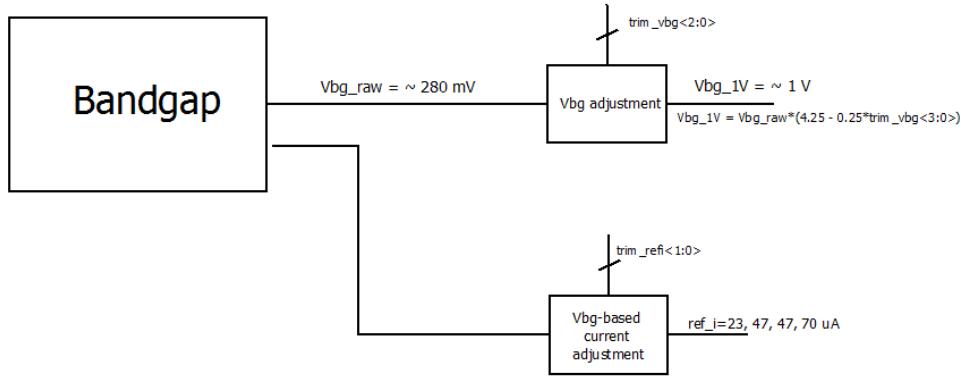
The shaper signal needs to be sampled to its maximum in order to optimize the signal-to-noise ratio; to do that the phase of the 40MHz sampling clock needs to be adjustable. The phase is adjusted by steps of ~ 1.5 ns (640 MHz period).

Some parameters, described in the “TOP” sub-address section, allow configuring the PLL.

### 1.5.2. Bandgap and voltage references

See the document [datasheet\\_BGP\\_130nm.pdf](#) for detailed information about the bandgap.

Typically, the band gap provides an output voltage of around 280 mV. We need to multiply this value to get a usable voltage reference around 1 V. From the bandgap voltage, a Vbg-based current is provided as well.



The Vbg\_1V is used to generate the upper reference voltage of the Calibration 11b-DAC, the reference voltage of the ADCs, ref. voltage for the TDCs, bias and to keep the TOT insensitive to the power supply variations. The ref\_i reference current is used to generate the offset and the steps of the 10b-DACs.

Four global 10 bits DACs provide voltage references for the analog part: the two discriminator thresholds, the inverted and non-inverted references for the shaper. As the circuit is symmetrical, there are two bandgaps and two sets of voltage references with DACs for the two right and left sub-parts.

The voltage references are fabricated from the bandgap (ref\_i), the 10b DAC and the preamplifier output of a common mode channel (channels CM<1> and CM<3>) so that the chip is not sensitive to the temperature (see Design Review report for more detailed information).

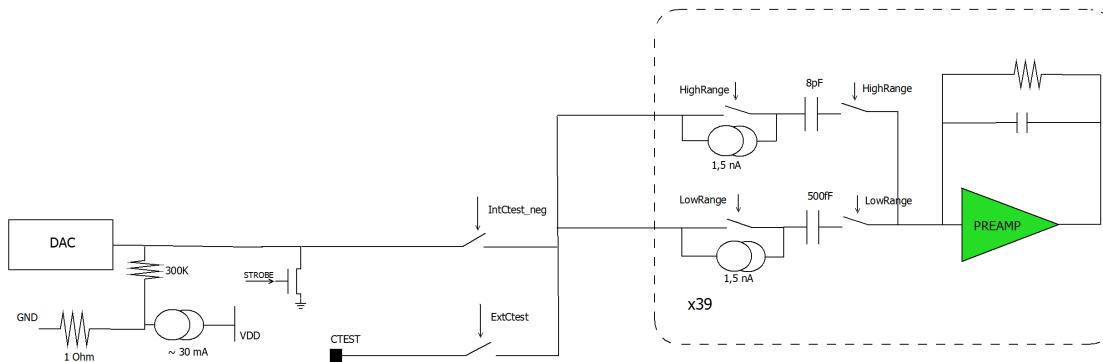
Note that since the reference voltages are made from the preamplifier output of CM<1> and CM<3>, these two channels are requested to be ON (the channel\_off parameters cannot be set to 1 for instance for these channels).

### 1.5.3. Calibration circuit

The following scheme shows the circuitry of the HGCROC-v3 calibration DAC. It is now a 12-bits DAC, the offset caused by the current flowing through the bandgap and the reference voltage buffers is removed by connecting the calibration DAC ground to the preamplifier grounding. Moreover, another source of offset is the gate leakage current of the switches (HighRange and LowRange) into each channel. Measurements on HGCROC-v2 show  $7\text{nA} + 1.5\text{nA} * N_{\text{CH\_ON}}$ , with  $N_{\text{CH\_ON}}$  the number of enabled injection capacitances. This would introduce an offset of 1.2 fC for  $N = 1$ , 10fC for  $N=40$ , it is the reason why we also reduce the size of the LowRange switches (by a factor 5) in order to hopefully keep the offset lower than 1fC for  $N=1$ , 2fC for  $N=40$ .

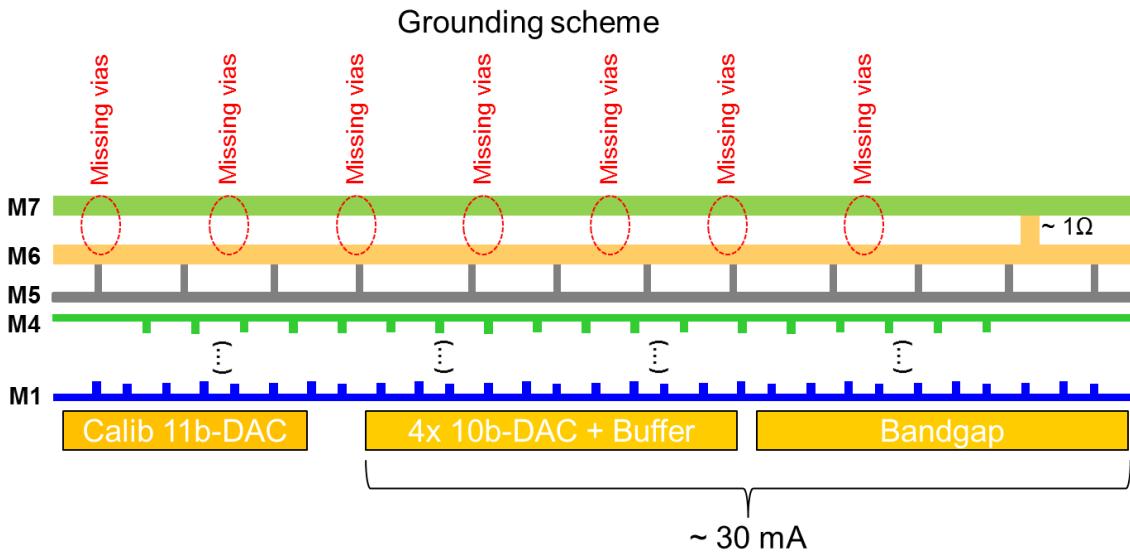
Assuming the 12b-DAC provides a voltage up to around 1 V (the bandgap value), the user may set 0.5pF or/and 8pF injection capacitances in the chosen channels, to study the 0 – 0.5 pC range, the 0 – 8 pC range or, by setting the both capacitances, the 0 – 8.5 pC range. To correctly calibrate the ADC/TOT behaviour, it is important to have an overlap between the available charge ranges.

The ctest node comes from an external pad and allows the user both to use an external pulser rather than the calibration DAC and also to calibrate the calibration 12b-DAC.



In HGCROC-v2, the 11-bit DAC had a ~30 mV offset when setting 0. The offset is relatively stable with temperature and irradiation. It was traced to a  $1 \Omega$  resistor in gnd\_dac in which  $\sim 30$  mA current is flowing to feed all the reference buffers. This resistance is lowered by M6-M7 connection and the calibration DAC is referred to gnd\_pa rather than gnd\_dac.

The following figure shows the HGCROC-V2 grounding scheme of the calibration DAC.



Finally, modifications to remove the calib\_dac offset

- The calibration DAC is 12-bit R2R instead of 11 bits.
- The ground is connected to gnd\_pa instead of gnd\_dac.
- The strobe switch is connected to gnd\_pa instead of vss.
- The output switch is removed to reduce gate leakage.
- The switch to the in\_ctest pad is reduced as it is used only for monitoring; Ctest goes now to the probe\_DC\_V through an analog multiplexer together with the other DACs.
- All the LowRange switches are reduced by a factor 5 to reduce the gate leakage and so the offset.

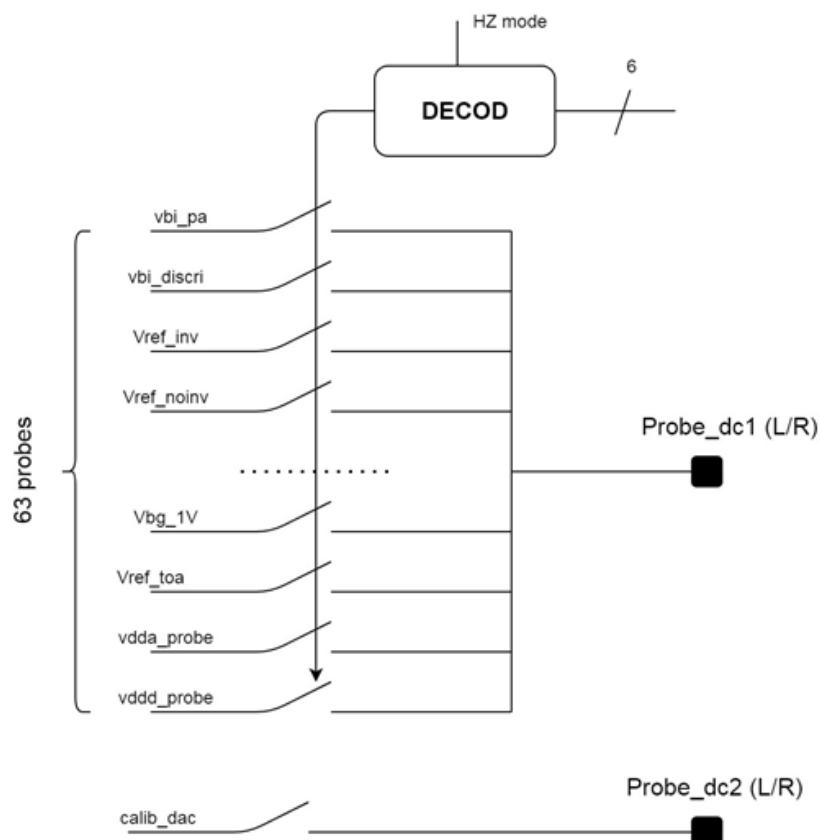
#### 1.5.3.1. Strobe pulse length

The calib fast command lasts 1 clock cycle but the strobe pulse lasts more than 10-12 clock cycles to not inject the opposite charge in BX+1. A counter (6 or 7 bits at 320MHz) is implemented to achieve this. The phase of the strobe pulse w.r.t. 40MHz clock will be adjustable by a delay of 1.56 ns (640 MHz).

A CalExt fast command sends a strobe to a pin.

A Callnt fast command sends a strobe to the internal CalibDAC.

#### 1.5.4. Monitoring



The internal DACs outputs (calibration, reference voltages and Current DAC) are multiplexed as described in the figure above. The signals will be sent to the ADC of the SCA chip. All the monitored points are in the 1V2 range.

On probe\_dc1, there are 64 available monitoring points.

If cmd\_probe\_dc1<0> is set, the next 32 voltages are accessible:



address	probe point	nominal value	Comment
0	vbi_pa	0,42	tunable over 4bits (dac_vbi_pa)
1	vbm_pa	0,6	
2	vbm2_pa	0,4	
3	vbm3_pa	0,13	
4	vbo_pa	0,4	tunable over 3bits (dac_vbo_pa)
5	vb_inputdac	0,3	tunable range over 3bits (indac)
6	vbi_discri_tot	0,8	
7	vbm_discri_tot	0,8	
8	vbo_discri_tot	0,16	
9	vbi_discri_toa	0,34	
10	vcasc_discri_toa	0,97	
11	vbm1_discri_toa	0,8	
12	vbm2_discri_toa	0,8	
13	vbo_discri_toa	0,44	
14	EXT_REF_TDC	0,1	
15	probe_VrefCf	0,43	equal to the TOA disciri input DC
16	vcn	0,65	
17	VD_FTDC_P_EXT	0,07	
18	VD_CTDC_P_EXT	0,07	
19	probe_VrefPa	0,174	equal to the preamp output (DC)
20	vcp	0,65	
21	VD_FTDC_N_EXT	0,19	
22	VD_CTDC_N_EXT	0,19	
23	vb_hyst_tot	0,46	
24	vbi_itot_neg	0,65	TOT gain, tunable over 6bits (dac_itot)
25	vbi_itot_pos	0,002	TOT gain, tunable over 6bits (dac_itot)
26	vbiN_sk	0,3	tunable over 2bits (ibi_sk)
27	vbiP_sk	0,81	tunable over 2bits (ibi_sk)
28	vbFCN_sk	0,31	tunable over 6bits (dac_sk)
29	vbFCP_sk	0,81	tunable over 6bits (dac_sk)
30	vbiN_noinv	0,3	tunable over 2bits (ibi_noinv)
31	vbiP_noinv	0,77	tunable over 2bits (ibi_noinv)

If cmd\_probe\_dc1<1> is set, the next 32 voltages are accessible:



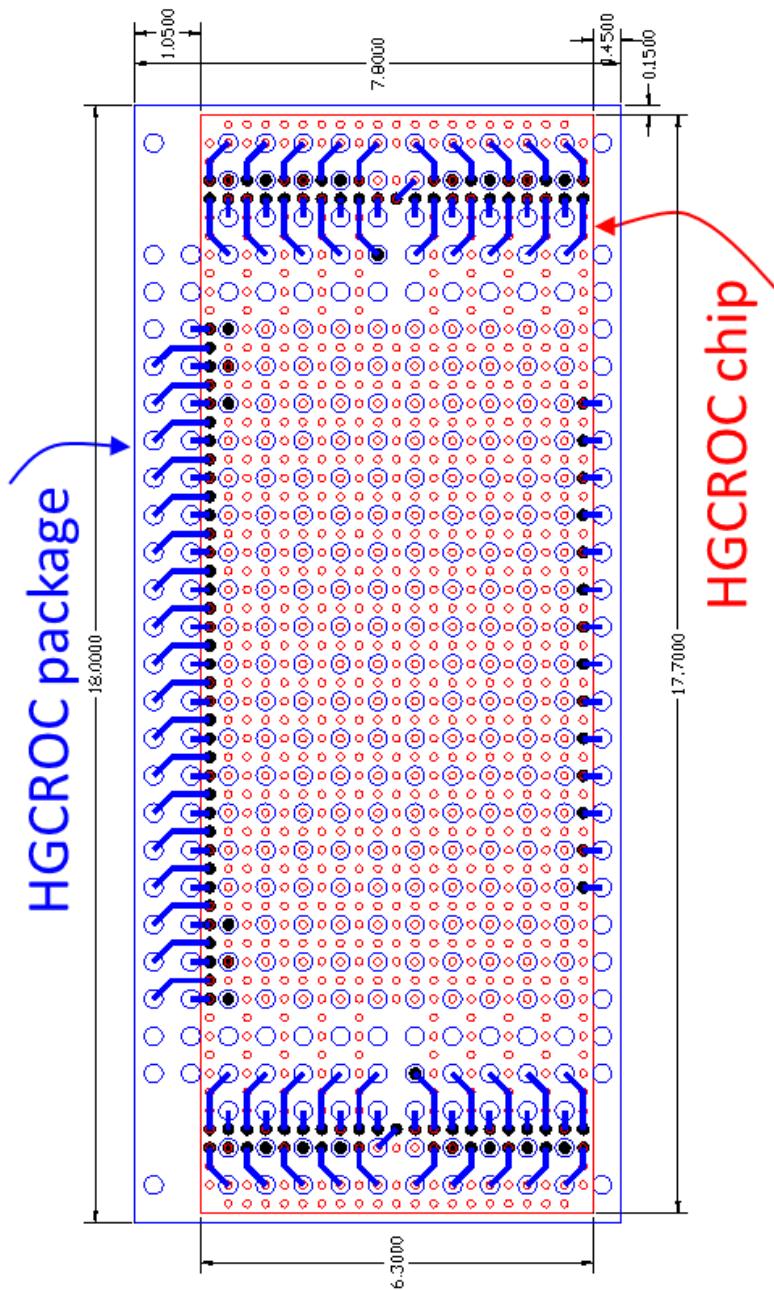
address	probe point	nominal value	Comment
0	vbFCN_noinv	0,3	tunable over 6bits (dac_noinv)
1	vbFCP_noinv	0,81	tunable over 6bits (dac_noinv)
2	vbiN_inv	0,3	tunable over 2bits (ibi_inv)
3	vbiP_inv	0,81	tunable over 2bits (ibi_inv)
4	vbFCN_inv	0,31	tunable over 6bits (dac_inv)
5	vbFCP_inv	0,81	tunable over 6bits (dac_inv)
6	vbiN_noinv_buf	0,33	tunable over 2bits (ibi_noinv_buf)
7	vbiP_noinv_buf	0,77	tunable over 2bits (ibi_noinv_buf)
8	vbFCN_noinv_buf	0,37	tunable over 6bits (dac_noinv_buf)
9	vbFCP_noinv_buf	0,72	tunable over 6bits (dac_noinv_buf)
10	vbiN_inv_buf	0,33	tunable over 2bits (ibi_inv_buf)
11	vbFCP_inv_buf	0,72	tunable over 6bits (dac_inv_buf)
12	vbiP_inv_buf	0,77	tunable over 2bits (ibi_inv_buf)
13	vbFCN_inv_buf	0,37	tunable over 6bits (dac_inv_buf)
14	vb_5bdac_out_inv	0,68	tunable range over 3bits (idac_inv)
15	vb_5bdac_tot	0,72	tunable range over 3bits (idac_tot)
16	vb_5bdac_toa	0,72	tunable range over 3bits (idac_tot)
17	vcm_Op6_inv	0,58	
18	vcm_Op6_noinv	0,58	
19	vrefp_adc	1,15	
20	vcm_adc	0,56	
21	Vref_sk	0,173	
22	Vref_noinv	0,237	tunable over 10bits (Vref_noinv)
23	Vref_inv	0,463	tunable over 10bits (Vref_inv)
24	Vref_tot	0,392	tunable over 10bits (Vref_tot)
25	Vref_toa	0,49	tunable over 10bits (Vref_toa)
26	vbg_1V	1,1	tunable over 3bits (vbg_1V)
27	probe_center	0,5	VOUT_INIT_EXT / EXT_REF_PLL(half 1/0)
28	ibi_ref_adc	1,01	tunable over 2bits (b_ref_adc)
29	ibo_ref_adc	0,79	
30	probe_vddd	0,58	
31	probe_vdda	0,58	

If cmd\_probe\_dc1<1:0> are not set, the probe\_dc1 is in high impedance so that several probes can be tied together on the hexaboard.

If cmd\_probe\_dc2 is set, the calibration dac output is sent to probe\_dc2.

## 2. Packaging, I/Os and powering scheme

The chip pinout has been defined by the hexaboard constraints [Tommaso et al.] and the BGA package that will be used to house the chips, as shown in figure below.



The two figures below show the left view and the right view of the die. Can be noticed the common column 25 in both figures.



The figure below shows the top view of the BGA map as can be seen on the board.



TOP VIEW																															
A	I <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>	P <sub>IN</sub>			
B	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	
C	GND	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	IN<2>	
D	Vref <sub>L</sub> ,S,R	IN<6>	IN<7>	IN<6>	IN<7>	IN<6>	IN<7>	IN<6>	IN<7>	IN<6>	IN<7>	IN<6>	IN<7>	IN<6>	IN<7>	IN<6>	IN<7>	IN<6>	IN<7>	IN<6>	IN<7>	IN<6>	IN<7>	IN<6>	IN<7>	IN<6>	IN<7>	IN<6>	IN<7>	IN<6>	
E	Vref <sub>S</sub> ,S,R	IN<8>	IN<9>	IN<8>	IN<9>	IN<8>	IN<9>	IN<8>	IN<9>	IN<8>	IN<9>	IN<8>	IN<9>	IN<8>	IN<9>	IN<8>	IN<9>	IN<8>	IN<9>	IN<8>	IN<9>	IN<8>	IN<9>	IN<8>	IN<9>	IN<8>	IN<9>	IN<8>	IN<9>	IN<8>	
F	Vref <sub>L</sub> ,IN,L,R	IN<10>	IN<11>	IN<10>	IN<11>	IN<10>	IN<11>	IN<10>	IN<11>	IN<10>	IN<11>	IN<10>	IN<11>	IN<10>	IN<11>	IN<10>	IN<11>	IN<10>	IN<11>	IN<10>	IN<11>	IN<10>	IN<11>	IN<10>	IN<11>	IN<10>	IN<11>	IN<10>	IN<11>	IN<10>	
G	Vref <sub>L</sub> ,IN,R	IN<12>	IN<13>	IN<12>	IN<13>	IN<12>	IN<13>	IN<12>	IN<13>	IN<12>	IN<13>	IN<12>	IN<13>	IN<12>	IN<13>	IN<12>	IN<13>	IN<12>	IN<13>	IN<12>	IN<13>	IN<12>	IN<13>	IN<12>	IN<13>	IN<12>	IN<13>	IN<12>	IN<13>	IN<12>	
H	Vref <sub>L</sub> ,DG,R	NC	AVD <sub>0</sub> ,L	AVD <sub>0</sub> ,R	AVD <sub>1</sub> ,L	AVD <sub>1</sub> ,R	AVD <sub>2</sub> ,L	AVD <sub>2</sub> ,R	AVD <sub>3</sub> ,L	AVD <sub>3</sub> ,R	AVD <sub>4</sub> ,L	AVD <sub>4</sub> ,R	AVD <sub>5</sub> ,L	AVD <sub>5</sub> ,R	AVD <sub>6</sub> ,L	AVD <sub>6</sub> ,R	AVD <sub>7</sub> ,L	AVD <sub>7</sub> ,R	AVD <sub>8</sub> ,L	AVD <sub>8</sub> ,R	AVD <sub>9</sub> ,L	AVD <sub>9</sub> ,R	AVD <sub>10</sub> ,L	AVD <sub>10</sub> ,R	AVD <sub>11</sub> ,L	AVD <sub>11</sub> ,R	AVD <sub>12</sub> ,L	AVD <sub>12</sub> ,R	AVD <sub>13</sub> ,L	AVD <sub>13</sub> ,R	
I	Vref <sub>L</sub> ,DG,U,R	IN<14>	IN<15>	IN<14>	IN<15>	IN<14>	IN<15>	IN<14>	IN<15>	IN<14>	IN<15>	IN<14>	IN<15>	IN<14>	IN<15>	IN<14>	IN<15>	IN<14>	IN<15>	IN<14>	IN<15>	IN<14>	IN<15>	IN<14>	IN<15>	IN<14>	IN<15>	IN<14>	IN<15>	IN<14>	IN<15>
K	Prob <sub>L</sub> ,P <sub>U</sub> ,R	IN<16>	IN<17>	IN<16>	IN<17>	IN<16>	IN<17>	IN<16>	IN<17>	IN<16>	IN<17>	IN<16>	IN<17>	IN<16>	IN<17>	IN<16>	IN<17>	IN<16>	IN<17>	IN<16>	IN<17>	IN<16>	IN<17>	IN<16>	IN<17>	IN<16>	IN<17>	IN<16>	IN<17>	IN<16>	IN<17>
L	Vref <sub>L</sub> ,CMD <sub>U</sub> ,R	IN<18>	IN<19>	IN<18>	IN<19>	IN<18>	IN<19>	IN<18>	IN<19>	IN<18>	IN<19>	IN<18>	IN<19>	IN<18>	IN<19>	IN<18>	IN<19>	IN<18>	IN<19>	IN<18>	IN<19>	IN<18>	IN<19>	IN<18>	IN<19>	IN<18>	IN<19>	IN<18>	IN<19>	IN<18>	IN<19>
M	GND	IN<20>	IN<21>	IN<20>	IN<21>	IN<20>	IN<21>	IN<20>	IN<21>	IN<20>	IN<21>	IN<20>	IN<21>	IN<20>	IN<21>	IN<20>	IN<21>	IN<20>	IN<21>	IN<20>	IN<21>	IN<20>	IN<21>	IN<20>	IN<21>	IN<20>	IN<21>	IN<20>	IN<21>	IN<20>	IN<21>
N	GND	GND	IN<22>	IN<23>	IN<22>	IN<23>	IN<22>	IN<23>	IN<22>	IN<23>	IN<22>	IN<23>																			
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27					

## 2.1. I/Os list

Names	Comment	Type	# ports	# signals	Level
Clk e-link	Fast commands @ 320M	Input	1	2	CLPS
FastCmd e-link	Fast commands @ 320M	Input	1	2	CLPS
Trigger e-link	Trigger @ 1,28G	Output	4	8	CLPS
DAQ e-link	DAQ @ 1,28G	Output	2	4	CLPS



I2C - SDA	I2C data	Bidir	1	1	CMOS
I2C - SCL	I2C clock	Input	1	1	CMOS
I2C - address	I2C address bonded on the board	Input	1	4	CMOS
Power-ON disable	Disable the POR	Input	1	1	CMOS
ReHb	Global reset, global to the hexaboard	Input	1	1	CMOS
ReSb	Global soft reset	Input	1	1	CMOS
Error B **	Error Flag	Output	1	1	Open Drain
Efuse			1	1	
SiPM calibration***	Calibration enable (to strobe LEDs)	Output	1	1	CMOS
		<b>Total</b>		<b>28</b>	

(\*\*) Error B. Error signal flagging a SEU event in the I2C register parameters. The implementation has to be defined.

(\*\*\*) SiPM calibration. Only for the SiPM version of the chip.

The PLL locked signal will be accessible through a direct addressing I2C register.

Name	Comment	Type	# ports	# signals
Probe_DC_V	To monitor/calibrate vDACs, bias	Output	-	1
Probe_DC_I	To monitor/calibrate iDACs	Output	-	1
<b>Total</b>				<b>2</b>

## 2.2. Pin List

	Pin Name	Pin type		Comments	Default value
P O W ER S/ GR O	P2V5	POWER		analog pad ring power supply	1,2V
	AVDD_0	POWER		analog power supply	1,2V
	AVDD_1	POWER		analog power supply	1,2V
	GND	GROUND		analog ground	0V
	DVDD	POWER		digital power supply	1,2V
	VSS	GROUND		digital ground	0V
	VDD_PLL	POWER		PLL power supply	1,2V
	GND_PLL	GROUND		PLL ground	0V
	VDD_SC	POWER		Slow Control power supply	1,2V



<b>UNDS</b>					
	VHI_R	ANALOG		By default keep free ; for monitoring.	1V
	VHI_L	ANALOG		By default keep free ; for monitoring.	1V
	Vref_SK_R	ANALOG	Input/Output	Reference voltage of the Sallen-Key amplifier, value close to the preamp input voltage. By default it must be kept free. Can be monitored/checkered.	200mV
	Vref_noinv_R	ANALOG	Input/Output	Reference voltage of the non-inverted shaper. Value given by an internal 10b_DAC. Can be checked/monitored externally. By default it must be kept free.	
	Vref_inv_R	ANALOG	Input/Output	Reference voltage of the inverted shaper. Value given by an internal 10b_DAC. Can be checked/monitored externally. By default must be kept free.	
	Vref_Toa_R	ANALOG	Input/Output	Reference voltage of the TOA threshold. Value given by an internal 10b_DAC. Can be checked/monitored externally. By default it must be kept free.	
	Vref_Tot_R	ANALOG	Input/Output	Reference voltage of the TOT threshold. Value given by an internal 10b_DAC. Can be checked/monitored externally. By default must be kept free.	
	VBG_1V_R	ANALOG	Input/Output	BANDGAP voltage. Value can be tuned by slow-control. Default value = 1V	1V
	Probe_PA_R	ANALOG		preamplifier analog probe. Channel-wise	
<b>Half part (ch. 36 to 71)</b>	IN_Ctest_R	ANALOG	Input	By default, external pin test, but possibility to connect the calibration DAC output to this pin by slow-control.	
	VrefP_ADC_R	ANALOG	Input/Output	Positive reference voltage of the ADC.	
	VrefN_ADC_R	ANALOG	Input/Output	Negative reference voltage of the ADC.	
	Vcm_ADC_R	ANALOG	Input/Output	Common mode reference voltage of the ADC.	
	Probe_DC1_R	ANALOG	Input/Output	DC analog probe (bias, ref.)	



Probe_DC2_R	ANALOG	Input/Output	DC analog probe (bias, ref.)	
VGNR	ANALOG	Input/Output	VGN monitoring	
Probe_Toa_R	CMOS	Output	TOA discri output probe. Channel-wise	
Probe_Tot_R	CMOS	Output	TOT discri output probe. Channel-wise	
Trig1_R	CMOS	Input	External trigger 1	
Trig2_R	CMOS	Input	External trigger 2	
ADCP_R	ANALOG	Input/Output	Analog non-inverted probe; positive ADC input	100mV
ADCN_R	ANALOG	Input/Output	Analog inverted probe; negative ADC input	1,1V
IN<35:0>	ANALOG	Channel input		200mV
CALIB<0>	ANALOG	Channel input		200mV
CM<1:0>	ANALOG	Channel input		200mV

Co m m on pi ns	I2C_rstb	CMOS	Input	I2C reset; ACTIVE LOW	0V
	SDA	I2C signal		I2C data	
	SCL	I2C clk	Input	I2C clock	
	ErrorB	Open drain	Output	Open collector; external resistor must be added. OR of all the slow-control cells' error signals	1,2V
	ADD<3:0>	I2C chip adress	Input	I2C chip address	
	ReSync_ext	CMOS	Input	External ReSync selectable by Slow Control	
	L1_Ext	CMOS	Input	External L1 signal selectable by slow control	
	Strobe_Ext	CMOS	Input	External calibration signal selectable by slow control	
	SiPM_Calib	CMOS	Output	Calibration signal for SiPM	
	ReSyncLoad	CMOS	Input	Serializers synchronization signal	0V
	Sel_CK_ext	CMOS	Input	External 40MHz clock selection (debug purpose); ACTIVE HIGH.	0V
	Rstb	CMOS	Input	General reset; ACTIVE LOW	
	EFUSE	ANALOG		pin connected to an internal resistor. HGCROC2 => 100Ω. HGCROC2A => 1KΩ. H2GCROC2 => 10KΩ. H2GCROC2A => 100KΩ.	
	PLL_lock	CMOS	Output		
	Clk320_p	CLPS	Input	Fast command 320 MHz clock	
	Clk320_n	CLPS	Input		



Fcmd_p	CLPS	Input	Fast command data	
Fcmd_n	CLPS	Input		
Clk40_p	CLPS	Input	40MHz clock for debug purpose	
Clk40_n	CLPS	Input		
PLL_p	CLPS	Output	PLL output probe	
PLL_n	CLPS	Output		
Daq0_p	CLPS	Output	Data 0 link	
Daq0_n	CLPS	Output		
Daq1_p	CLPS	Output	Data 1 link	
Daq1_n	CLPS	Output		
Trig0_p	CLPS	Output	Trigger 0 link	
Trig0_n	CLPS	Output		
Trig1_p	CLPS	Output	Trigger 1 link	
Trig1_n	CLPS	Output		
Trig2_p	CLPS	Output	Trigger 2 link	
Trig2_n	CLPS	Output		
Trig3_p	CLPS	Output	Trigger 3 link	
Trig3_n	CLPS	Output		
Spare<3:1>	NC			

<b>Ha lf pa rt (c h. 0 to 35 )</b>	Vref_SK_L	ANALOG	Input/Output	Reference voltage of the Sallen-Key amplifier, value close to the preamp input voltage. By default it must be kept free. Can be monitored/checked.	200mV
	Vref_noinv_L	ANALOG	Input/Output	Reference voltage of the non-inverted shaper. Value given by an internal 10b_DAC. Can be checked/monitored externally. By default it must be kept free.	
	Vref_inv_L	ANALOG	Input/Output	Reference voltage of the inverted shaper. Value given by an internal 10b_DAC. Can be checked/monitored externally. By default it must be kept free.	
	Vref_Toa_L	ANALOG	Input/Output	Reference voltage of the TOA threshold. Value given by an internal 10b_DAC. Can be checked/monitored externally. By default it must be kept free.	
	Vref_Tot_L	ANALOG	Input/Output	Reference voltage of the TOT threshold. Value given by an internal 10b_DAC. Can be	



			checked/monitored externally. By default it must be kept free.	
VBG_1V_L	ANALOG	Input/Output	BANDGAP voltage. Value can be tuned by slow-control. Default value = 1V	1V
Probe_PA_L	ANALOG		preamplifier analog probe. Channel-wise	
IN_Ctest_L	ANALOG	Input	By default, external pin test, but possibility to connect the calibration DAC output to this pin by slow-control.	
VrefP_ADC_L	ANALOG	Input/Output	Positive reference voltage of the ADC.	
VrefN_ADC_L	ANALOG	Input/Output	Negative reference voltage of the ADC.	
Vcm_ADC_L	ANALOG	Input/Output	Common mode reference voltage of the ADC.	
Probe_DC1_L	ANALOG	Input/Output	DC analog probe (bias, ref.)	
Probe_DC2_L	ANALOG	Input/Output	DC analog probe (bias, ref.)	
VGNL	ANALOG	Input/Output	VGN monitoring	
Probe_Toa_L	CMOS	Output	TOA discri output probe. Channel-wise	
Probe_Tot_L	CMOS	Output	TOT discri output probe. Channel-wise	
Trig1_L	CMOS	Input	External trigger 1	
Trig2_L	CMOS	Input	External trigger 2	
ADCP_L	ANALOG	Input/Output	Analog non-inverted probe; positive ADC input	100mV
ADCN_L	ANALOG	Input/Output	Analog inverted probe; negative ADC input	1,1V
IN<71:36>	ANALOG	Channel input		200mV
CALIB<1>	ANALOG	Channel input		200mV
CM<3:2>	ANALOG	Channel input		200mV

### 3. ASIC parameters

As described in section [1.4.2 I2C](#), the I2C circuit has 8 internal registers; the four first are dedicated to write/read the slow-control parameters. Registers 4, 5 and 6 are direct access registers and the register 7 is a status register in read-only mode.

In the tables below, the content of the direct access registers is described.



Bit	Name	Default	Description
0	AutoReload	"0"	Allows to rewrite the value after a SEU
1	EdgeSel	"0"	Selection of the clock edge of the 320MHz clock
2	NA	"0"	
3	NA	"0"	
4	NA	"0"	
5	NA	"0"	
6	NA	"0"	
7	NA	"0"	

I2C Register R5

Bit	Name	Default	Description
0	NA	"0"	
1	NA	"0"	
2	NA	"0"	
3	NA	"0"	
4	NA	"0"	
5	NA	"0"	
6	NA	"0"	
7	NA	"0"	

I2C Register R6

Bit	Name	Default	Description
0	NA	"0"	
1	NA	"0"	
2	NA	"0"	
3	NA	"0"	
4	NA	"0"	
5	NA	"0"	
6	NA	"0"	
7	NA	"0"	

I2C Register R7 (Status , read only)

Bit	Name	Default	Description
0	Error		OR of all the slow-control cells' error signals
1	Parity		Parity of all the slow-control cells' values
2	PLL_Lock		PLL lock flag. "1" pll is locked
3	NA		
4	NA		
5	NA		
6	NA		
7	NA		



### 3.1. I2C Addressing

The four first I2C registers, R0-3, allow to define the address and the data to write or read into the internal 8 bits registers distributed into the chip sub-parts. R0 and R1 define the address of internal 8 bits registers. R2 defines the data to write into the chosen internal 8 bits register. R3 allows the user to access successive register address.

The chip, from I2C protocol point of view, is divided into sub-blocks containing a maximum of 32 registers each. In consequence, the address in 16 bits (given in R0 and R1) is composed of two sub-address:

- The 11 MSB bits code the address of the sub-block
- The 5 LSB bits code the address of the register of the sub-block

The table below gives the address, the name and a short description of all the sub-blocks.

Sub-block name	Sub-block address	Description
CM_2	0	Registers described in “channel wise” I2C parameters table
CM_3	1	
Channel_36	2	
Channel_37	3	
Channel_38	4	
Channel_39	5	
Channel_40	6	
Channel_41	7	
Channel_42	8	
Channel_43	9	



Channel_44	10
Channel_45	11
Channel_46	12
Channel_47	13
Channel_48	14
Channel_49	15
Channel_50	16
Channel_51	17
Channel_52	18
Channel_53	19
Channel_54	20
Channel_55	21
Channel_56	22
Channel_57	23
Channel_58	24
Channel_59	25
Channel_60	26



Channel_61	27	
Channel_62	28	
Channel_63	29	
Channel_64	30	
Channel_65	31	
Channel_66	32	
Channel_67	33	
Channel_68	34	
Channel_69	35	
Channel_70	36	
Channel_71	37	
CALIB_1	38	
<b>No Register</b>		
Reference_Voltage_1	40	Registers described in “ <b>Reference Voltage</b> ” table
Global_Analog_1	41	Registers described in “ <b>Global analog</b> ” table
Master_TDC_1	42	Registers described in “ <b>Master TDC</b> ” table



Digital_Half_1	43	Registers described in “ <b>Digital half</b> ” table
HALFWISE_1	44	Registers described in “ <b>channel wise</b> ” I2C parameters table
Top	45	Registers described in “ <b>Top sub-block</b> ” table
CM_0	46	Registers described in “ <b>channel wise</b> ” I2C parameters table
CM_1	47	
Channel_0	48	
Channel_1	49	
Channel_2	50	
Channel_3	51	
Channel_4	52	
Channel_5	53	
Channel_6	54	
Channel_7	55	
Channel_8	56	
Channel_9	57	
Channel_10	58	



Channel_11	59	
Channel_12	60	
Channel_13	61	
Channel_14	62	
Channel_15	63	
Channel_16	64	
Channel_17	65	
Channel_18	66	
Channel_19	67	
Channel_20	68	
Channel_21	69	
Channel_22	70	
Channel_23	71	
Channel_24	72	
Channel_25	73	
Channel_26	74	
Channel_27	75	



Channel_28	76	
Channel_29	77	
Channel_30	78	
Channel_31	79	
Channel_32	80	
Channel_33	81	
Channel_34	82	
Channel_35	83	
CALIB_0	84	
<b>No Register</b>		
Reference_Voltage_0	86	Registers described in “ <b>Reference Voltage</b> ” table
Global_Analog_0	87	Registers described in “ <b>Global analog</b> ” table
Master_TDC_0	88	Registers described in “ <b>Master TDC</b> ” table
Digital_Half_0	89	Registers described in “ <b>Digital half</b> ” table
HALFWISE_0	90	Registers described in “ <b>channel wise</b> ” I2C parameters table

**CHANNEL-WISE TABLE**

Register # 0			
Bit	Name	Default	Description
0	Inputdac<0>	"0"	Leakage current Input-DAC Value
1	Inputdac<1>	"0"	
2	Inputdac<2>	"0"	
3	Inputdac<3>	"0"	
4	Inputdac<4>	"0"	
5	Inputdac<5>	"0"	
6	Inputdac<6>	"0"	
7	Inputdac<7>	"0"	

Register # 1			
Bit	Name	Default	Description
0	Mask_toa	"0"	
1	Sel_trig_toa	"0"	
2	Trim_toa<0>	"0"	Local 5b-DAC for TOA threshold tuning



<b>3</b>	Trim_toa<1>	"0"	
<b>4</b>	Trim_toa<2>	"0"	
<b>5</b>	Trim_toa<3>	"0"	
<b>6</b>	Trim_toa<4>	"0"	
<b>7</b>	Trim_toa<5>	"0"	

**Register # 2**

Bit	Name	Default	Description
<b>0</b>	NA	"0"	
<b>1</b>	NA	"0"	
<b>2</b>	Trim_tot<0>	"0"	
<b>3</b>	Trim_tot<1>	"0"	
<b>4</b>	Trim_tot<2>	"0"	
<b>5</b>	Trim_tot<3>	"0"	
<b>6</b>	Trim_tot<4>	"0"	
<b>7</b>	Trim_tot<5>	"0"	

**Register # 3**



Bit	Name	Default	Description
0	Probe_inv	"0"	
1	Probe_noinv	"0"	
2	Trim_inv<0>	"0"	
3	Trim_inv<1>	"0"	
4	Trim_inv<2>	"0"	
5	Trim_inv<3>	"0"	
6	Trim_inv<4>	"0"	
7	Trim_inv<5>	"0"	

---

**Register # 4**

Bit	Name	Default	Description
0	Probe_pa	"0"	Preamplifier output probe
1	LowRange	"0"	0.5pF injection cap
2	HighRange	"0"	8pF injection cap
3	Channel_off	"0"	"1" = preamplifier input tied to ground



4	Sel_trigger_tot	"0"	External trigger selection for TOT ("0" = Ext Trig1; "1" = Ext Trig2)
5	Mask_tot	"0"	TOT discri output mask ("1" = masked)
6	Probe_tot	"0"	TOT discri output probe
7	Probe_toa	"0"	TOA discri output probe

Register # 5			
Bit	Name	Default	Description
0	DAC_CAL_CTDC_TOT<0>	"0"	Tune the fine gain of the TOT CTDC: 5 bits DAC <0:4> ( $1k\Omega \times 32$ ) by the BIAS_CAL_DAC_P and the sign 1 bit <5>  VD_P CAL = VD_P + (sign <5> x <0:4> $\times 1k\Omega \times \text{BIAS\_CAL\_DAC\_P}$ )
1	DAC_CAL_CTDC_TOT<1>	"0"	
2	DAC_CAL_CTDC_TOT<2>	"0"	
3	DAC_CAL_CTDC_TOT<3>	"0"	
4	DAC_CAL_CTDC_TOT<4>	"0"	
5	DAC_CAL_CTDC_TOT<5>	"0"	
6	NA	"0"	
7	Mask_adc	"0"	"1" = ADC clock off

Register # 6			
--------------	--	--	--



Bit	Name	Default	Description
0	DAC_CAL_CTDC_TOA<0>	"0"	Tune the fine gain of the TOA CTDC: 5 bits DAC <0:4> ( $1\text{k}\Omega \times 32$ ) by the BIAS_CAL_DAC_P and the sign 1 bit <5>  VD_P CAL = VD_P + (sign <5> x <0:4> $\times 1\text{k}\Omega \times \text{BIAS\_CAL\_DAC\_P}$ )
1	DAC_CAL_CTDC_TOA<1>	"0"	
2	DAC_CAL_CTDC_TOA<2>	"0"	
3	DAC_CAL_CTDC_TOA<3>	"0"	
4	DAC_CAL_CTDC_TOA<4>	"0"	
5	DAC_CAL_CTDC_TOA<5>	"0"	
6	HZ_noinv	"0"	
7	HZ_inv	"0"	

**Register # 7**

Bit	Name	Default	Description
0	DAC_CAL_FTDC_TOT<0>	"0"	Tune the fine gain of the TOT FTDC: 5 bits DAC <0:4> ( $1\text{k}\Omega \times 32$ ) by the BIAS_CAL_DAC_P and the sign 1 bit <5>  VD_P CAL = VD_P + (sign <5> x <0:4> $\times 1\text{k}\Omega \times \text{BIAS\_CAL\_DAC\_P}$ )
1	DAC_CAL_FTDC_TOT<1>	"0"	
2	DAC_CAL_FTDC_TOT<2>	"0"	
3	DAC_CAL_FTDC_TOT<3>	"0"	
4	DAC_CAL_FTDC_TOT<4>	"0"	



5	DAC_CAL_FTDC_TOT<5>	"0"	
6	NA	"0"	
7	NA	"0"	

**Register # 8**

Bit	Name	Default	Description
0	DAC_CAL_FTDC_TOA<0>	"0"	Tune the fine gain of the TOA FTDC: 5 bits DAC <0:4> ( $1\text{k}\Omega \times 32$ ) by the BIAS_CAL_DAC_P and the sign 1 bit <5>
1	DAC_CAL_FTDC_TOA<1>	"0"	
2	DAC_CAL_FTDC_TOA<2>	"0"	$\text{VD}_P \text{ CAL} = \text{VD}_P + (\text{sign } <5> \times <0:4>$ $\times 1\text{k}\Omega \times \text{BIAS\_CAL\_DAC\_P})$
3	DAC_CAL_FTDC_TOA<3>	"0"	
4	DAC_CAL_FTDC_TOA<4>	"0"	
5	DAC_CAL_FTDC_TOA<5>	"0"	
6	NA	"0"	
7	NA	"0"	

**Register # 9**

Bit	Name	Default	Description



<b>0</b>	IN_FTDC_ENCODER_TOA<0>	"0"	Adjust the ToA FTDC offset by 5 bits <0:4> and the sign 1 bit <5>  OFFSET = FTDC_TDC<0:5> + (sign <5> x DATA<0:4>)
<b>1</b>	IN_FTDC_ENCODER_TOA<1>	"0"	
<b>2</b>	IN_FTDC_ENCODER_TOA<2>	"0"	
<b>3</b>	IN_FTDC_ENCODER_TOA<3>	"0"	
<b>4</b>	IN_FTDC_ENCODER_TOA<4>	"0"	
<b>5</b>	IN_FTDC_ENCODER_TOA<5>	"0"	
<b>6</b>	NA	"0"	
<b>7</b>	NA	"0"	

**Register # 10**

Bit	Name	Default	Description
<b>0</b>	IN_FTDC_ENCODER_TOT<0>	"0"	Adjust the Tot FTDC offset by 5 bits <0:4> and the sign 1 bit <5>  OFFSET = FTDC_TDC<0:5> + (sign <5> x DATA<0:4>)
<b>1</b>	IN_FTDC_ENCODER_TOT<1>	"0"	
<b>2</b>	IN_FTDC_ENCODER_TOT<2>	"0"	
<b>3</b>	IN_FTDC_ENCODER_TOT<3>	"0"	
<b>4</b>	IN_FTDC_ENCODER_TOT<4>	"0"	
<b>5</b>	IN_FTDC_ENCODER_TOT<5>	"0"	



6	NA	"0"	
7	DIS_TDC	"0"	"1" = TDC clocks off

**Register # 11**

Bit	Name	Default	Description
0	ExtData<8>	"0"	Forced ADC data, bits 9 & 8
1	ExtData<9>	"0"	
2	NA	"0"	
3	NA	"0"	
4	NA	"0"	
5	NA	"0"	
6	NA	"0"	
7	Mask_AlignBuffer	"0"	"1" = AlignBuffer clock off

**Register # 12**

Bit	Name	Default	Description
0	Adc_pedestal<0>	"0"	ADC pedestal value
1	Adc_pedestal<1>	"0"	



2	Adc_pedestal<2>	"0"
3	Adc_pedestal<3>	"0"
4	Adc_pedestal<4>	"0"
5	Adc_pedestal<5>	"0"
6	Adc_pedestal<6>	"0"
7	Adc_pedestal<7>	"0"

---

**Register # 13**

Bit	Name	Default	Description
0	ExtData<0>	"0"	Forced ADC data, bits 7 downto 0
1	ExtData<1>	"0"	
2	ExtData<2>	"0"	
3	ExtData<3>	"0"	
4	ExtData<4>	"0"	
5	ExtData<5>	"0"	
6	ExtData<6>	"0"	
7	ExtData<7>	"0"	

**GLOBAL\_ANALOG TABLE**

Register # 0			
Bit	Name	Default	Description
0	ON_pa	"1"	
1	Trim_vbo_pa<0>	"0"	
2	Trim_vbo_pa<1>	"1"	
3	Trim_vbo_pa<2>	"1"	
4	Trim_vbi_pa<0>	"1"	
5	Trim_vbi_pa<1>	"1"	
6	Trim_vbi_pa<2>	"1"	
7	Trim_vbi_pa<3>	"0"	

Register # 1			
Bit	Name	Default	Description
0	ON_rtr	"1"	
1	Dac_pol	"1"	
2	Range_tot<0>	"0"	



<b>3</b>	Range_tot<1>	"1"	
<b>4</b>	Range_tot<2>	"0"	
<b>5</b>	Range_indac<0>	"0"	
<b>6</b>	Range_indac<1>	"1"	
<b>7</b>	Range_indac<2>	"0"	

**Register # 2**

Bit	Name	Default	Description
<b>0</b>	ON_toa	"1"	"1" = enable TOA discri bias
<b>1</b>	ON_tot	"1"	"1" = enable TOT discri bias
<b>2</b>	Gain_itot<0>	"0"	6b-DAC for TOT gain tuning ("000000" = no feedback current as the original TOT architecture)
<b>3</b>	Gain_itot<1>	"0"	
<b>4</b>	Gain_itot<2>	"0"	
<b>5</b>	Gain_itot<3>	"0"	
<b>6</b>	Gain_itot<4>	"1"	
<b>7</b>	Gain_itot<5>	"1"	

**Register # 3**



Bit	Name	Default	Description
0	Ibi_sk<0>	"0"	S-K amplifier input stage current
1	Ibi_sk<1>	"0"	
2	Ibo_sk<0>	"0"	S-K amplifier output stage current
3	Ibo_sk<1>	"1"	
4	Ibo_sk<2>	"0"	
5	Ibo_sk<3>	"1"	
6	Ibo_sk<4>	"0"	
7	Ibo_sk<5>	"0"	

**Register # 4**

Bit	Name	Default	Description
0	Ibi_inv<0>	"0"	Inverter amplifier input stage current
1	Ibi_inv<1>	"0"	
2	Ibo_inv<0>	"0"	Inverter amplifier output stage current
3	Ibo_inv<1>	"1"	
4	Ibo_inv<2>	"0"	



5	Ibo_inv<3>	"1"
6	Ibo_inv<4>	"0"
7	Ibo_inv<5>	"0"

Register # 5			
Bit	Name	Default	Description
0	Ibi_noinv<0>	"0"	Non Inverter amplifier input stage current
1	Ibi_noinv<1>	"0"	
2	Ibo_noinv<0>	"0"	Non Inverter amplifier output stage current
3	Ibo_noinv<1>	"1"	
4	Ibo_noinv<2>	"0"	
5	Ibo_noinv<3>	"1"	
6	Ibo_noinv<4>	"0"	
7	Ibo_noinv<5>	"0"	

Register # 6			
Bit	Name	Default	Description
0	Ibi_inv_buf<0>	"0"	Inverter buffer input stage current



<b>1</b>	Ibi_inv_buf<1>	"1"	
<b>2</b>	Ibo_inv_buf<0>	"0"	Inverter buffer output stage current
<b>3</b>	Ibo_inv_buf<1>	"1"	
<b>4</b>	Ibo_inv_buf<2>	"1"	
<b>5</b>	Ibo_inv_buf<3>	"0"	
<b>6</b>	Ibo_inv_buf<4>	"0"	
<b>7</b>	Ibo_inv_buf<5>	"1"	

**Register # 7**

Bit	Name	Default	Description
<b>0</b>	Ibi_noinv_buf<0>	"0"	Non Inverter buffer input stage current
<b>1</b>	Ibi_noinv_buf<1>	"1"	
<b>2</b>	Ibo_noinv_buf<0>	"0"	Non Inverter buffer output stage current
<b>3</b>	Ibo_noinv_buf<1>	"1"	
<b>4</b>	Ibo_noinv_buf<2>	"1"	
<b>5</b>	Ibo_noinv_buf<3>	"0"	
<b>6</b>	Ibo_noinv_buf<4>	"0"	



7	Ibo_noinv_buf<5>	"1"	
---	------------------	-----	--

**Register # 8**

Bit	Name	Default	Description
0	Rc<1>	"1"	Shaping time adjustment
1	Rc<0>	"0"	
2	NA		
3	En_hyst_tot	"1"	"1" = enable the TOT discri hysteresis
4	Cf_comp<0>	"0"	Preamp feedback comp. cap. <0> = 100fF, <1> = 200ff
5	Cf_comp<1>	"1"	
6	Neg	"1"	"1" = negative input polarity (Default)
7	Pol_trig_toa	"1"	Polarity of the TOA discri output

**Register # 9**

Bit	Name	Default	Description
0	Cf<0>	"0"	Preamp feedback cap. <0> = 50fF, <1> = 100fF, <2> = 200ff, <3> = 400ff In //
1	Cf<1>	"0"	
2	Cf<2>	"1"	



<b>3</b>	Cf<3>	"0"	
<b>4</b>	Rf<0>	"1"	Preamp feedback Res. <0> = 25K, <1> = 50K, <2> = 66.66K, <3> = 100K In //
<b>5</b>	Rf<1>	"0"	
<b>6</b>	Rf<2>	"1"	
<b>7</b>	Rf<3>	"0"	

**Register # 10**

Bit	Name	Default	Description
<b>0</b>	Range_inv<0>	"0"	
<b>1</b>	Range_inv<1>	"0"	
<b>2</b>	Range_inv<2>	"1"	
<b>3</b>	SelExtADC	"0"	"1" = Forced ADC data send to the DRAM
<b>4</b>	Clr_ADC	"1"	
<b>5</b>	S_sk<0>	"0"	S-K amp Miller cap. <0> = 50fF, <1> = 100fF, <2> = 200fF
<b>6</b>	S_sk<1>	"1"	
<b>7</b>	S_sk<2>	"0"	



<b>Register # 11</b>			
<b>Bit</b>	<b>Name</b>	<b>Default</b>	<b>Description</b>
<b>0</b>	ClrShaperTail	"1"	
<b>1</b>	SelRisingEdge	"1"	
<b>2</b>	S_inv<0>	"0"	Inverter amp Miller cap. <0> = 50fF, <1> = 100fF, <2> = 200fF
<b>3</b>	S_inv<1>	"1"	
<b>4</b>	S_inv<2>	"0"	
<b>5</b>	S_noinv<0>	"0"	Non Inverter amp Miller cap. <0> = 50fF, <1> = 100fF, <2> = 200fF
<b>6</b>	S_noinv<1>	"1"	
<b>7</b>	S_noinv<2>	"0"	

<b>Register # 12</b>			
<b>Bit</b>	<b>Name</b>	<b>Default</b>	<b>Description</b>
<b>0</b>	NA	"0"	
<b>1</b>	NA	"0"	
<b>2</b>	S_inv_buf<0>	"0"	Inverter buffer Miller cap. <0> = 100fF, <1> = 200fF, <2> = 400fF
<b>3</b>	S_inv_buf<1>	"1"	



<b>4</b>	S_inv_buf<2>	"0"	
<b>5</b>	S_noinv_buf<0>	"0"	Non Inverter buffer Miller cap. <0> = 100fF, <1> = 200fF, <2> = 400fF
<b>6</b>	S_noinv_buf<1>	"1"	
<b>7</b>	S_noinv_buf<2>	"0"	

---

**Register # 13**

Bit	Name	Default	Description
<b>0</b>	Ref_adc<0>	"0"	Input stage current of the Ref ADC OTA
<b>1</b>	Ref_adc<1>	"0"	
<b>2</b>	Delay40<0>	"0"	Delay tuning for bits <4:0> "000" = faster conversion
<b>3</b>	Delay40<1>	"0"	
<b>4</b>	Delay40<2>	"0"	Delay tuning for bits <6:5> "000" = faster conversion
<b>5</b>	Delay65<0>	"0"	
<b>6</b>	Delay65<1>	"0"	
<b>7</b>	Delay65<2>	"0"	

---

**Register # 14**

Bit	Name	Default	Description



<b>0</b>	ON_ref_adc	"1"	"1" = enable ADC ref OTA
<b>1</b>	NA		
<b>2</b>	Delay87<0>	"0"	Delay tuning for bits <8:7> "000" = faster conversion
<b>3</b>	Delay87<1>	"0"	
<b>4</b>	Delay87<2>	"0"	
<b>5</b>	Delay9<0>	"0"	
<b>6</b>	Delay9<1>	"0"	Delay tuning for bit <9> "000" = faster conversion
<b>7</b>	Delay9<2>	"0"	

**REFERENCE VOLTAGE TABLE**

Register # 0			
Bit	Name	Default	Description
<b>0</b>	NA		
<b>1</b>	NA		
<b>2</b>	Trim_refi<0>	"0"	Bandgap current tuning
<b>3</b>	Trim_refi<1>	"0"	



<b>4</b>	Trim_vbg_1v<0>	"0"	1V bandgap ref. tuning
<b>5</b>	Trim_vbg_1v<1>	"0"	
<b>6</b>	Trim_vbg_1v<2>	"1"	
<b>7</b>	ON_dac	"1"	"1" = enable DACs

Register # 1			
Bit	Name	Default	Description
<b>0</b>	Noinv_vref<0>	"0"	
<b>1</b>	Noinv_vref<1>	"0"	
<b>2</b>	Inv_vref<0>	"0"	
<b>3</b>	Inv_vref<1>	"0"	
<b>4</b>	Toa_vref<0>	"0"	
<b>5</b>	Toa_vref<1>	"0"	
<b>6</b>	Tot_vref<0>	"0"	
<b>7</b>	Tot_vref<1>	"0"	

Register # 2			
Bit	Name	Default	Description



<b>0</b>	Tot_vref<2>	"0"	TOT threshold global value
<b>1</b>	Tot_vref<3>	"0"	
<b>2</b>	Tot_vref<4>	"1"	
<b>3</b>	Tot_vref<5>	"1"	
<b>4</b>	Tot_vref<6>	"0"	
<b>5</b>	Tot_vref<7>	"1"	
<b>6</b>	Tot_vref<8>	"1"	
<b>7</b>	Tot_vref<9>	"0"	

**Register # 3**

Bit	Name	Default	Description
<b>0</b>	Toa_vref<2>	"0"	TOA threshold global value
<b>1</b>	Toa_vref<3>	"0"	
<b>2</b>	Toa_vref<4>	"1"	
<b>3</b>	Toa_vref<5>	"1"	
<b>4</b>	Toa_vref<6>	"1"	
<b>5</b>	Toa_vref<7>	"0"	



6	Toa_vref<8>	"0"	
7	Toa_vref<9>	"0"	

Register # 4			
Bit	Name	Default	Description
0	Inv_vref<2>	"1"	Inverter shaper global reference
1	Inv_vref<3>	"1"	
2	Inv_vref<4>	"1"	
3	Inv_vref<5>	"1"	
4	Inv_vref<6>	"0"	
5	Inv_vref<7>	"0"	
6	Inv_vref<8>	"0"	
7	Inv_vref<9>	"1"	

Register # 5			
Bit	Name	Default	Description
0	Noinv_vref<2>	"0"	Non Inverter shaper global reference
1	Noinv_vref<3>	"0"	



2	Noinv_vref<4>	"0"
3	Noinv_vref<5>	"0"
4	Noinv_vref<6>	"0"
5	Noinv_vref<7>	"1"
6	Noinv_vref<8>	"0"
7	Noinv_vref<9>	"0"

---

**Register # 6**

Bit	Name	Default	Description
0	Calib_dac<0>	"0"	Calibration DAC value
1	Calib_dac<1>	"0"	
2	Calib_dac<2>	"0"	
3	Calib_dac<3>	"0"	
4	Calib_dac<4>	"0"	
5	Calib_dac<5>	"0"	
6	Calib_dac<6>	"0"	
7	Calib_dac<7>	"0"	

**Register # 7**

Bit	Name	Default	Description
0	Calib_dac<8>	"0"	Calibration DAC value
1	Calib_dac<9>	"0"	
2	Calib_dac<10>	"0"	
3	Calib_dac<11>	"0"	
4	NA	"0"	
5	NA	"0"	
6	IntCtest	"0"	Selection of the Calibration DAC
7	ExtCtest	"0"	Selection of the external pulse test

**Register # 8**

Bit	Name	Default	Description
0	Probe_dc<0>	"0"	
1	Probe_dc<1>	"0"	
2	Probe_dc<2>	"0"	
3	Probe_dc<3>	"0"	



<b>4</b>	Probe_dc<4>	"0"	
<b>5</b>	Probe_dc1_0	"0"	
<b>6</b>	Probe_dc1_1	"0"	
<b>7</b>	Probe_dc2	"0"	

**MASTER TDC TABLE**

Register # 0			
Bit	Name	Default	Description
<b>0</b>	GLOBAL_TA_SELECT_GAIN_TOA<0>	"1"	
<b>1</b>	GLOBAL_TA_SELECT_GAIN_TOA<1>	"1"	
<b>2</b>	GLOBAL_TA_SELECT_GAIN_TOA<2>	"1"	
<b>3</b>	GLOBAL_TA_SELECT_GAIN_TOA<3>	"0"	
<b>4</b>	GLOBAL_TA_SELECT_GAIN_TOT<0>	"1"	
<b>5</b>	GLOBAL_TA_SELECT_GAIN_TOT<1>	"1"	
<b>6</b>	GLOBAL_TA_SELECT_GAIN_TOT<2>	"0"	
<b>7</b>	GLOBAL_TA_SELECT_GAIN_TOT<3>	"0"	

**Register # 1**

Bit	Name	Default	Description
0	GLOBAL_MODE_NO_TOT_SUB	"0"	
1	GLOBAL_LATENCY_TIME<0>	"0"	
2	GLOBAL_LATENCY_TIME<1>	"1"	
3	GLOBAL_LATENCY_TIME<2>	"0"	
4	GLOBAL_LATENCY_TIME<3>	"1"	
5	GLOBAL_MODE_FTDC_TOA_0	"0"	
6	GLOBAL_MODE_FTDC_TOA_1	"1"	
7	GLOBAL_SEU_TIME_OUT	"1"	

**Register # 2**

Bit	Name	Default	Description
0	BIAS_FOLLOWER_CAL_P_D<0>	"0"	
1	BIAS_FOLLOWER_CAL_P_D<1>	"0"	
2	BIAS_FOLLOWER_CAL_P_D<2>	"0"	
3	BIAS_FOLLOWER_CAL_P_D<3>	"0"	



4	BIAS_FOLLOWER_CAL_P_CTDC_EN	"0"	
5	INV_FRONT_40MHZ	"0"	
6	START_COUNTER	"1"	
7	CALIB_CHANNEL_DLL	"0"	

---

**Register # 3**

Bit	Name	Default	Description
0	VD_CTDC_P_D<0>	"0"	
1	VD_CTDC_P_D<1>	"0"	
2	VD_CTDC_P_D<2>	"0"	
3	VD_CTDC_P_D<3>	"0"	
4	VD_CTDC_P_D<4>	"0"	
5	VD_CTDC_P_DAC_EN	"0"	
6	EN_MASTER_CTDC_VOUT_INIT	"0"	
7	EN_MASTER_CTDC_DLL	"1"	

---

**Register # 4**

Bit	Name	Default	Description



<b>0</b>	BIAS_CAL_DAC_CTDC_P_D<0>	“0”	
<b>1</b>	BIAS_CAL_DAC_CTDC_P_D<1>	“0”	
<b>2</b>	CTDC_CALIB_FREQUENCY<0>	“0”	
<b>3</b>	CTDC_CALIB_FREQUENCY<1>	“1”	
<b>4</b>	CTDC_CALIB_FREQUENCY<2>	“0”	
<b>5</b>	CTDC_CALIB_FREQUENCY<3>	“0”	
<b>6</b>	CTDC_CALIB_FREQUENCY<4>	“0”	
<b>7</b>	CTDC_CALIB_FREQUENCY<5>	“0”	

**Register # 5**

Bit	Name	Default	Description
<b>0</b>	GLOBAL_MODE_TOA_DIRECT_OUTPUT	“0”	
<b>1</b>	BIAS_I_CTDC_D<0>	“0”	
<b>2</b>	BIAS_I_CTDC_D<1>	“0”	
<b>3</b>	BIAS_I_CTDC_D<2>	“0”	
<b>4</b>	BIAS_I_CTDC_D<3>	“1”	
<b>5</b>	BIAS_I_CTDC_D<4>	“1”	



6	BIAS_I_CTDC_D<5>	"0"	
7	FOLLOWER_CTDC_EN	"1"	

Register # 6			
Bit	Name	Default	Description
0	GLOBAL_EN_BUFFER_CTDC	"0"	
1	VD_CTDC_N_FORCE_MAX	"1"	
2	VD_CTDC_N_D<0>	"0"	
3	VD_CTDC_N_D<1>	"0"	
4	VD_CTDC_N_D<2>	"0"	
5	VD_CTDC_N_D<3>	"0"	
6	VD_CTDC_N_D<4>	"0"	
7	VD_CTDC_N_DAC_EN	"0"	

Register # 7			
Bit	Name	Default	Description
0	CTRL_IN_REF_CTDC_P_D<0>	"0"	
1	CTRL_IN_REF_CTDC_P_D<1>	"0"	



2	CTRL_IN_REF_CTDC_P_D<2>	"0"	
3	CTRL_IN_REF_CTDC_P_D<3>	"0"	
4	CTRL_IN_REF_CTDC_P_D<4>	"0"	
5	CTRL_IN_REF_CTDC_P_EN	"0"	
6	BIAS_CAL_DAC_CTDC_P_D<2>	"0"	
7	BIAS_CAL_DAC_CTDC_P_D<3>	"0"	

---

**Register # 8**

Bit	Name	Default	Description
0	CTRL_IN_SIG_CTDC_P_D<0>	"0"	
1	CTRL_IN_SIG_CTDC_P_D<1>	"0"	
2	CTRL_IN_SIG_CTDC_P_D<2>	"0"	
3	CTRL_IN_SIG_CTDC_P_D<3>	"0"	
4	CTRL_IN_SIG_CTDC_P_D<4>	"0"	
5	CTRL_IN_SIG_CTDC_P_EN	"0"	
6	GLOBAL_INIT_DAC_B_CTDC	"0"	
7	BIAS_CAL_DAC_CTDC_P_EN	"0"	



Register # 9			
Bit	Name	Default	Description
0	VD_FTDC_P_D<0>	"0"	
1	VD_FTDC_P_D<1>	"0"	
2	VD_FTDC_P_D<2>	"0"	
3	VD_FTDC_P_D<3>	"0"	
4	VD_FTDC_P_D<4>	"0"	
5	VD_FTDC_P_DAC_EN	"0"	
6	EN_MASTER_FTDC_VOUT_INIT	"0"	
7	EN_MASTER_FTDC_DLL	"1"	

Register # 10			
Bit	Name	Default	Description
0			
1	BIAS_FOLLOWER_CAL_P_FTDC_EN	"0"	
2	FTDC_CALIB_FREQUENCY<0>	"0"	
3	FTDC_CALIB_FREQUENCY<1>	"1"	



4	FTDC_CALIB_FREQUENCY<2>	"0"	
5	FTDC_CALIB_FREQUENCY<3>	"0"	
6	FTDC_CALIB_FREQUENCY<4>	"0"	
7	FTDC_CALIB_FREQUENCY<5>	"0"	

---

**Register # 11**

Bit	Name	Default	Description
0	EN_REF_BG	"1"	
1	BIAS_I_FTDC_D<0>	"0"	
2	BIAS_I_FTDC_D<1>	"0"	
3	BIAS_I_FTDC_D<2>	"0"	
4	BIAS_I_FTDC_D<3>	"1"	
5	BIAS_I_FTDC_D<4>	"1"	
6	BIAS_I_FTDC_D<5>	"0"	
7	FOLLOWER_FTDC_EN	"1"	

---

**Register # 12**

Bit	Name	Default	Description



<b>0</b>	GLOBAL_EN_BUFFER_FTDC	"0"	
<b>1</b>	VD_FTDC_N_FORCE_MAX	"1"	
<b>2</b>	VD_FTDC_N_D<0>	"0"	
<b>3</b>	VD_FTDC_N_D<1>	"0"	
<b>4</b>	VD_FTDC_N_D<2>	"0"	
<b>5</b>	VD_FTDC_N_D<3>	"0"	
<b>6</b>	VD_FTDC_N_D<4>	"0"	
<b>7</b>	VD_FTDC_N_DAC_EN	"0"	

---

**Register # 13**

Bit	Name	Default	Description
<b>0</b>	CTRL_IN_SIG_FTDC_P_D<0>	"0"	
<b>1</b>	CTRL_IN_SIG_FTDC_P_D<1>	"0"	
<b>2</b>	CTRL_IN_SIG_FTDC_P_D<2>	"0"	
<b>3</b>	CTRL_IN_SIG_FTDC_P_D<3>	"0"	
<b>4</b>	CTRL_IN_SIG_FTDC_P_D<4>	"0"	
<b>5</b>	CTRL_IN_SIG_FTDC_P_EN	"0"	



6	BIAS_FOLLOWER_CAL_P_FTDC_D<0>	"0"	
7	BIAS_FOLLOWER_CAL_P_FTDC_D<1>	"0"	

**Register # 14**

Bit	Name	Default	Description
0	CTRL_IN_REF_FTDC_P_D<0>	"0"	
1	CTRL_IN_REF_FTDC_P_D<1>	"0"	
2	CTRL_IN_REF_FTDC_P_D<2>	"0"	
3	CTRL_IN_REF_FTDC_P_D<3>	"0"	
4	CTRL_IN_REF_FTDC_P_D<4>	"0"	
5	CTRL_IN_REF_FTDC_P_EN	"0"	
6	BIAS_FOLLOWER_CAL_P_FTDC_D<2>	"0"	
7	BIAS_FOLLOWER_CAL_P_FTDC_D<3>	"0"	

**Register # 15**

Bit	Name	Default	Description
0	GLOBAL_DISABLE_TOT_LIMIT	"0"	
1	GLOBAL_FORCE_EN_CLK	"0"	



<b>2</b>	GLOBAL_FORCE_EN_OUTPUT_DATA	"0"	
<b>3</b>	GLOBAL_FORCE_EN_TOT	"0"	
<b>4</b>	GLOBAL_FORCE_EN_TOT_PRIORITY	"0"	
<b>5</b>	GLOBAL_EN_TUNE_GAIN_DAC	"0"	
<b>6</b>	Sel_clk_rcg<0>	"0"	
<b>7</b>	Sel_clk_rcg<1>	"0"	

**DIGITAL HALF TABLE**

Register # 0			
Bit	Name	Default	Description
<b>0</b>	ClrADCTot_trig<0>	"0"	
<b>1</b>	ClrADCTot_trig<1>	"0"	
<b>2</b>	ClrADCTot_trig<2>	"0"	
<b>3</b>	ClrADCTot_trig<3>	"0"	
<b>4</b>	ClrADCTot_trig<4>	"0"	
<b>5</b>	ClrADCTot_trig<5>	"0"	
<b>6</b>	ClrADCTot_trig<6>	"0"	



7	ClrADCTot_trig<7>	"0"	
---	-------------------	-----	--

Register # 1			
Bit	Name	Default	Description
0	ClrADCTot_trig<8>	"0"	
1	ClrADCTot_trig<9>	"0"	
2	ClrADCTot_trig<10>	"0"	
3	ClrADCTot_trig<11>	"0"	
4	ClrADCTot_trig<12>	"0"	
5	ClrADCTot_trig<13>	"0"	
6	ClrADCTot_trig<14>	"0"	
7	ClrADCTot_trig<15>	"0"	

Register # 2			
Bit	Name	Default	Description
0	ClrADCTot_trig<16>	"0"	
1	ClrADCTot_trig<17>	"0"	
2	ClrADCTot_trig<18>	"0"	



<b>3</b>	ClrADCTot_trig<19>	"0"	
<b>4</b>	ClrADCTot_trig<20>	"0"	
<b>5</b>	ClrADCTot_trig<21>	"0"	
<b>6</b>	ClrADCTot_trig<22>	"0"	
<b>7</b>	ClrADCTot_trig<23>	"0"	

---

**Register # 3**

Bit	Name	Default	Description
<b>0</b>	ClrADCTot_trig<24>	"0"	
<b>1</b>	ClrADCTot_trig<25>	"0"	
<b>2</b>	ClrADCTot_trig<26>	"0"	
<b>3</b>	ClrADCTot_trig<27>	"0"	
<b>4</b>	ClrADCTot_trig<28>	"0"	
<b>5</b>	ClrADCTot_trig<29>	"0"	
<b>6</b>	ClrADCTot_trig<30>	"0"	
<b>7</b>	ClrADCTot_trig<31>	"0"	

---

**Register # 4**



Bit	Name	Default	Description
0	ClrADCTot_trig<32>	"0"	
1	ClrADCTot_trig<33>	"0"	
2	ClrADCTot_trig<34>	"0"	
3	ClrADCTot_trig<35>	"0"	
4	Tot_P_Add	"0"	A "0" value corresponds to a positive TOT pedestal used in the charge linearization for the trigger path
5	SC_testRAM	"0"	
6	CalibrationSC	"0"	
7	SelTC4	"1"	

Register # 5			
Bit	Name	Default	Description
0	Tot_TH3<0>	"0"	TOT threshold used in TP (common to 9 channels)
1	Tot_TH3<1>	"0"	
2	Tot_TH3<2>	"0"	
3	Tot_TH3<3>	"0"	



4	Tot_TH3<4>	"0"	
5	Tot_TH3<5>	"0"	
6	Tot_TH3<6>	"0"	
7	Tot_TH3<7>	"0"	

---

**Register # 6**

Bit	Name	Default	Description
0	Tot_TH2<0>	"0"	TOT threshold used in TP (common to 9 channels)
1	Tot_TH2<1>	"0"	
2	Tot_TH2<2>	"0"	
3	Tot_TH2<3>	"0"	
4	Tot_TH2<4>	"0"	
5	Tot_TH2<5>	"0"	
6	Tot_TH2<6>	"0"	
7	Tot_TH2<7>	"0"	

---

**Register # 7**

Bit	Name	Default	Description



<b>0</b>	Tot_TH1<0>	"0"	TOT threshold used in TP (common to 9 channels)
<b>1</b>	Tot_TH1<1>	"0"	
<b>2</b>	Tot_TH1<2>	"0"	
<b>3</b>	Tot_TH1<3>	"0"	
<b>4</b>	Tot_TH1<4>	"0"	
<b>5</b>	Tot_TH1<5>	"0"	
<b>6</b>	Tot_TH1<6>	"0"	
<b>7</b>	Tot_TH1<7>	"0"	

---

**Register # 8**

Bit	Name	Default	Description
<b>0</b>	Tot_TH0<0>	"0"	TOT threshold used in TP (common to 9 channels)
<b>1</b>	Tot_TH0<1>	"0"	
<b>2</b>	Tot_TH0<2>	"0"	
<b>3</b>	Tot_TH0<3>	"0"	
<b>4</b>	Tot_TH0<4>	"0"	
<b>5</b>	Tot_TH0<5>	"0"	



6	Tot_TH0<6>	"0"	
7	Tot_TH0<7>	"0"	

Register # 9			
Bit	Name	Default	Description
0	Tot_P3<0>	"0"	TOT pedestal used in TP (common to 9 channels)
1	Tot_P3<1>	"0"	
2	Tot_P3<2>	"0"	
3	Tot_P3<3>	"0"	
4	Tot_P3<4>	"0"	
5	Tot_P3<5>	"0"	
6	Tot_P3<6>	"0"	
7	NA	"0"	

Register # 10			
Bit	Name	Default	Description
0	Tot_P2<0>	"0"	TOT pedestal used in TP (common to 9 channels)
1	Tot_P2<1>	"0"	



2	Tot_P2<2>	"0"	
3	Tot_P2<3>	"0"	
4	Tot_P2<4>	"0"	
5	Tot_P2<5>	"0"	
6	Tot_P2<6>	"0"	
7	NA	"0"	

---

**Register # 11**

Bit	Name	Default	Description
0	Tot_P1<0>	"0"	TOT pedestal used in TP (common to 9 channels)
1	Tot_P1<1>	"0"	
2	Tot_P1<2>	"0"	
3	Tot_P1<3>	"0"	
4	Tot_P1<4>	"0"	
5	Tot_P1<5>	"0"	
6	Tot_P1<6>	"0"	
7	NA	"0"	

**Register # 12**

Bit	Name	Default	Description
0	Tot_P0<0>	"0"	TOT pedestal used in TP (common to 9 channels)
1	Tot_P0<1>	"0"	
2	Tot_P0<2>	"0"	
3	Tot_P0<3>	"0"	
4	Tot_P0<4>	"0"	
5	Tot_P0<5>	"0"	
6	Tot_P0<6>	"0"	
7	NA	"0"	

**Register # 13**

Bit	Name	Default	Description
0	MultFactor<0>	"1"	TOT vs ADC ratio for linearization (default ~25)
1	MultFactor<1>	"0"	
2	MultFactor<2>	"0"	
3	MultFactor<3>	"1"	



<b>4</b>	MultFactor<4>	"1"	
<b>5</b>	NA	"0"	
<b>6</b>	NA	"0"	
<b>7</b>	NA	"0"	

Register # 14			
Bit	Name	Default	Description
<b>0</b>	Adc_TH<0>	"0"	Threshold corresponding to noise in ADC count
<b>1</b>	Adc_TH<1>	"0"	
<b>2</b>	Adc_TH<2>	"0"	
<b>3</b>	Adc_TH<3>	"0"	
<b>4</b>	Adc_TH<4>	"0"	
<b>5</b>	NA	"0"	
<b>6</b>	NA	"0"	
<b>7</b>	L1Offset<8>	"0"	

Register # 15			
Bit	Name	Default	Description



<b>0</b>	L1Offset<0>	"0"	L1 offset corresponding to L1 Latency
<b>1</b>	L1Offset<1>	"0"	
<b>2</b>	L1Offset<2>	"0"	
<b>3</b>	L1Offset<3>	"1"	
<b>4</b>	L1Offset<4>	"0"	
<b>5</b>	L1Offset<5>	"0"	
<b>6</b>	L1Offset<6>	"0"	
<b>7</b>	L1Offset<7>	"0"	

**Register # 16**

Bit	Name	Default	Description
<b>0</b>	IdleFrame<0>	"0"	Default 28 LSB "1100 --- 1100" of idle DAQ/T Frame
<b>1</b>	IdleFrame<1>	"0"	
<b>2</b>	IdleFrame<2>	"1"	
<b>3</b>	IdleFrame<3>	"1"	
<b>4</b>	IdleFrame<4>	"0"	
<b>5</b>	IdleFrame<5>	"0"	



6	IdleFrame<6>	"1"	
7	IdleFrame<7>	"1"	

Register # 17			
Bit	Name	Default	Description
0	IdleFrame<8>	"0"	Default 28 LSB "1100 --- 1100" of idle DAQ/T Frame
1	IdleFrame<9>	"0"	
2	IdleFrame<10>	"1"	
3	IdleFrame<11>	"1"	
4	IdleFrame<12>	"0"	
5	IdleFrame<13>	"0"	
6	IdleFrame<14>	"1"	
7	IdleFrame<15>	"1"	

Register # 18			
Bit	Name	Default	Description
0	IdleFrame<16>	"0"	Default 28 LSB "1100 --- 1100" of idle DAQ/T Frame
1	IdleFrame<17>	"0"	



2	IdleFrame<18>	"1"
3	IdleFrame<19>	"1"
4	IdleFrame<20>	"0"
5	IdleFrame<21>	"0"
6	IdleFrame<22>	"1"
7	IdleFrame<23>	"1"

---

**Register # 19**

Bit	Name	Default	Description
0	IdleFrame<24>	"0"	Default 28 LSB "1100 --- 1100" of idle DAQ/T Frame
1	IdleFrame<25>	"0"	
2	IdleFrame<26>	"1"	
3	IdleFrame<27>	"1"	
4	NA	"0"	
5	NA	"0"	
6	NA	"0"	
7	NA	"0"	

**Register # 20**

Bit	Name	Default	Description
0	Sc_testRAM<0>	"0"	Default test RAM pattern
1	Sc_testRAM<1>	"0"	
2	Sc_testRAM<2>	"1"	
3	Sc_testRAM<3>	"1"	
4	Sc_testRAM<4>	"0"	
5	Sc_testRAM<5>	"0"	
6	Sc_testRAM<6>	"1"	
7	Sc_testRAM<7>	"1"	

**Register # 21**

Bit	Name	Default	Description
0	Sc_testRAM<8>	"0"	Default test RAM pattern
1	Sc_testRAM<9>	"0"	
2	Sc_testRAM<10>	"1"	
3	Sc_testRAM<11>	"1"	



<b>4</b>	Sc_testRAM<12>	“0”	
<b>5</b>	Sc_testRAM<13>	“0”	
<b>6</b>	Sc_testRAM<14>	“1”	
<b>7</b>	Sc_testRAM<15>	“1”	

---

**Register # 22**

Bit	Name	Default	Description
<b>0</b>	Sc_testRAM<16>	“0”	Default test RAM pattern
<b>1</b>	Sc_testRAM<17>	“0”	
<b>2</b>	Sc_testRAM<18>	“1”	
<b>3</b>	Sc_testRAM<19>	“1”	
<b>4</b>	Sc_testRAM<20>	“0”	
<b>5</b>	Sc_testRAM<21>	“0”	
<b>6</b>	Sc_testRAM<22>	“1”	
<b>7</b>	Sc_testRAM<23>	“1”	

---

**Register # 23**

Bit	Name	Default	Description



<b>0</b>	Sc_testRAM<24>	"0"	Default test RAM pattern
<b>1</b>	Sc_testRAM<25>	"0"	
<b>2</b>	Sc_testRAM<26>	"1"	
<b>3</b>	Sc_testRAM<27>	"1"	
<b>4</b>	Sc_testRAM<28>	"0"	
<b>5</b>	Sc_testRAM<29>	"0"	
<b>6</b>	Sc_testRAM<30>	"1"	
<b>7</b>	Sc_testRAM<31>	"1"	

---

**Register # 24**

Bit	Name	Default	Description
<b>0</b>	Bx_trigger<0>	"1"	
<b>1</b>	Bx_trigger<1>	"1"	
<b>2</b>	Bx_trigger<2>	"1"	
<b>3</b>	Bx_trigger<3>	"1"	
<b>4</b>	Bx_trigger<4>	"0"	
<b>5</b>	Bx_trigger<5>	"0"	



6	Bx_trigger<6>	"0"	
7	Bx_trigger<7>	"0"	

Register # 25			
Bit	Name	Default	Description
0	Bx_offset<0>	"1"	Reset value of the bunch crossing counter (BCD)
1	Bx_offset<1>	"0"	
2	Bx_offset<2>	"0"	
3	Bx_offset<3>	"0"	
4	Bx_offset<4>	"0"	
5	Bx_offset<5>	"0"	
6	Bx_offset<6>	"0"	
7	Bx_offset<7>	"0"	

Register # 26			
Bit	Name	Default	Description
0	Bx_offset<8>	"0"	



<b>1</b>	Bx_offset<9>	"0"	
<b>2</b>	Bx_offset<10>	"0"	
<b>3</b>	Bx_offset<11>	"0"	
<b>4</b>	Bx_trigger<8>	"0"	
<b>5</b>	Bx_trigger<9>	"0"	
<b>6</b>	Bx_trigger<10>	"0"	
<b>7</b>	Bx_trigger<11>	"0"	

**TOP TABLE**

Register # 0			
Bit	Name	Default	Description
<b>0</b>	RunR	"0"	
<b>1</b>	RunL	"0"	
<b>2</b>	TestMode	"0"	
<b>3</b>	N_counter_reset	"1"	
<b>4</b>	EdgeSel_T1	"0"	
<b>5</b>	In_inv_cmd_rx	"0"	



6	PreL1AOffset<0>	"0"	
7	PreL1AOffset<1>	"0"	

Register # 1			
Bit	Name	Default	Description
0	EN_PLL	"1"	
1	DIV_PLL<0>	"1"	
2	DIV_PLL<1>	"1"	
3	FOLLOWER_PLL_EN	"1"	
4	Sel_strobe_ext	"0"	
5	Sel_40M_ext	"0"	
6	Sel_error	"0"	
7	Sel_lock	"0"	

Register # 2			
Bit	Name	Default	Description
0	VOUT_INIT_EXT_EN	"0"	
1	VOUT_INIT_EXT_D<0>	"0"	



2	VOUT_INIT_EXT_D<1>	"0"	
3	VOUT_INIT_EXT_D<2>	"0"	
4	VOUT_INIT_EXT_D<3>	"0"	
5	VOUT_INIT_EXT_D<4>	"0"	
6	EN_REF_BG	"1"	
7	VOUT_INIT_EN	"0"	

---

**Register # 3**

Bit	Name	Default	Description
0	EN_HIGH_CAPA	"1"	
1	BIAS_I_PLL_D<0>	"0"	
2	BIAS_I_PLL_D<1>	"0"	
3	BIAS_I_PLL_D<2>	"0"	
4	BIAS_I_PLL_D<3>	"1"	
5	BIAS_I_PLL_D<4>	"1"	
6	BIAS_I_PLL_D<5>	"0"	
7	EN_RCG	"0"	



<b>Register # 4</b>			
<b>Bit</b>	<b>Name</b>	<b>Default</b>	<b>Description</b>
<b>0</b>	INIT_DAC_EN	"0"	
<b>1</b>	INIT_D<0>	"0"	
<b>2</b>	INIT_D<1>	"0"	
<b>3</b>	INIT_D<2>	"0"	
<b>4</b>	INIT_D<3>	"0"	
<b>5</b>	INIT_D<4>	"0"	
<b>6</b>	Rcg_gain<0>	"0"	
<b>7</b>	Rcg_gain<1>	"0"	

<b>Register # 5</b>			
<b>Bit</b>	<b>Name</b>	<b>Default</b>	<b>Description</b>
<b>0</b>	EN<0>	"1"	
<b>1</b>	EN<1>	"1"	
<b>2</b>	EN<2>	"0"	
<b>3</b>	ENpE<0>	"0"	



4	ENpE<1>	"0"	
5	ENpE<2>	"0"	
6	S<0>	"0"	
7	S<1>	"0"	

Register # 6			
Bit	Name	Default	Description
0	EN_LOCK_CONTROL	"1"	
1	ERROR_LIMIT_SC<0>	"0"	
2	ERROR_LIMIT_SC<1>	"1"	
3	ERROR_LIMIT_SC<2>	"0"	
4	NA	"0"	
5	PLL_Lockec_sc	"0"	
6	EN_probe_pll	"0"	
7	EN_PhaseShift	"1"	

Register # 7			
Bit	Name	Default	Description



<b>0</b>	Phase_ck<0>	"0"	
<b>1</b>	Phase_ck<1>	"0"	
<b>2</b>	Phase_ck<2>	"0"	
<b>3</b>	Phase_ck<3>	"0"	
<b>4</b>	Phase_strobe<0>	"0"	
<b>5</b>	Phase_strobe<1>	"0"	
<b>6</b>	Phase_strobe<2>	"0"	
<b>7</b>	Phase_strobe<3>	"0"	

---

**Register # 8**

Bit	Name	Default	Description
<b>0</b>	B_out<0>	"0"	
<b>1</b>	B_out<1>	"0"	
<b>2</b>	B_out<2>	"0"	
<b>3</b>	B_out<3>	"0"	
<b>4</b>	B_out<4>	"0"	
<b>5</b>	B_out<5>	"0"	



6	B_out<6>	"0"	
7	B_out<7>	"0"	

Register # 9			
Bit	Name	Default	Description
0	B_in<0>	"0"	
1	B_in<1>	"0"	
2	B_in<2>	"0"	
3	B_in<3>	"0"	
4	B_in<4>	"0"	
5	B_in<5>	"0"	
6	B_in<6>	"0"	
7	B_in<7>	"0"	

Register # 10			
Bit	Name	Default	Description
0	B_out<8>	"0"	
1	B_out<9>	"0"	



<b>2</b>	B_out<10>	"0"	
<b>3</b>	B_out<11>	"0"	
<b>4</b>	B_out<12>	"0"	
<b>5</b>	B_out<13>	"0"	
<b>6</b>	B_out<14>	"0"	
<b>7</b>	B_out<15>	"0"	

---

**Register # 11**

Bit	Name	Default	Description
<b>0</b>	B_in<8>	"0"	
<b>1</b>	B_in<9>	"0"	
<b>2</b>	B_in<10>	"0"	
<b>3</b>	B_in<11>	"0"	
<b>4</b>	B_in<12>	"0"	
<b>5</b>	B_in<13>	"0"	
<b>6</b>	B_in<14>	"0"	
<b>7</b>	B_in<15>	"0"	

**Register # 12**

Bit	Name	Default	Description
0	B_out<16>	"0"	
1	B_out<17>	"0"	
2	B_out<18>	"0"	
3	B_out<19>	"0"	
4	B_out<20>	"0"	
5	B_out<21>	"0"	
6	B_out<22>	"0"	
7	B_out<23>	"0"	

**Register # 13**

Bit	Name	Default	Description
0	B_in<16>	"0"	
1	B_in<17>	"0"	
2	B_in<18>	"0"	
3	B_in<19>	"0"	



<b>4</b>	B_in<20>	"0"	
<b>5</b>	B_in<21>	"0"	
<b>6</b>	B_in<22>	"0"	
<b>7</b>	B_in<23>	"0"	

---

**Register # 14**

Bit	Name	Default	Description
<b>0</b>	B_out<24>	"0"	
<b>1</b>	B_out<25>	"0"	
<b>2</b>	B_out<26>	"0"	
<b>3</b>	B_out<27>	"0"	
<b>4</b>	B_out<28>	"0"	
<b>5</b>	B_out<29>	"0"	
<b>6</b>	B_out<30>	"0"	
<b>7</b>	SROUT	"0"	

---

**Register # 15**

Bit	Name	Default	Description



<b>0</b>	B_in<24>	“0”	
<b>1</b>	B_in<25>	“0”	
<b>2</b>	B_in<26>	“0”	
<b>3</b>	B_in<27>	“0”	
<b>4</b>	B_in<28>	“0”	
<b>5</b>	B_in<29>	“0”	
<b>6</b>	B_in<30>	“0”	
<b>7</b>	B_in<31>	“0”	

---

**Register # 16**

Bit	Name	Default	Description
<b>0</b>	statusR<0>	“0”	
<b>1</b>	statusR<1>	“0”	
<b>2</b>	statusR<2>	“0”	
<b>3</b>	statusR<3>	“0”	
<b>4</b>	statusL<0>	“0”	
<b>5</b>	statusL<1>	“0”	



6	statusL<2>	"0"	
7	statusL<3>	"0"	

Register # 17			
Bit	Name	Default	Description
0	Lock_count<0>	"0"	
1	Lock_count<1>	"0"	
2	Lock_count<2>	"0"	
3	Lock_count<3>	"0"	
4	Lock_count<4>	"0"	
5	Lock_count<5>	"0"	
6	Lock_count<6>	"0"	
7	Lock_count<7>	"0"	

Register # 18			
Bit	Name	Default	Description
0	Fc_error_count<0>	"0"	
1	Fc_error_count<1>	"0"	



<b>2</b>	Fc_error_count<2>	"0"	
<b>3</b>	Fc_error_count<3>	"0"	
<b>4</b>	Fc_error_count<4>	"0"	
<b>5</b>	Fc_error_count<5>	"0"	
<b>6</b>	Fc_error_count<6>	"0"	
<b>7</b>	Fc_error_count<7>	"0"	

---

**Register # 19**

Bit	Name	Default	Description
<b>0</b>	Err_countL<0>	"0"	
<b>1</b>	Err_countL<1>	"0"	
<b>2</b>	Err_countL<2>	"0"	
<b>3</b>	Err_countL<3>	"0"	
<b>4</b>	Err_countL<4>	"0"	
<b>5</b>	Err_countL<5>	"0"	
<b>6</b>	Err_countL<6>	"0"	
<b>7</b>	Err_countL<7>	"0"	



Register # 20			
Bit	Name	Default	Description
0	Err_countR<0>	"0"	
1	Err_countR<1>	"0"	
2	Err_countR<2>	"0"	
3	Err_countR<3>	"0"	
4	Err_countR<4>	"0"	
5	Err_countR<5>	"0"	
6	Err_countR<6>	"0"	
7	Err_countR<7>	"0"	

## 4. Measurements of v2 for inputs for HGCROCv3 specifications

### 4.1. Power consumption: RUN mode vs. SLEEP mode

For the SLEEP mode, Preamplifier, shapers, discriminators are switched off, StartUp\_Ok = 0 (no writing in RAM1)

For the RUN mode, all the chip is ON.

The power consumption decreases by 60%.

### 4.2. Power consumption: HD vs. LD

For the LD hexaboard, we would like to turn off channels 8, 17, 26, 35, 44, 53, 62, 71.

If we use channel\_off enabled (preamp input tied to ground) and ADC, TDC and AlignBuffer masked, the power consumption decreases by 3%.



## 5. Production testing and in-situ calibration

### 5.1. Production testing

The purpose of the production testing is to select the good chips and reject the others. Two robots will be available at LLR and OMEGA. The testing duration needs to last less than 3 minutes.

The following items should be checked:

- All DACs acts as expected
  - Four 10b DAC: Vref\_inv, Vref\_noinv, Vref\_toa, Vref\_tot
  - One 12b DAC: calibration DAC
  - several trimming DAC: bias
  - bandgap value: set to 1V

⇒ accessible via the monitoring probe\_dc1/2

- all channels
  - correct biasing: check the pedestal and noise
  - waveforms scan of all the channels, some charge (ADC, TOT), ADC phase / CalPulse phase, TOA
- Data path
  - RAM checks
- Trigger path
- Analyse digital data

### 5.2. In-situ testing

Ideally, the monitoring points should not be used by default for the in-situ calibration, the digitized data should be enough.

- loading of the default parameters (normally the ones set by HardResetb)
- settings of BX, trigger counter offset
- synchronization of the E-links
- waveform scan to find out the correct ADC phase
- global pedestal adjustment: two 2D scan on inv and non-inv shaper outputs
- channel-wise pedestal adjustment: 6b trimming dac
- global and channel-wise Toa thresholds adjustment on the pedestals
- Set Tot threshold to the highest value in order to find out the higher charge of the ADC range
- global and channel-wise Tot thresholds adjustment on the higher charge of the ADC range