

4.10.3 PCB stack-up and routing

The Level 1 PCB was manufactured by Lab Circuits S.A. with 12 layers, according to the stack-up shown in figure 4.29. This stack-up is appropriate to obtain low noise and to reduce the interference from the digital circuits into the analog signals, while at the same time, facilitating the further integration with Dragon board which uses a rather similar stack-up.

	Subclass Name	Type	Material	Thickness (MIL)	Conductivity (mho/cm)	Dielectric Constant	Loss Tangent	Negative Artwork	Shield	Width (MIL)
1		SURFACE	AIR			1	0			
2	TOP	CONDUCTOR	COPPER	0.709	595900	4.5	0	<input type="checkbox"/>		4.00
3		DIELECTRIC	FR-4	6.3	0	4.5	0.035			
4	AGND_TOP	PLANE	COPPER	1.378	595900	4.5	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
5		DIELECTRIC	FR-4	3.937	0	4.5	0.035			
6	L3	CONDUCTOR	COPPER	1.378	595900	4.5	0	<input type="checkbox"/>		4.00
7		DIELECTRIC	FR-4	3.937	0	4.5	0.035			
8	-3V_ANALOG	PLANE	COPPER	1.378	595900	4.5	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
9		DIELECTRIC	FR-4	3.937	0	4.5	0.035			
10	L5	CONDUCTOR	COPPER	1.378	595900	4.5	0	<input type="checkbox"/>		4.00
11		DIELECTRIC	FR-4	3.937	0	4.5	0.035			
12	+3V_DIGITAL	PLANE	COPPER	1.378	595900	4.5	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
13		DIELECTRIC	FR-4	3.937	0	4.5	0.035			
14	L7	CONDUCTOR	COPPER	1.378	595900	4.5	0.035	<input type="checkbox"/>		4.00
15		DIELECTRIC	FR-4	3.937	0	4.5	0.035			
16	DGND	PLANE	COPPER	1.378	595900	4.5	0.035	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
17		DIELECTRIC	FR-4	3.937	0	4.5	0.035			
18	+3V_ANALOG	PLANE	COPPER	1.378	595900	4.5	0.035	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
19		DIELECTRIC	FR-4	3.937	0	4.5	0.035			
20	L10	CONDUCTOR	COPPER	1.378	595900	4.5	0.035	<input type="checkbox"/>		4.00
21		DIELECTRIC	FR-4	3.937	0	4.5	0.035			
22	AGND_BOTTOM	PLANE	COPPER	1.378	595900	4.5	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
23		DIELECTRIC	FR-4	6.3	0	4.5	0.035			
24	BOTTOM	CONDUCTOR	COPPER	0.709	595900	4.5	0	<input type="checkbox"/>		4.00
25		SURFACE	AIR			1	0			

Total Thickness: 63.231 MIL Layer Type: ALL Material: ALL Field to Set: Thickness Value to Set: Update Fields Show Single Impedance Show Diff Impedance OK Apply Cancel Refresh Materials > Help

Figure 4.29: Level 1 PCB stack-up

The Level 1 was designed with Cadence Allegro software (see figure 4.30), configuring it properly to control the impedances of the lines and differential pairs. This is critical, specially to avoid the mismatching of the analog signals. Additionally, the analog lines are routed trying to use as few via holes as possible, trying to use the TOP and BOTTOM layers to reduce the noise coming from the digital lines, and avoiding sharp shapes. In short, the analog lines are routed according to the classical recommendations for high frequency routing.

4.11 Test bench

A complex test bench was developed at UCM to test the Level 1, including the manufacturing of some boards. The Level 1 Test Board (figure 4.31) was the first one, providing a place to plug the Level 1 and routing the signals in the Samtec connectors to other connectors compatible with the other elements of the test bench. In this way, the Level 1 Test Board provides connectivity to the power supplies, to the SPI adapter, and includes SMA connectors for the six input differential pairs as well as for the outputs. The delay of the input paths has been equalized to perform the

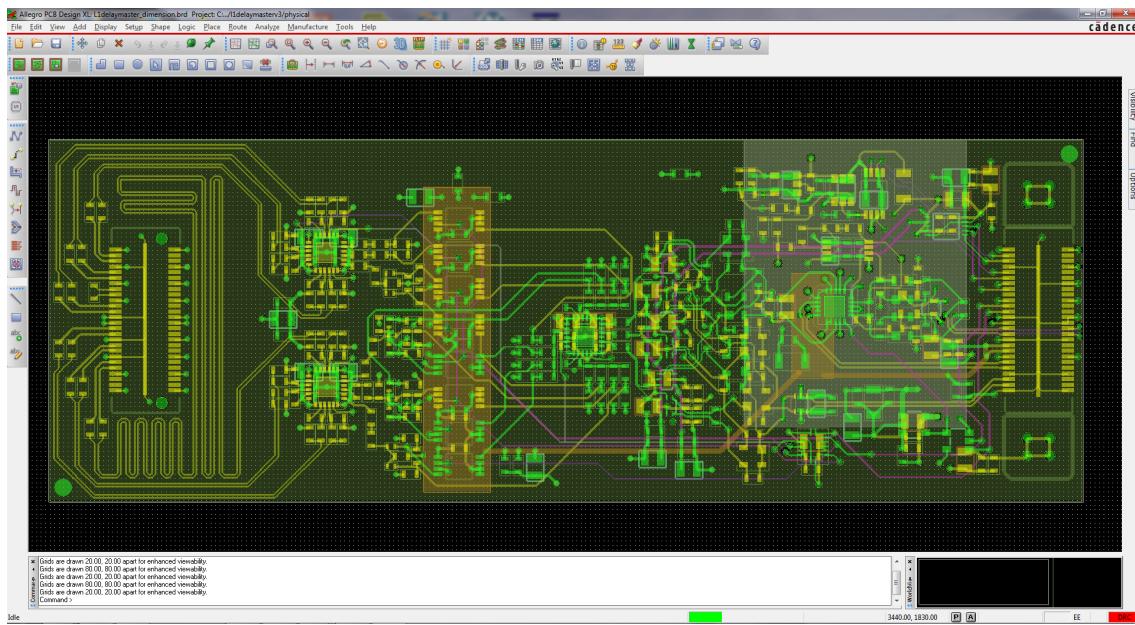


Figure 4.30: Level 1 PCB design in Allegro software

additions in the Level 1 properly, and the impedance is controlled. Additionally, the Level 1 test board contains a switch to control the bits A0 and A1 which are used to select the working mode.

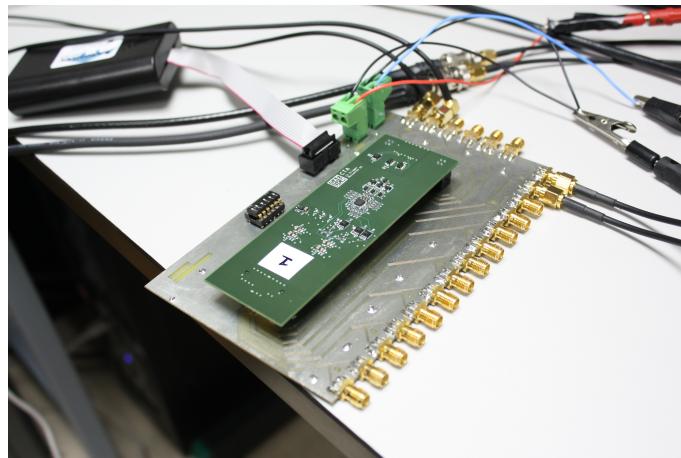


Figure 4.31: Level 1 test board

The second board manufactured to test the Level 1 was a differential analog pulse replicator, able to generate six copies of the differential input pulse. This board was used to replicate the pulses from the Agilent 81110A [127] and then being able to introduce signals in several inputs at the same time. The replicator board is connected to the Level 1 Test Board by means of common coaxial cables with SMA connectors, so by changing the length of the cables it is possible to introduce the pulses to the Level 1 at different times. The circuit used for the fan-out of the signals is the same one used in the Level 0 fan-out board, which is described in section 3.2.4.

The typical test bench used to measure the different parameters of the Level 1 is sketched in

figure 4.32, and figure 4.33 shows a photograph of it.

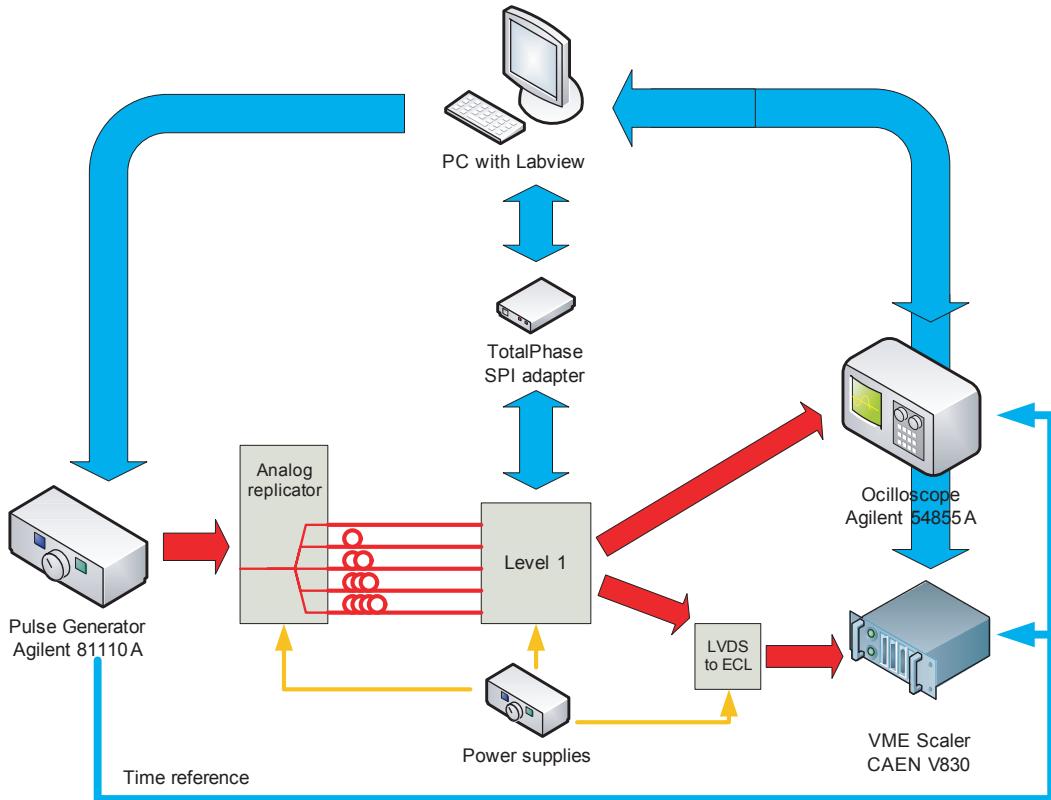


Figure 4.32: Level 1 test bench

The pulses from the pulse replicator board are introduced into the Level 1, where they are adequately treated and added to form the combinations which are compared with the corresponding threshold. The threshold is set up by sending the right messages to the DAC, by means of a SPI interface. This interface is handled by a device named Aardvark [117], from TotalPhase Inc.³ which is connected through an USB bus to the PC, where a Labview program is managing all the instruments.

Then, after the comparators and the OR gate, the LVDS digital differential output is sent to an Agilent 54855A oscilloscope [129] to test the timing parameters (delay, pulse width) and check if the output voltages are correct. It is important to realize that a timing reference is usually required for the trigger, as indicated in figure 4.32. The oscilloscope has also been used to see the wave form of the signal in different points of the analog circuit using a high impedance active probe but, as there are not good test points, the measurements obtained in this way are not fully accurate. Nevertheless, the oscilloscope with the probe are useful to detect mismatches or distortion of the pulses in a qualitative way, and sometimes it is the only way to measure at certain points of the circuit.

In order to obtain good amplitude measurements at the input of the comparators we have performed rate scans (see section 3.2.7.2). To do such rate scans, the LVDS output of the Level 1 is

³Also another similar device called Cheetah [128] was used

connected to a scaler (a pulse counter typically used in nuclear instrumentation) CAEN V830 [130]. As the scaler available only accepts ECL logic inputs, an auxiliary board was developed to translate from LVDS to ECL. In this way, the rate scan algorithm is as follows:

1. Set the threshold to the lowest value.
2. Generate a large burst of pulses (1000 or 10000) and count how many produces trigger outputs.
3. Increase the threshold and go back to 2, until reaching the maximum threshold value.
4. Represent the number of triggers counted versus the threshold voltages, and take the threshold value to count the 50% of the pulses in the burst as the amplitude of the analog pulses at the input of the comparators.

This algorithm is implemented in the Labview program running in the PC.

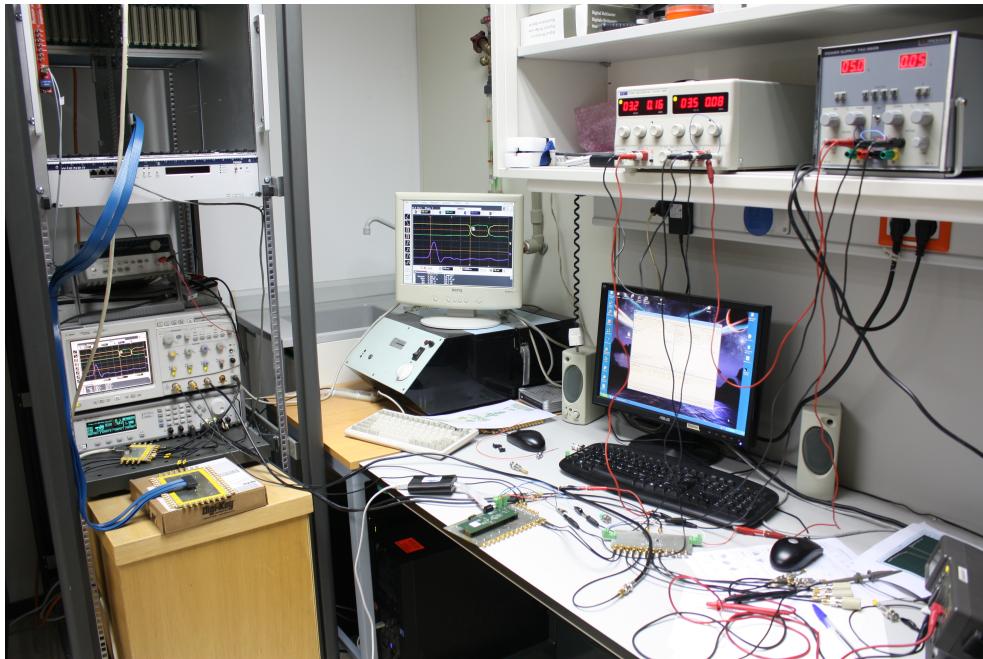


Figure 4.33: Photo of the Level 1 test bench

4.12 Measurements

This section gathers the most relevant measurements of the Level 1 as a standalone system.

4.12.1 Gain and linearity

The gain and linearity of the Level 1 has been measured for every channel in the different modes, to ensure a correct gain equalization. To obtain the different measurements, a rate scan was

performed for every input amplitude, considering the 50% threshold as the output amplitude. In this way, the graphics of figures 4.34, 4.35 and 4.36 show the gain and linearity of the Level 1 for the three different working modes, and for the last version of the Level 1 prototype. Every plot shows two graphics, because another output was added to the Level 1 to implement the COLIBRI, as will be explained in chapter 5. The gain value for each output was calculated by performing a linear fit of the measurement points, which are represented as the green and blue lines in the plots. The gain values obtained by every linear fit are written in the captions of the figures, and they ideally should be 0.9, as explained in previous sections. The measurements show very similar gain values for the different inputs and working modes, while the linearity is rather good for each single input channel with an amplitude between 0 and 1.2 V. Although the dynamic range is specified between 1 and 200 phe (8 mV to 1.6 V input), the highest values are only obtained by the addition of several inputs, being the maximum single channel amplitude limited to 1.2 V (150 phe).

4.12.2 Noise

The rate scans are also useful to measure the noise in the board with no more than disconnecting the inputs. By ramping the threshold voltage starting from 0 V, it can be seen that for very low thresholds there is a high amount of triggers due to noise. As the threshold increases, the number of triggers is reduced, until almost disappearing. The lowest threshold voltage for which there are almost no triggers⁴ due to noise can be considered as the noise of the Level 1, which contributes to the noise of the complete trigger system and, at the end, defines the minimum detectable signal (see section 3.2.7.3).

The lowest threshold which avoids triggering due to noise has been measured with the inputs open and connected to $50\ \Omega$ loads, and the results are gathered in table 4.4. The threshold voltages were measured directly with a voltmeter at the output of the DAC instead of reading the DAC code to avoid possible offset errors in the DAC. The level 1 noise measurements show noise values clearly lower than the 2 mV specified for the whole analog trigger chain (Level 0, Level 0 fan-out and Level 1), but the important noise measurement was the one shown in figure 3.36, including all the elements of the chain, which also showed to comply with the specifications. To sum-up, it can be say that the noise in the Level 1 trigger is low enough to have a global noise lower than 2 mV.

Input load	V_{th} board 1	V_{th} board 2
Open circuit	1.3 mV	1.4 mV
$50\ \Omega$	1.4 mV	1.4 mV

Table 4.4: Noise measurements of the Level 1 prototype, for two similar manufactured boards.

4.12.3 Pulse width and bandwidth

The analog stages of the Level 1 should receive 2.5 ns⁵ width pulses and should not stretch them, which means that the bandwidth should be around 400 MHz. As it is not possible to connect

⁴As the noise amplitude is typically gaussian, there can be noise pulses with large inputs, but very rarely. To be more formal, “almost no triggers” can be defined as less than 0.1% of the maximum number of triggers, measured with a 0 V threshold

⁵In fact, at least 3 ns due to stretching in the PMT itself, the Level 0 and Level 0 fan-out.

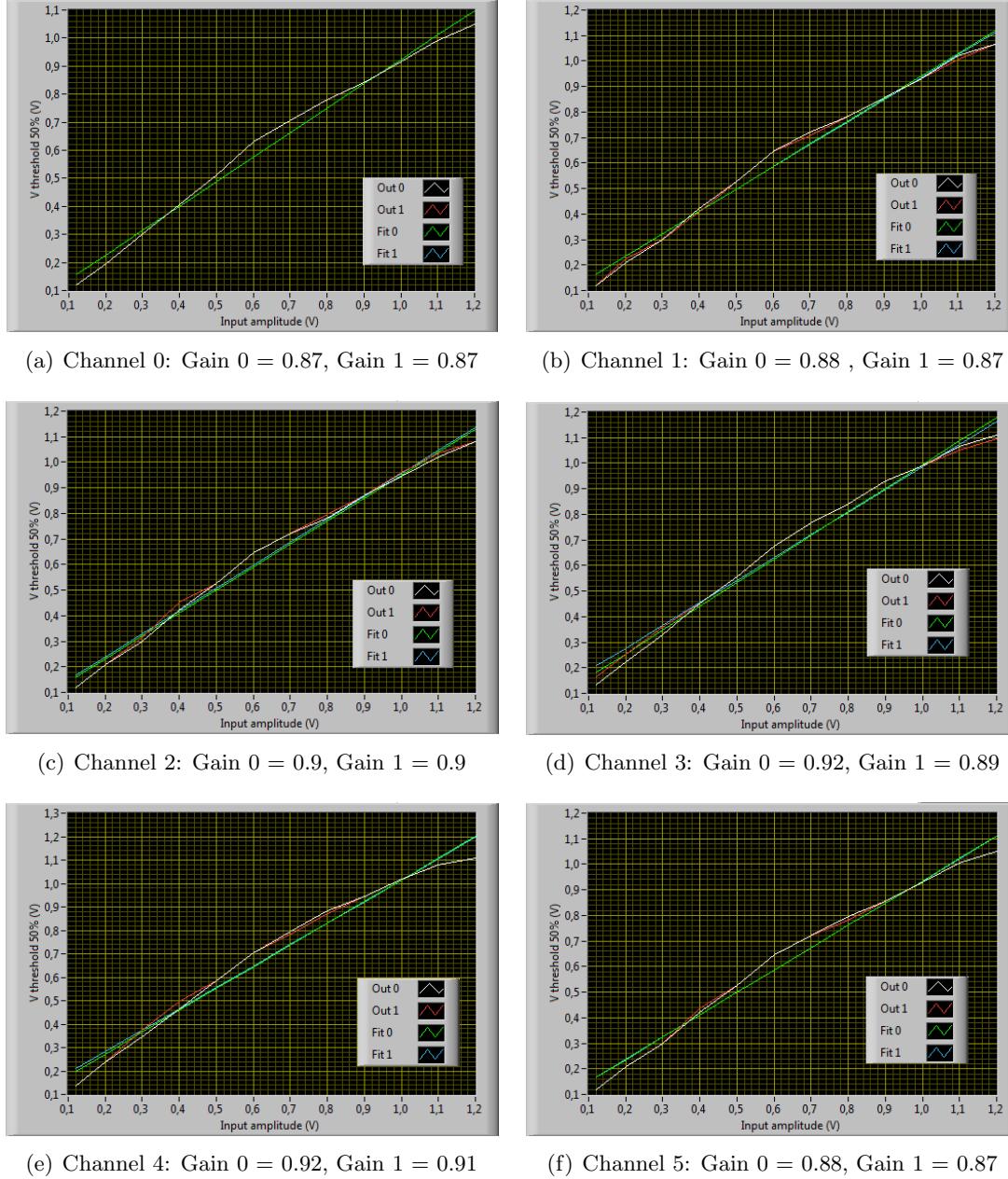


Figure 4.34: Measurements of the gain for the different channels in mode 4

a network vector analyzer to the input of the comparators and measure the frequency response directly, a different measurement was designed. Assuming a low pass response, the narrow pulses, with power at higher frequencies, should be more attenuated than the wider ones. In this way, pulses with different widths were set, and a rate scan was performed for each width to measure its amplitude at the input of the comparators (i.e., measuring the threshold to detect the 50% of the pulses). In this way, the values shown in red in figure 4.37 was obtained. It is obvious that the bandwidth was not high enough, being the pulses narrower than 4 ns strongly attenuated. The solution was to change the resistors of the differential to single-ended stage and the adder by smaller ones, to enlarge

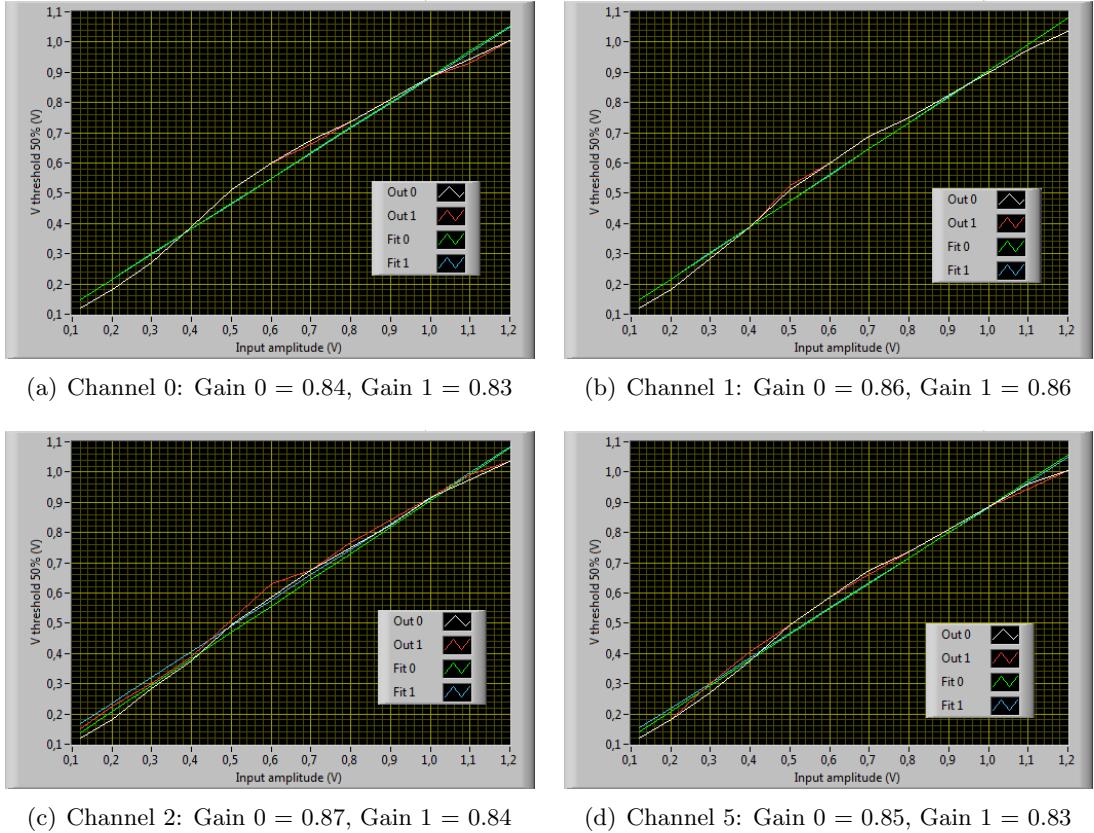


Figure 4.35: Measurements of the gain for the different channels in mode 3

the bandwidth of the low pass filters formed by the resistors and the parasitic capacitances. With the smaller resistors, the results obtained, which are shown in blue in figure 4.37, do not show any noticeable attenuation for pulses wider than 2 ns. The drawback of using lower resistance values is a slightly higher power consumption, but not large enough to be unaffordable.

Additionally, after the resistor change, the pulse width at the input and at the output of the adders (inputs of the comparators) was measured with an oscilloscope probe, observing no difference in the width. From both the measurement of figure 4.37 and the low precision direct observations with the oscilloscope, it can be deduced that the final bandwidth is at least as wide as the requirement.

4.12.4 Latency

The latency of the Level 1 and the test board were measured connecting the Level 1 LVDS output to the oscilloscope (yellow line in figure 4.38), and taking as reference another copy of the input pulse from the replicator board after an equivalent cable length (green line). The result was a global latency of 8.3 ns for the Level 1 and the test board as can be seen in figure 4.38. Green, purple and magenta lines in figure 4.38 correspond to different outputs of the replicator board, just to check that there are no noticeable differences between them. A low latency is beneficial in all the trigger elements, both to reduce the necessity of long analog memory buffers and to provide with more time to perform more complex trigger logic operations with a fixed memory length.

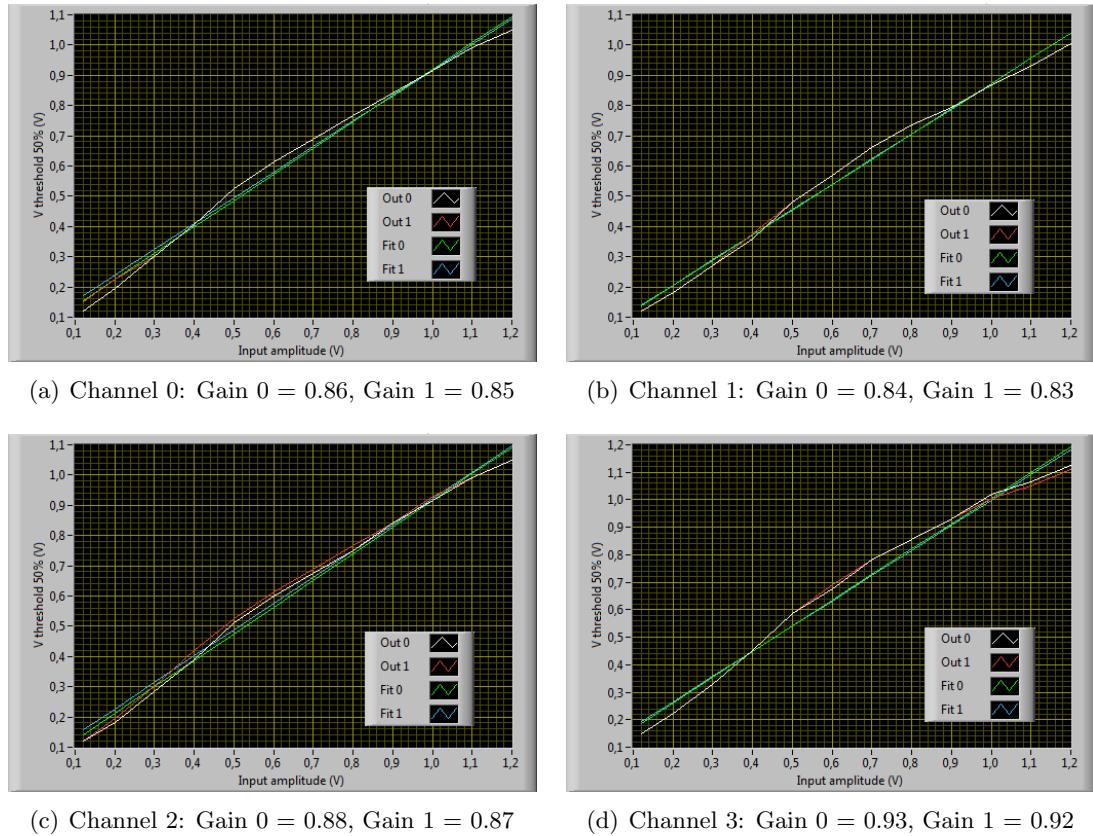


Figure 4.36: Measurements of the gain for the different channels in mode 2

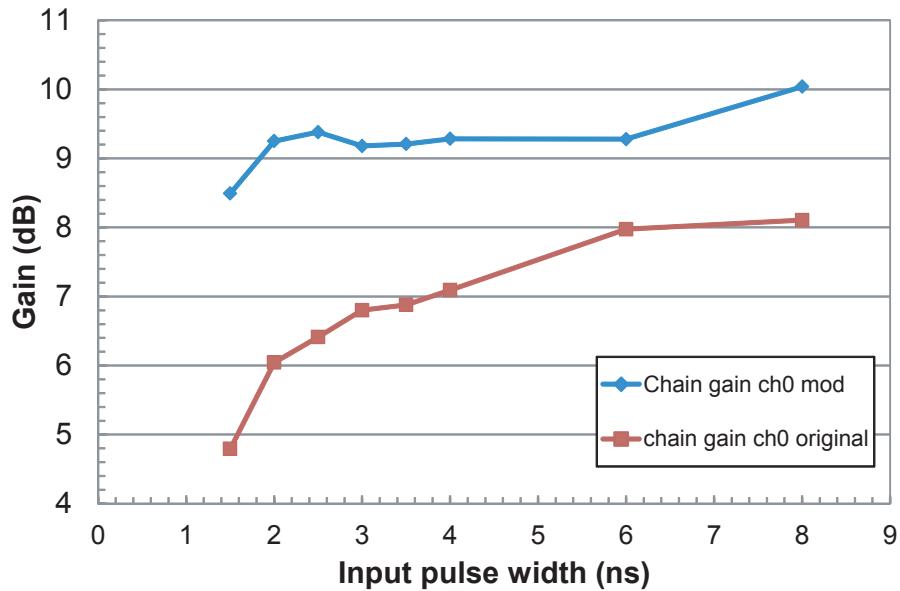


Figure 4.37: Gain vs pulse width

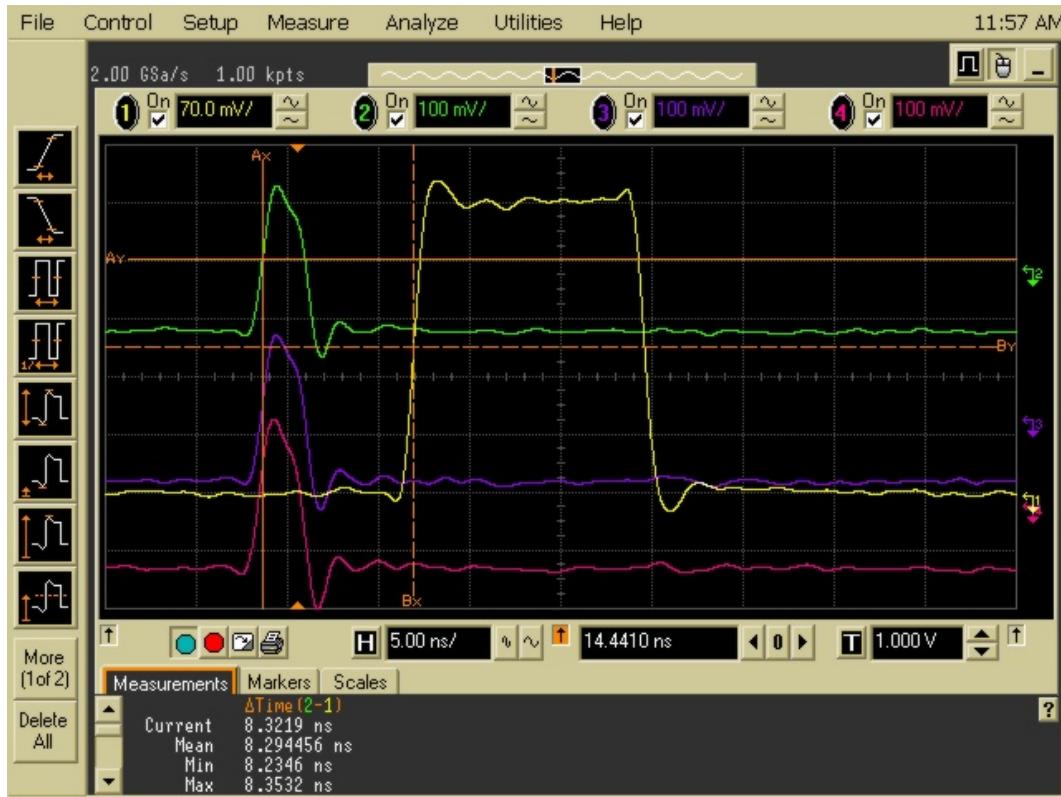


Figure 4.38: Latency of the Level 1

4.12.5 Time walk

The so-called time walk is an undesired effect, mainly due to the comparators, which variates the latency of the Level 1 depending on the voltage margin between the input amplitude and the threshold. Thus, if the threshold is exceeded by a large margin, e.g. 1 V, the output pulse appears sooner than in the case in which the threshold is exceeded by a narrow margin of few millivolts. To illustrate this effect, figure 4.39 shows the relative delay with respect to the output produced when the input amplitude is 2 V and the threshold is 0.1 V (i.e. the latency when the input is 1.9 V over the threshold has been chosen as “0 delay”). When the threshold is exceeded by more than 0.9 V, the relative delay remains lower than 100 ps but, when the input amplitude is reduced, becoming closer to the threshold, the output pulses have a longer latency. The effect is more prominent when the margin over the threshold is lower than 0.2 V and differences of up to 1.2 ns in the latency have been measured. It could seem strange to see points measured for amplitudes below the threshold in the plot 4.39, but the reason is very simple: when the input is set up to be slightly below the threshold, some pulses are still triggering due to noise and their delay was also measured and represented.

The time walk is a phenomenon present in most fast comparators and it is difficult to correct, adding a given uncertainty to the trigger latency. This uncertainty should be taken into account and start the digitizing of the analog memory samples some cells in advance to ensure the full recording of the pulse.

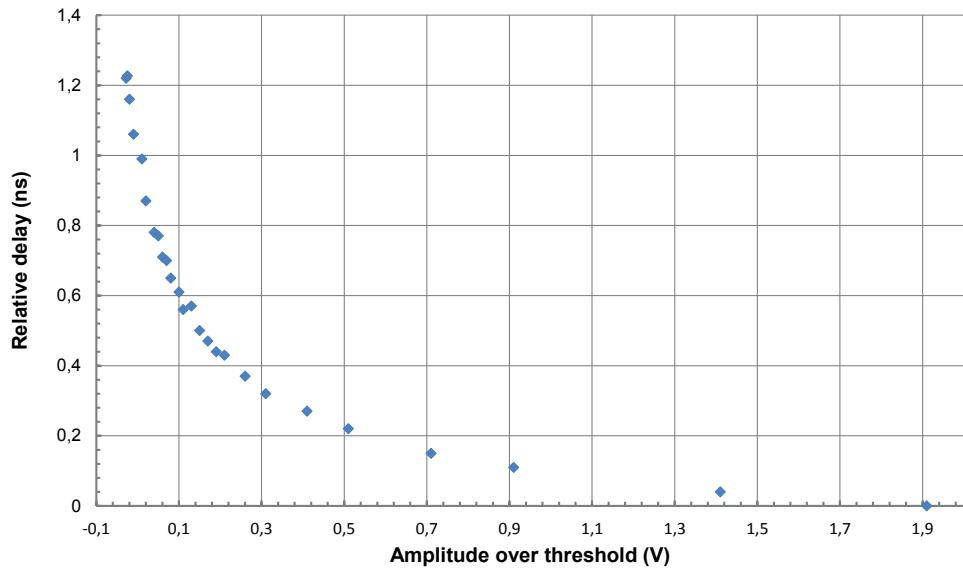


Figure 4.39: Time walk of the Level 1

4.12.6 Jitter

The jitter in this context can be defined as the latency variation due to the electronics, for the same input amplitude and the same threshold. This jitter has been measured as the standard deviation of the latency, and its value is only 21.6 ps as can be seen in figure 4.40. The difference in the global latencies shown in figure 4.40 and figure 4.38 is caused by the different cable lengths used for the output and the reference in the measurement of the jitter. In the measurement of the jitter the global latency was not important and for that reason the cables were not chosen carefully. As was explained in chapter 3, the jitter in the trigger system must be as low as possible in order to know exactly in which memory cell the data acquisition should start. The 21.6 ps measured show that the jitter introduced by the Level 1 is negligible.

4.13 Level 1 conclusions

A Level 1 trigger system has been designed, manufactured and tested by the author of this thesis. Its originality lies in the use of several components and techniques typical from RF and microwave design, such as the Wilkinson splitters, the RF switches or the Schottky diodes to achieve the required goals. The measurements of the Level 1 trigger (which also required a complex test bench developed by myself) show that the design described in this chapter fulfils the requirements in terms of gain, linearity, noise, latency and jitter, both for majority and sum trigger strategies. These results are essential to get the optimum sensitivity and performance in the final telescope. Additionally, the Level 1 system presented in this chapter is the first one able to change the size of the trigger regions by slow control ever done, and the development of the LVDS OR gate have resulted in a patent for generic LVDS gates.

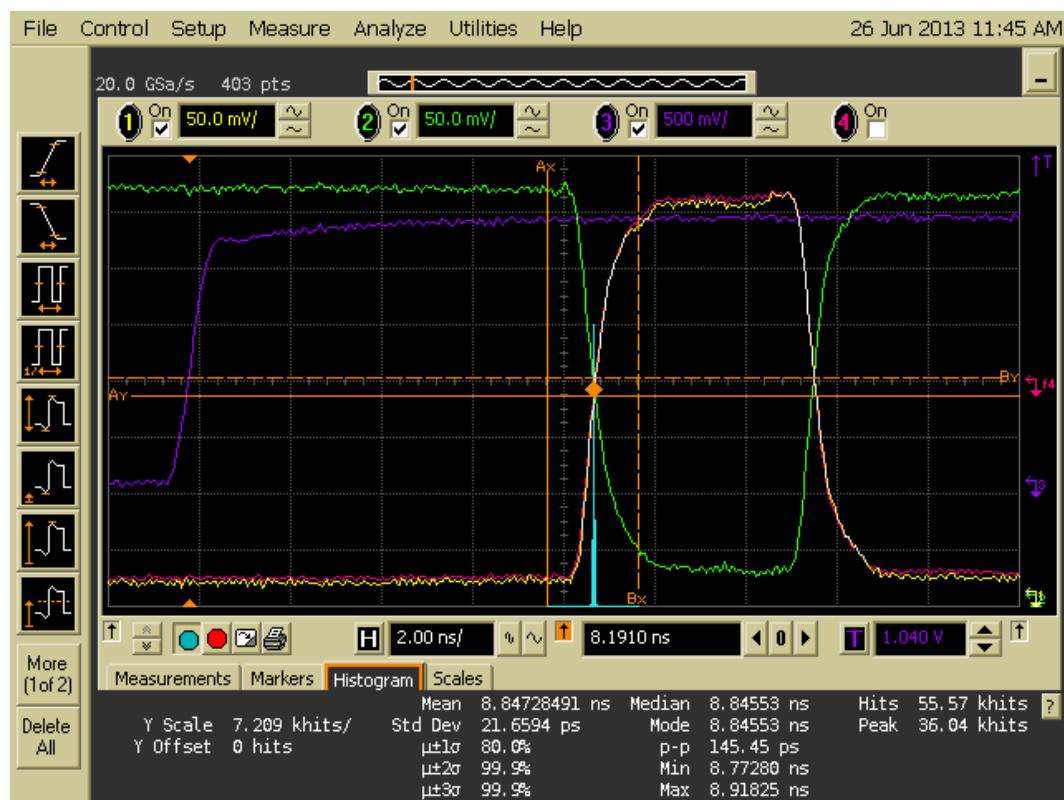


Figure 4.40: Jitter of the Level 1

Chapter 5

The COLIBRI

All IACT cameras developed before CTA work as fast intelligent-triggered photograph cameras: They capture the photons integrated in few tens of nanoseconds, in all the pixels of the camera at the same time, forming a static image. However, this approach has two important limitations, namely regarding the volume of data finally stored on disk and the efficiency to recover the photons of the shower image¹. The COLIBRI (Concept for an Optimized Local Image Building and Readout Infrastructure [8]) proposes an alternative concept, in which only those parts of the camera likely to actually contain image data (usually a small percentage) are readout, reducing the amount of data taken. Additionally, it takes advantage of the modular design of the cameras with quasi-independent clusters to readout different parts of the camera at slightly different times, thus allowing the readout to follow the development of the high-energy γ -ray shower images across the camera field of view, which can last up to tens of nanoseconds for images corresponding to showers above 10 TeV. This new concept and its advantages, briefly described in sections 5.1 to 5.5, were proposed and described in detail by C.L. Naumann in [8], together with its expected performance according to simulations. On the other hand, the hardware implementation of COLIBRI integrated in the trigger system described in section 3.2, was developed by G. Martínez and the author of this thesis, presented also in [8] together with the concept, and it will be described in section 5.6.

5.1 Partial readout

The long but narrow three-dimensional Cherenkov showers are mapped onto an elongated ellipse in the telescope camera, which typically covers only a small part of the camera (except for the highest energies). On average, not more than a few percent of the camera pixels are hit by the Cherenkov light; the rest of the camera only records random noise. As simulations of shower images in CTA have shown, the average shower size does not exceed about 5%² of the camera pixels, even for energies of several TeV. In a classical readout scheme, the whole camera would be read out for each event, regardless of the actual size of the relevant shower image. Although most of the pixels would contain only noise, and would be suppressed during data analysis, their readout creates a

¹In the literature, as well as in [8] and sometimes in this chapter, it is common to speak about “charge recovery” referring to the charge produced by the PMTs when detecting photons (photoelectrons) instead of referring to the photons detected, but they are equivalent.

²The exact value depends on the field of view of the telescope

significant data volume to be transferred out of the camera. When there are many tens of cameras triggering at several kHz rate the amount of data becomes difficult to handle, and storing a large proportion of noisy pixels incurs on important overcost. In addition, due to the intrinsic dead-time of the readout electronics, the noisy pixels will be also unavailable for further reading during this time.

Thus, by concentrating the read-out only on those parts of the camera that actually contain physical image information, the camera data rate for a given trigger rate could be significantly reduced (by a factor five to ten, allowing a safety margin around the edges of the image), without losing any relevant image information if the thresholds are properly selected, as will be shown in section 5.5. There are other possibilities to reduce the amount of data by using fast data processing units performing loss free data reduction or zero suppression algorithms before storing them. However, in the case of partial readout, as only relatively small parts of the camera would be involved in readout and digitisation, and thus affected by dead time, the rest of the camera would, at least in principle, be available for detection and readout of additional events registered directly after the first event³. The reduction in both the data rate and the fraction of clusters suffering dead time, could then in principle allow the telescopes to be operated at a higher trigger rate, with a lower trigger threshold than normally used, thus possibly lowering the energy threshold of the whole telescope system. This reduction of bandwidth and dead time will be most important for the large and medium sized telescopes, which have a large number of pixels, will be operated very close to the minimum threshold and will, in the case of the LSTs, record all the samples of each pulse of each pixel.

5.2 Sliding readout window

Another advantage of partial camera read-out is its ability to readout different parts of the camera at slightly different times, thus allowing the readout to follow the development of the images across the camera plane.

In conventional readout modes, all of the camera pixels are read out in exactly the same time window around the global camera trigger, and the charge of each pixel is integrated over this time interval. This assumes that the local charge maximum appears at roughly the same time in all pixels of the camera. However, Monte-Carlo studies, as well as the analysis of recorded data from the H.E.S.S. telescope, have shown that this is not always the case. In the case of Cherenkov showers caused by high-energy γ -rays or hadrons, the development of the shower images in the camera can be far from instantaneous, with the photon arrival times spread out over several hundreds of nanoseconds in the most extreme cases. For instance, figure 5.1(a) shows the average total duration of a shower in the camera and figure 5.1(b) illustrates how different pixels receive the photons from a Cherenkov shower generated by a high-energy proton at different times. As it is shown, the showers illuminate only a small fraction of close pixels at the same time, while the illuminated region moves through the camera plane during some tens of nanoseconds.

If the whole camera is read out in a fixed time window defined by the first triggers in the camera, only the earliest parts of the developing shower image will be captured. On the contrary, all parts of the image that arrive only after this readout window will be lost systematically truncating the late “tail” of the shower images. Thus, not only is the total registered image charge reduced but also

³A system able to accept triggers while part of the clusters are digitizing is theoretically possible, but extremely difficult to implement in practice.

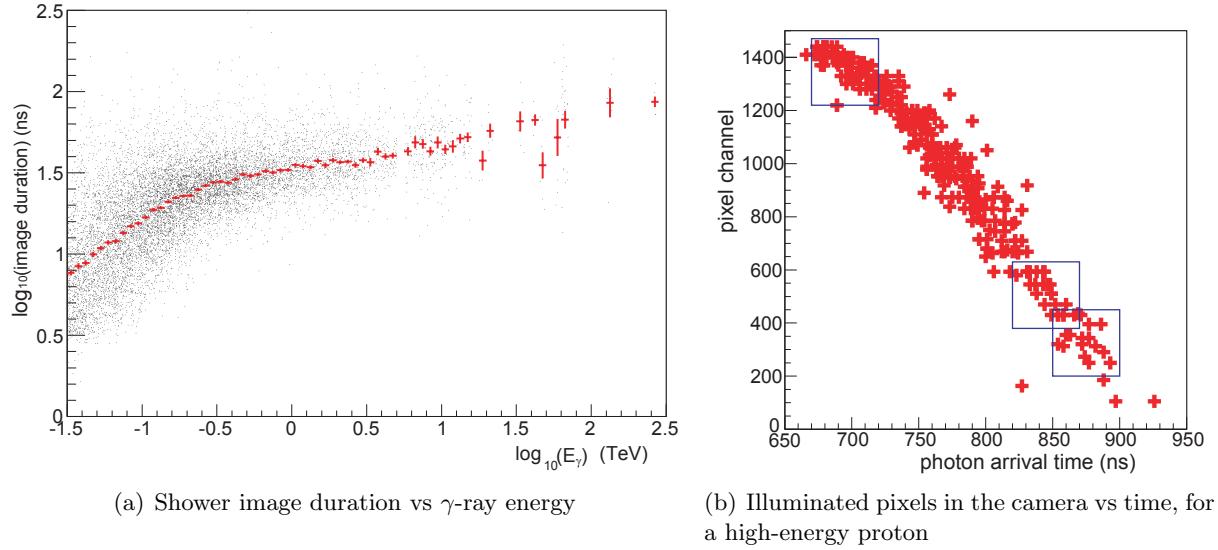


Figure 5.1: Monte Carlo simulations of duration and time development of showers in the camera.

important image shape information is lost, as a significant part of the image remains unread. This leads to distorted images, which will be more difficult to reconstruct properly.

To avoid this loss of image information, the effective length of the readout window has to be increased. If this were done for all pixels at the same time, however, this would lead to an unacceptable amount of noise (figure 5.2(b)) and digitizing more samples would mean a significantly increment of the readout dead-time for each read-out. What is necessary, instead, is to let the camera readout follow the image development by reading locally only around the time when there is actually image charge present (the so-called sliding-window readout). Again, this can be done in a quite natural way, if a partial readout scheme is adopted; in this case, the total camera image will be composed of a series of small, locally constrained subimages taken at slightly different times, so that the total readout window covers the whole shower development while the local readout window of each pixel is much shorter (figure 5.2(c)).

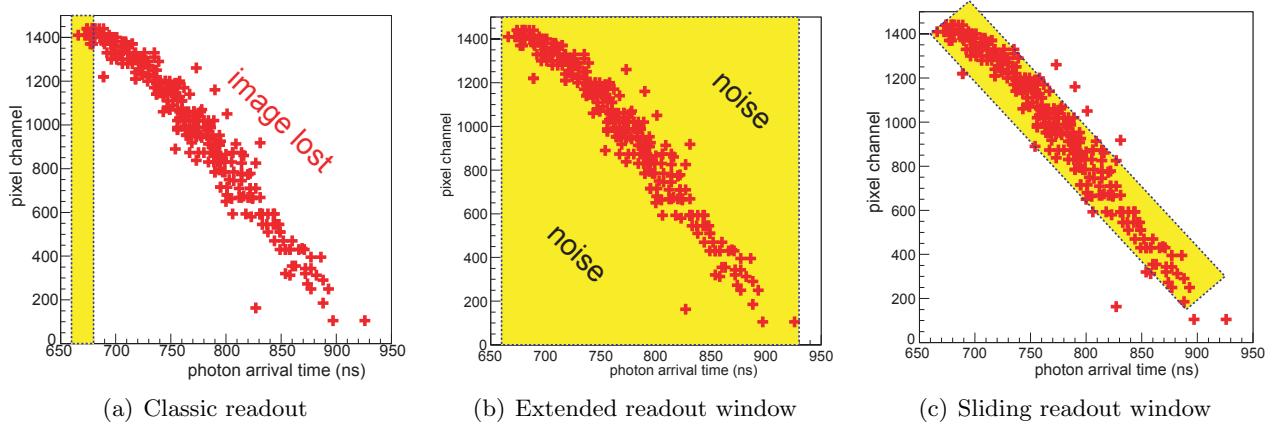


Figure 5.2: Sketch of different readout schemes for a slowly developing image

This sliding readout will be most useful for highly-energetic events and make it possible not only to improve the reconstruction errors for those events but also retain a sizeable portion of those events that would normally have been impossible to reconstruct due to truncation. This could then in turn increase the effective reach of each telescope and thus the energy range of the telescope array. For this reason, it will be most important for the small size telescopes operating in the high-energy regime.

5.3 Localized triggering and readout with a single threshold

In the standard readout scheme proposed for CTA, as described in chapter 3, a trigger from the Analog Trigger System always forces the readout of the whole camera. Then, the local data acquisition is stopped and the readout of the buffered data is started around the time of the original trigger signal. As discussed above, the majority of these channels will not contain any significant data from the shower image, and the local readout windows are not necessarily centred on the real distribution of charge in each channel. To avoid this, the presence of local triggers can be used to define local readout regions (so that readout will only happen if there has been a shower image in this part of the camera). This local trigger gives the “core part” of the shower image. Surrounding clusters can be included as well, even though they are themselves below the trigger threshold, to retain as much shower information as possible avoiding image truncation. How many neighbouring “layers” to retain will depend on a trade-off between image fidelity and readout efficiency. The triggering cluster and its neighbours constitute a single readout region. The data from their pixels is read out from the analogue memory around the time of the local trigger, in order to get the best-possible signal to noise ratio. A current local time stamp of each cluster should be stored together with its data, allowing a reconstruction in space and time of the complete camera image from the sum of all readout regions (effectively creating a shower “video” instead of a still image). Larger shower images from more highly-energetic showers will produce several local triggers. Each of those defines its own readout region, which will then be read out around its own local trigger time. That way, the readout will automatically follow the development of slower shower images (sliding readout).

After the readout has started, the affected clusters will be occupied for a certain dead time; as the readout of each cluster is independent, however, the rest of the camera will be unaffected. So, at least in principle, a different part of the camera could trigger and be read out while the first one is still occupied. In case of overlap, this can potentially lead to truncated events, which should be identified as such and could be discarded later.

While this single-threshold solution offers a simple way for local readout control, it has several practical drawbacks. One issue is that the lack of central control means that there is no obvious a-priori way to define which sub-events belong to the same event. This has to be done off-line, based on the individual time stamps of the readout regions and, even in this case it is still not clear how to make this compatible with an array trigger scheme.

Another important problem of using a single threshold for defining local readout regions and triggering the readout, is the influence of random NSB triggers and the limitation this sets on the trigger thresholds. The random Level 1 trigger rate depends strongly on the Level 0 and Level 1 thresholds. To avoid excessively high local triggering rates, the thresholds will have to be set accordingly high. However, if the Level 1 threshold is chosen too high, each low-energy event will

likely produce only one or few triggers along its event, close to the densest “core” of the image. Low-intensity “tails”, which lie below the threshold, will not trigger and thus not contribute to the creation of local readout regions. This will result in severe image truncation (strongly degrading gamma/hadron separation), unless the readout regions are artificially enlarged to several clusters beyond the triggering cluster. In this case, in the readout efficiency would be sacrificed (by including many empty pixels) and the possibility of sliding readout would be greatly reduced.

Thus, a simple single-threshold mechanism cannot at the same time provide good image coverage in space and time and high readout efficiency. There is an actual trade-off between data reduction and image truncation which is difficult to solve.

5.4 Two-threshold trigger and readout

As a possible alternative, a 2-threshold system is proposed: here, one uses a low threshold (“ L ”) only for the definition of readout regions, while the actual camera trigger and readout decision depends on a second, higher threshold (“ H ”). In that case, the camera triggering rate will only depend on the higher threshold, while the size and shape of the readout region depend on the low threshold. The low threshold can then be set to very low values, thus following the image shape more efficiently.

In this way, a local trigger at the L level defines a local readout region and starts its readout, but only if there has been a global trigger at the higher H level somewhere in the camera which can be associated with the same physical event. In this sense, the Level 1 with the higher threshold is now used as a global triggering system such as in the “classical” case, only it does not itself initiate any readout in the camera but instead enables localized readout for a limited period of time.

Once a trigger on the H level happens somewhere in the camera, it has to be distributed to all the clusters in the camera via the Level 1 distribution system. On arrival, this signal will set an enable state in each cluster, which is valid for a certain limited (and programmable) time, during which the local cluster is primed to process local L triggers. At the same time, the high level trigger can be used to define the presence of an event in the camera: the high level trigger will be used in the array trigger level and all clusters that are read out during the enable validity time will contribute to one and the same event. Using the internal ring buffer, the local readout window for each cluster can be chosen to correspond to a time interval around the occurrence of its local L trigger, not the time of the global H trigger. Therefore, the readout will naturally follow the development of the image inside the camera, as any region will only be read as soon as a local trigger has happened (within the validity period of the global enable).

In the absence of a global enable via the high-threshold system, local triggers will not initiate readout and will be ignored. In particular, NSB-induced local coincidences on or above the L level but below the H level will not contribute to the trigger nor data rate of the camera. Thus, the triggering rate of the camera is essentially independent from the L threshold, and local image building and NSB suppression are controlled separately.

5.4.1 Timing considerations

The high-level trigger has to be transmitted through the whole camera, incurring a significant but well-known and fixed latency of the order of 170 ns if the telescope is working alone or of the

order of 2500 ns if it is working in hardware stereo mode (see chapter 7), while the local distribution of the L trigger is much faster. Therefore, a local low level trigger signal would arrive at the clusters in the readout region before any enable state associated with the same event could be set, and thus be dropped by the readout system. Also, the first L triggers in the camera are likely to happen before the first triggers at the H level.

To circumvent this problem, it is necessary to introduce an artificial delay into the L trigger path to compensate for the latency of the high level triggers. In that way, while the L triggers still happen before the H signal returns and the enable state is set, they will only be processed afterwards, during the validity time of the enable. This setup is described in figure 5.3, where a shower image moving from left to right produces triggers in three different regions, one after the other. In all three regions, the low-level threshold is surpassed, and a trigger is created in the L channel (bottom of the figure). Those are not immediately processed but delayed by a programmable delay Δt_L . Additionally, in this example, the second and third regions trigger the H threshold. The resulting enable signal is transmitted via the H system (top of the figure) with a fixed delay Δt_H , after which the internal enable state is set in each cluster. As $\Delta t_L > \Delta t_H$, all local triggers are evaluated during the validity time of the enable signal, and all clusters in the readout regions are read out.

The enable validity time, T_{enable} , governs how long after the last global trigger the local triggers will be accepted to be included in the same event. The optimum time will have to be optimized by Monte Carlo studies, but, to allow sliding readout of faint events without multiple triggers, it should be at least a few times the local readout window length. Thus, for this study, a value of 100 ns was used as a good starting point. An upper limit is given by the necessity to reduce the contamination by random triggers from NSB that happen during the enable validity period. This constraint is not very stringent, however, as for example in the case of a validity period of 100 ns, even a random L trigger rate of the order of 100 kHz would only produce a single “fake” readout region in a few percent of events. Accidental “chaining” of H events (where a single physical or fake event is prolonged by random H triggers, should be completely negligible.

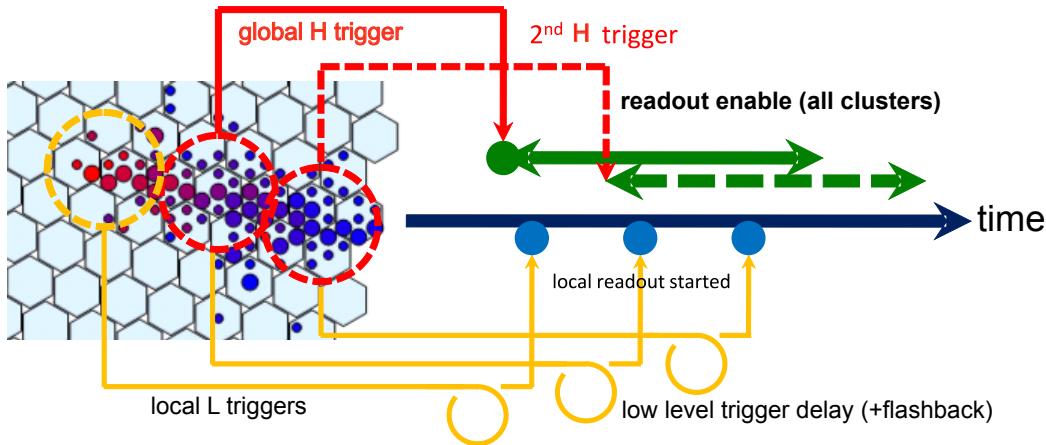


Figure 5.3: Two threshold timing scheme: Yellow paths represent the additional delay added to the local L triggers to make them fall inside the enabled window initiated by the H trigger (red). By making this delay longer than the propagation latency of the global H trigger it is possible to read also the L triggers which happened just before the H level was exceeded (flashback). Additionally, the enabled window can be extended if the H level is exceeded again inside the first enabled window.

The flexibility of this programmable delay together with the readout enable tag can be shown in two additional examples:

In the case of long developing shower images, the local triggers in the “later” parts of the image could lie outside the validity period of the first H trigger of the event. If, however, there are additional H triggers in the camera (here shown in the third region) happening during the validity period of the first, the arriving H enable signals reset the enable state of each cluster, effectively prolonging the total readout validity time of the camera. In this way, even very slow showers can be fully covered (as long as they produce several H triggers), even if the enable validity time for each H trigger is shorter than the total event length.

As shown in figure 5.3, it is also possible to include low-level triggers that happen before the first H trigger by simply increasing the L trigger delay: any L triggers that happen inside this “flash-back” time before the high level event will still be processed during its enable validity and thus be contained in the readout event. In this way, $\Delta t_L = \Delta t_H + \Delta t_{flashback}$, increasing the required analog memory depth. This effect can become more important for slow showers of low intensity in the camera, such as very distant high-energy events that are just above the trigger threshold. In those cases, a classical readout would cut out the early parts of the shower, creating a strongly truncated image. Even for lower-energy events this flash-back will not degrade the image quality, as the probability of an uncorrelated L trigger happening during the flashback time is negligible.

5.5 Expected performance

The expected performance of a two threshold flexible readout system (hereinafter COLIBRI) inside a CTA telescope has been evaluated by Monte-Carlo studies using simulated electromagnetic and hadronic showers and NSB together with a detailed simulation of the triggering and readout hardware. For the simulation of the physical showers, as well as the generation, propagation and detection of the Cherenkov light, the standard CTA simulation package [131] was used, with some modifications to preserve independent trigger information from the individual clusters.

To study the size distribution of gamma and hadron events in the camera, and the resulting readout region size, a large sample of events of both types was generated, with energies ranging from 30 GeV to 30 TeV. The triggering scheme was a majority one, with 4 pixels out of 28 at a level of 4.0 photoelectrons for triggering (H) and 2 of 28 for local image building (L).

Figure 5.4 shows two typical events in an MST camera: a 1.4 TeV γ -ray and a 12 TeV proton. The γ -ray image is very compact, contained in a slim slice of the camera. The corresponding readout regions, defined by the 7-module clusters triggered by the above mentioned COLIBRI two-level scheme, reproduces this shape very well, containing nearly all of the image pixels and some safety margin around the image, within only 15 % of the camera. Of particular importance for the hadronic event, all outliers and subshowers are also retained in the image, facilitating γ -hadron separation.

The efficiency of the proposed scheme to capture most of the image can be seen even better in figure 5.5. Especially for energies below a few TeV, the average image size in the camera is only of the order of a few percent of the total camera size; but even at the highest energies, the average size does not exceed 5-10% (although in individual cases it can be much larger). The red points show the average readout region size when reading only the triggering clusters and the direct neighbours.

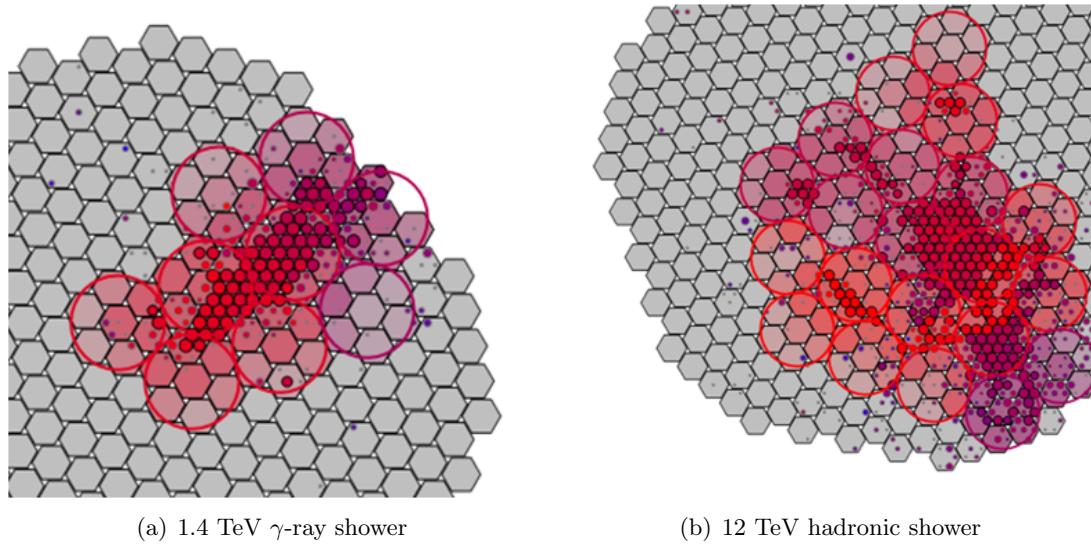


Figure 5.4: Simulated showers and readout regions for an MST camera

Thus, the mean readout region size does not depend strongly on energy and stays below 10% of the camera size in this particular readout scheme. On the other hand, despite the average fraction of recovered image pixels is only around 80% at some energies, those pixels in the readout region contain more than 90% of the image charge.

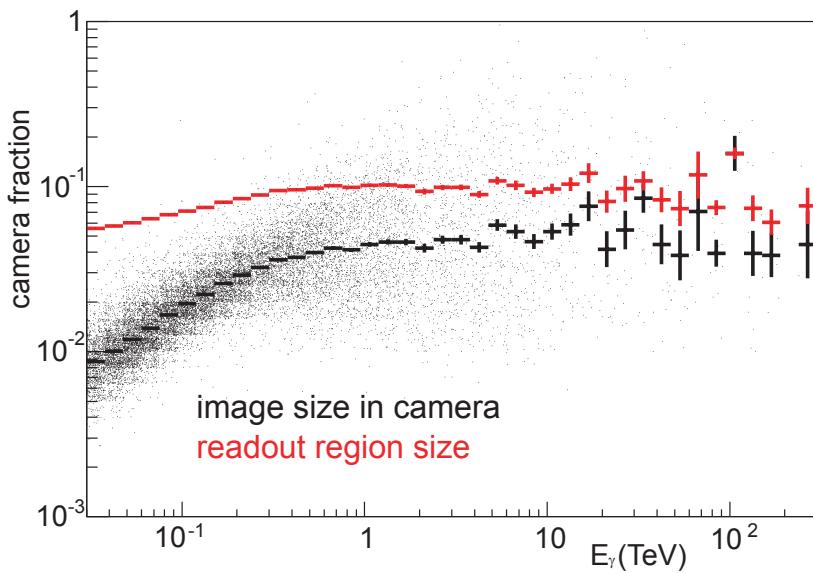


Figure 5.5: Image and readout region size vs energy

5.5.1 Comparison between different readout schemes

For partial readout modes, charge can be lost in two ways: From pixels outside the readout region (spatial cutting) and from photons arriving before or after the local readout window (temporal

cutting). While the former is specific for partial readout, the latter is specially important for “classical” readout modes. Due to the concentration of image charge in the pixels close to the shower core (which have a high probability of being contained in the readout regions), the fractional loss of image charge due to spatial cutting was consistently much smaller than the fraction of unread pixels containing at least some part of the image charge.

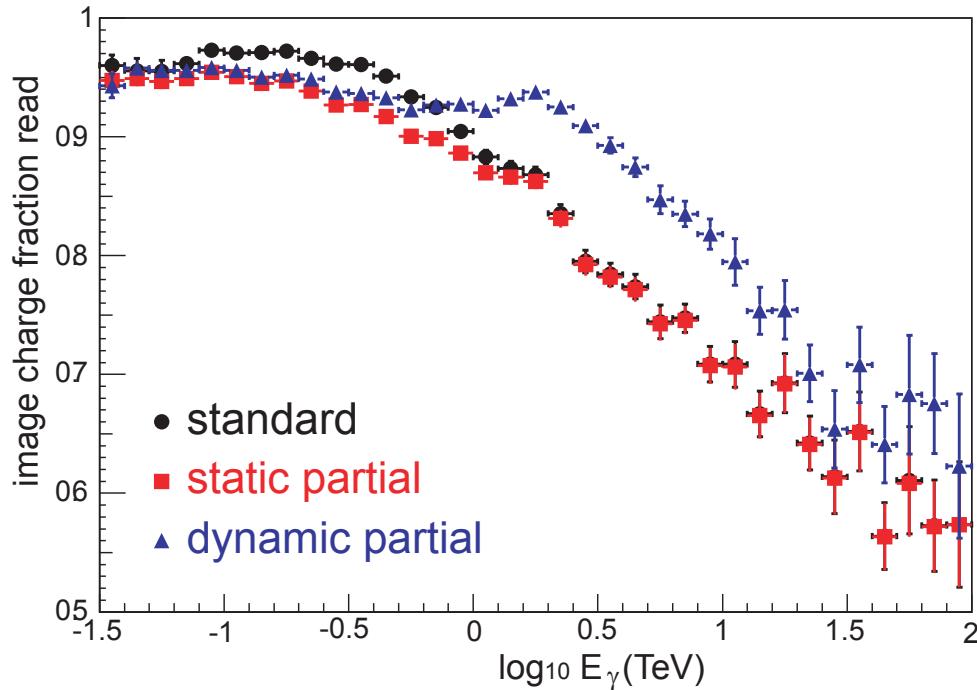


Figure 5.6: Comparison between different trigger and readout schemes, in terms of the charge fraction read

Figure 5.6 shows the mean image charge read out as a function of γ -ray energy when reading out the full camera (black), for a partial camera readout with a fixed time window around the global trigger time (red) and for partial camera readout with sliding window (blue). Looking at the plots the two charge loosing ways can be observed.

For energies lower than 1 TeV spatial cutting is the dominant effect, because low energy showers tend to create compact readout regions, and therefore charge from scattered light in outlying pixels is lost if a partial readout scheme is used. So, reading the full camera is the only way to recover the charge of the outliers and this is why it obtains better results in figure 5.6. Nevertheless it must be taken into account that low energy showers produce small images and using partial readout only 5% of the pixels would be read, with a great data reduction, only losing 2% of the charge of the event, with respect to the full camera readout method.

On the other hand, for energies higher than 1 TeV, it can be seen that the effect of spatial cutting losses importance because the proportion of charge in the shower core compared with the outliers is higher while, on the contrary, the temporal cutting of the showers becomes more noticeable. As has been explained previously in this chapter, for high energy showers the image in the camera develops longer, so the tail of the images is cut if a sliding window is not used. This is why the partial readout with sliding window achieves the best charge fraction read in this energy range.

It is also important to point out that software image cleaning algorithms are commonly used in the data analysis, and these algorithms usually suppress the photons in outlying pixels, at least in traditional algorithms [132]. If these photons are removed, then the slight advantage of the classical readout of the complete camera disappears. However, if the outliers are recorded, it is always possible to take them into account in the analysis by using a more complex algorithm [133].

5.5.2 Quality of reconstructed images

To assess the quality of the cut images and the effect of the different readout modes on a potential reconstructability with the whole array, a simple image reconstruction was performed for the single camera images. Here, a standard Hillas reconstruction algorithm was used [134], giving length, width and direction of the reconstructed ellipse as parameters for comparison. Apart from the reconstructed direction, one value of particular interest here was the elongation of the Hillas ellipse (length/width). As the temporal development of shower images follows the ellipse's major axis, events that are truncated in time will be artificially shortened, thus a high elongation is a sign of good image coverage. Also, for higher elongations the direction of the ellipse and thus of the shower is more clearly defined, thus longer elongations are normally correlated with better angular resolution at the array level (see figure 5.7).

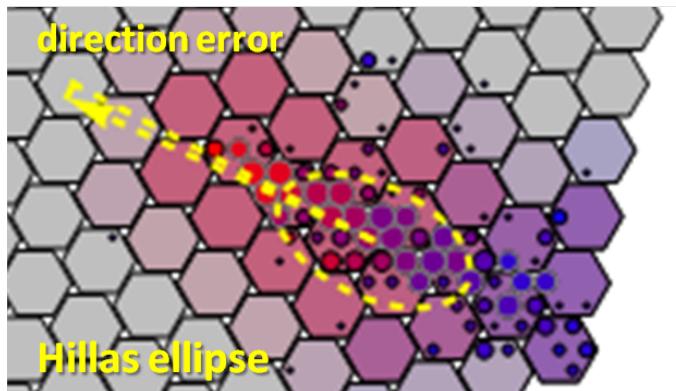


Figure 5.7: Simple Hillas reconstruction of a shower image

Figure 5.8 compares the resulting Hillas ellipse elongation as a function of energy for standard readout, COLIBRI and for the “ideal” image (after image cleaning cuts). In the sub-TeV region, all three modes are very similar and show a steady increment in the elongation of the Hillas ellipse as the energy increases. Above 1 TeV, however, this elongation starts to decrease again for the case of the standard readout, due to the loss of charge from late pixels at the tail of the shower image. This decreasing elongation makes a reliable reconstruction of the showers increasingly difficult. On the other hand, for the sliding readout this decrease sets in only at higher energies, and the elongations are always very similar to the case of full readout. Thus, any loss of image information here is not due to the readout scheme itself but due to other effects, such as the unavoidable clipping of very large shower images at the border of the camera. Also, the similarity between the curves for sliding and for full readout imply that any remaining charge loss for the sliding readout (mainly due to the arrival time distribution in the individual pixels becoming wider than the local readout windows) does not affect the image in a noticeable way.

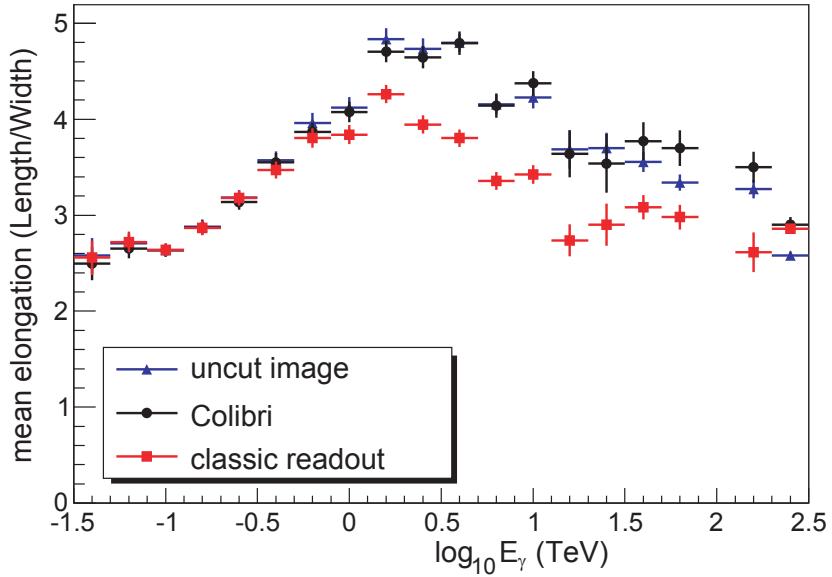


Figure 5.8: Ellipse elongation for different readout modes, as function of energy

5.5.3 Gamma-hadron separation

It is important also to study how the COLIBRI scheme affects the γ -hadron separation efficiency. The distinction is done by obtaining the Hillas parameters from the image and analyzing if they are compatible with a γ -ray shower or, on the contrary, they are more likely to correspond to an hadronic shower. If the shower was originated by a high energy γ -ray, the image quality is improved by using the COLIBRI, reducing the temporal cut and, at the end obtaining images with higher elongation which should be easier to recognize as a γ -ray originated. In principle this should make the γ -hadron separation simpler but, in fact, at high energies the γ -ray shower images are elliptical and with a high elongation even with the temporal cut, so current algorithms already have a very good performance. This means that the COLIBRI does not provide a large improvement in the γ -hadron separation, although it can be affirmed, without any doubt, that it does not make it more difficult.

On the other hand, for low energy γ -ray originated showers, the COLIBRI can have a harmful effect. As in every partial readout scheme, pixels with very weak signals may not be read if they don't exceed the low threshold. This could eliminate the outlier pixels typical from hadron shower images, which are an important characteristic to distinguish them from γ -ray images. This loss of information can be avoided by reducing the lower threshold to a very low value, but the lower the threshold the more pixels which would be read and the COLIBRI readout would tend to behave like a classic full camera readout. At the end, there will be always a trade-off between the sensitivity to very weak showers and the data reduction by partial camera readout. The optimal threshold will depend on the object under observation, the environmental conditions and the characteristics of the telescope. Nevertheless, the final trigger system which implements the COLIBRI scheme, should also allow to use a global camera readout scheme, in a slow-control selectable manner, so that each of the systems can be used depending on the observation constraints.

5.6 Hardware implementation

In order to allow the implementation of the COLIBRI scheme (which could be slow-control-selectable), some changes have been included in the trigger system presented in the previous chapters of these thesis. These changes affect the Level 1 and the Level 1 distribution subsystems.

5.6.1 Changes in the Level 1

The first requirement that the trigger system must accomplish to implement the COLIBRI scheme is to be able to generate two trigger signals at two different thresholds. This affects the Level 1 subsystem, whose block diagram can be seen in figure 5.9.

The main changes to implement COLIBRI are the following:

- The DAC have to generate two different threshold levels. This was solved by using an AD5663R DAC [116], which has two independent outputs controllable through the same 3-line SPI link. The threshold levels are controlled by the front-end FPGA using the same lines as in the first version of the Level 1, with only few changes in the firmware. Now, the messages to set up a precise output voltage contain an address field to refer to one or other output. Additionally, AD5663R has an internal reference voltage with very good stability. More information about this DAC can be found in section 4.8.
- Three more comparators are required. To implement the COLIBRI scheme, each of the three (or two when mode 3 is selected) additions must be compared with two thresholds, with two comparators per addition. This means a total of 6 fast comparators ADCMP604 [102] similar to the ones used in the original Level 1. Additionally, having two comparators at the output of each adder, the resistance values of the resistors present at the input of the comparators must change from $100\ \Omega$ to $200\ \Omega$ to maintain the optimal $100\ \Omega$ load at the output of the adders. This is essential to avoid reflections and get the largest bandwidth.
- Two OR gates are needed: One to check if any of the three additions is exceeding the higher threshold and the other to check if the lower threshold is being surpassed. These two OR gates generate the two trigger outputs required by the COLIBRI strategy.

These changes in the Level 1 subsystem mean a slight increment in cost ($\approx 6\%$) and power consumption ($\approx 21\%$) per cluster, but this is a small price compared to the potential benefits of the COLIBRI strategy.

Figure 5.10 shows a Level 1 mezzanine implementing COLIBRI changes. Several prototypes have already been tested, showing a similar performance to the one obtained in mezzanines without COLIBRI in terms of delay, gain, linearity and noise. In fact, the measurements of the Level 1 shown in chapters 3 and 4 correspond to the last version of the Level 1 mezzanine, including COLIBRI improvements.

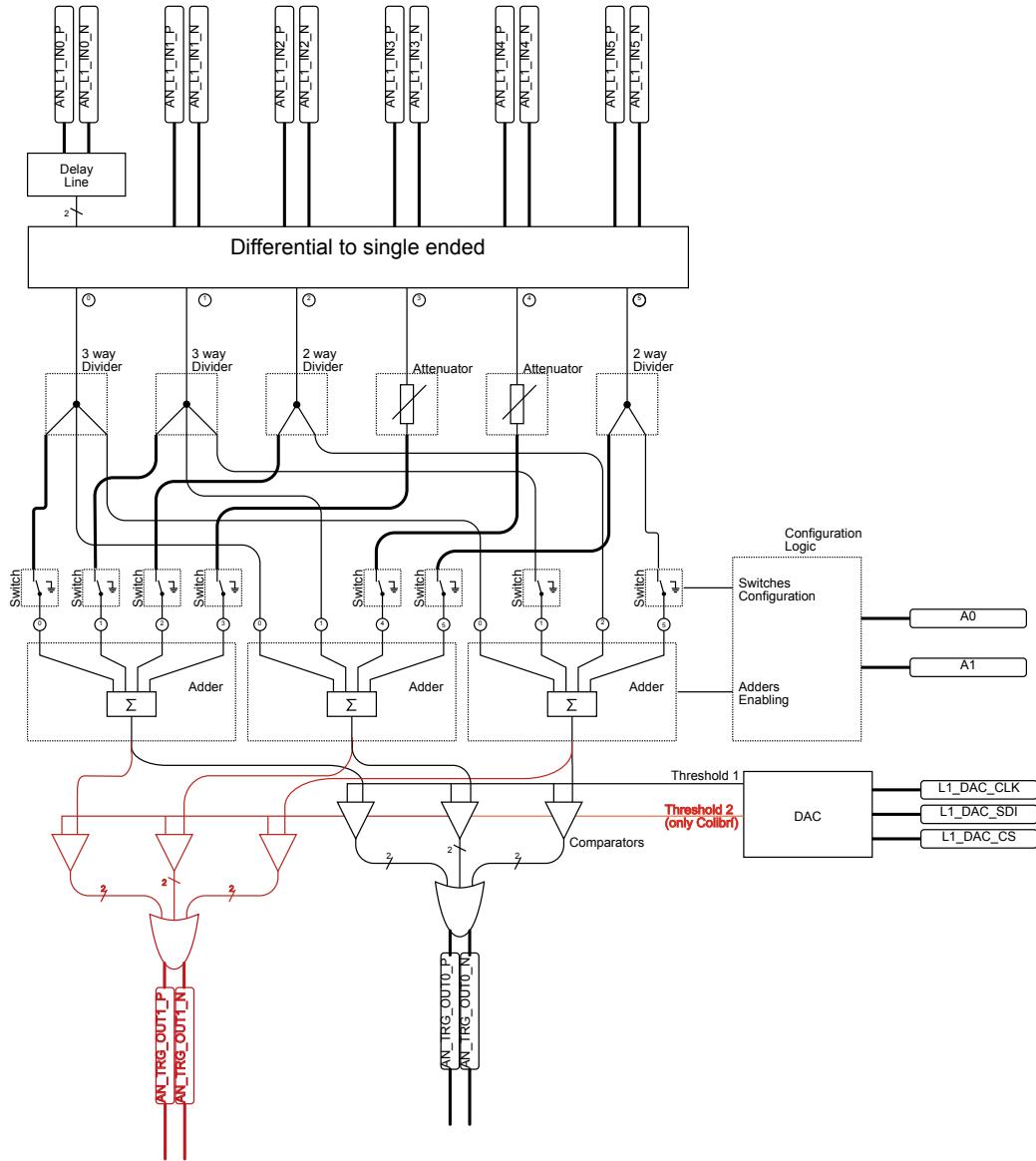
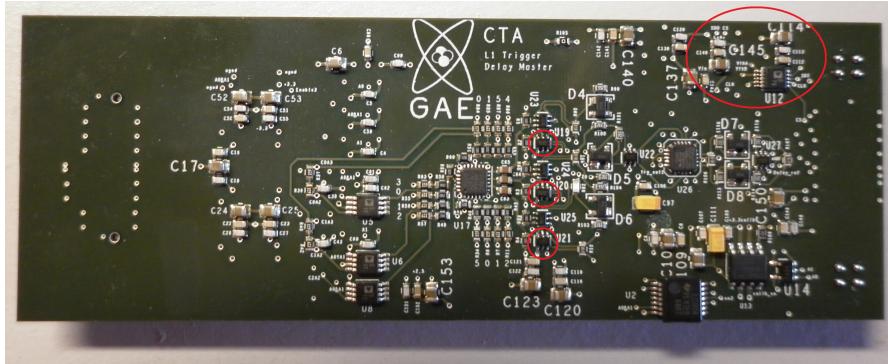


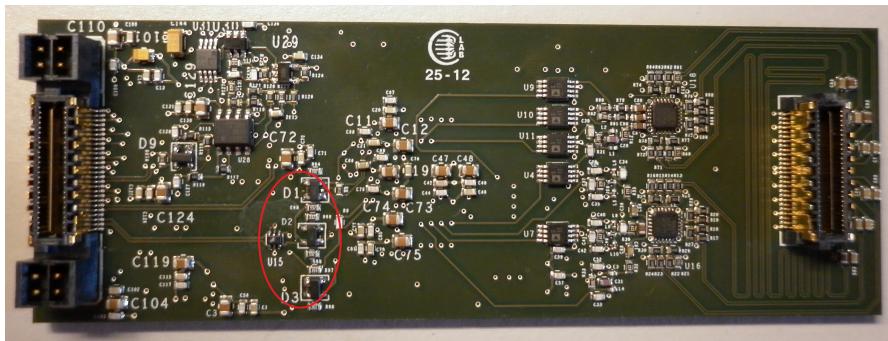
Figure 5.9: Level 1 scheme, with the modifications to implement COLIBRI in red

5.6.2 Changes in the Level 1 distribution

The other important changes required by the COLIBRI affect the Level 1 distribution, which now has to handle two trigger signals. If apart from the clusters exceeding the low threshold, also their neighbours want to be read, another two differential pairs between each two clusters is required. On the contrary, if only the clusters exceeding the thresholds will be read, no hardware modifications are required at all because the old distribution system can be used to distribute the high threshold trigger and the low threshold trigger would not need to be distributed. In order to be compatible with both options, the backplane includes lines enough in the strips between neighbour clusters to distribute the two triggers if required.



(a) Top



(b) Bottom

Figure 5.10: Photos of the Level 1 mezzanine, suitable to implement the COLIBRI scheme. The specific components added for Colibri are inside red circles.

In this way, the other changes are related with the firmware running in the backplane FPGA. The high threshold trigger would be distributed exactly in the same way as in the classical trigger, as described in section 3.2.6. The only difference is that, with COLIBRI, this trigger signal does not start the acquisition, but only opens the enable window, which does not require so much accuracy. With COLIBRI, an accuracy of a few nanoseconds in the distribution of the high threshold trigger would be enough.

Regarding the local low threshold trigger, it should be delayed in the backplane a time equivalent to the distribution latency of the high level trigger plus the flashback window. For short delays, of the order of 200 ns, the delays can be implemented in the FPGA in a similar way as the delays used to synchronize the arrival time of the distributed trigger among the clusters. However, for the higher delays required when using hardware stereoscopic techniques (more than 3 μ s) this way of implementing the delays could be not accurate enough.

Since the functionalities of the partial readout are controlled solely by the FPGAs of the distribution system, it will be possible to operate a COLIBRI-type camera in several different ways: In addition to the flexible, locally controlled readout, classical readout can be achieved by ignoring the low level trigger signals and instead using the high threshold trigger to initiate the readout in all clusters of the camera. As the local trigger information will still be available, this mode can be used to monitor the functioning of the COLIBRI system and to gather background data with minimum bias. Thus, a COLIBRI based camera can always be used in the same way as a classical

one, whenever necessary.

5.7 Summary and suitability of COLIBRI

As a summary, the COLIBRI readout provides with a powerful way to reduce the amount of data taken and improve the photon collection -and therefore the quality of the images- for showers generated by high energy γ -rays or hadrons. In this energy range it also improves, or at least it does not complicate, the γ -hadron separation. On the other hand, low energy γ -ray shower images do not benefit from the temporal sliding window and require the low thresholds to be extremely low in order to avoid loosing interesting information. And using so low thresholds means reading many pixels in the camera, including noisy ones and thus loosing in part the COLIBRI feature regarding data reduction, which still would be a better solution than reading out the whole camera.

These characteristics make the COLIBRI very suitable to be implemented in the CTA MSTs and SSTs. There will be several tens of these kinds of telescopes, so data reduction is important⁴, and in addition they would benefit from the improved images. On the contrary, the COLIBRI scheme would not mean an important advantage for the LSTs, where very weak signals in a short time window are under consideration. Nevertheless, as the same hardware can implement COLIBRI or classical trigger and readout schemes with no more than changing the firmware of the Level 1 distribution, both the trigger systems for LSTs and MSTs will be adapted for COLIBRI, in order to take advantage of the economy of scale of the MSTs. The trigger system of SSTs is out of the scope of this thesis.

⁴In addition, the wide field of view of these telescopes makes the shower images to cover smaller areas in the camera, so the reduction in the volume of data taken achieved with partial readout would be larger

Chapter 6

PMT Transit Time Delay Compensation

Both majority and sum trigger strategies are based on the fact that the light from a Cherenkov shower produced by a single primary particle (like a gamma-ray or a hadron) will be detected by several PMTs at nearly the same time. However, the time required by a PMT to generate the output pulse corresponding to an impinging Cherenkov photon -typically dubbed PMT transit time- depends strongly on the high voltage applied to the dynodes. These high voltages also determine the amplitude of the output pulses of the PMTs. Therefore, if the high voltages of different PMTs are selected with the aim of equalizing their gains throughout the camera¹, the PMTs will introduce different transit times, the additions or the coincidences required for the trigger logic will not happen properly and the trigger performance will be degraded.

For example, the transit time introduced by a PMT polarized with different high voltages can vary up to 8 ns as it is shown in figure 6.1 for a R11920 PMT from Hamamatsu, which is one of the candidate PMTs under consideration for LST and MSTs. In a realistic situation, it is not necessary to take into account the most extreme values because the PMTs are never biased at their maximum or minimum voltage, but even in this case a transit time variation of up to 6 ns can happen.

The most simple way of overcoming this problem is to select PMTs with a similar behaviour with respect to gain and transit time for the same trigger regions, but this strategy incurs a significant over cost. Moreover, after some years of operation, ageing effects can change the gain of the PMTs in an unpredictable way, so they would need to be recalibrated using different high voltages, therefore unequalizing the transit time differences.

6.1 Impact of the different transit times

To quantify the effect of the different transit times introduced by the PMTs for the sum and majority trigger strategies, the response of both trigger schemes in a Cherenkov telescope camera were simulated by F. Dubois [9]. The simulations were performed by using the standard CTA Monte Carlo simulation tools [131] and an additional software called *trigsim* [135] used to add the NSB

¹The process to equalize the gains of the PMTs is known as “flat-fielding”.

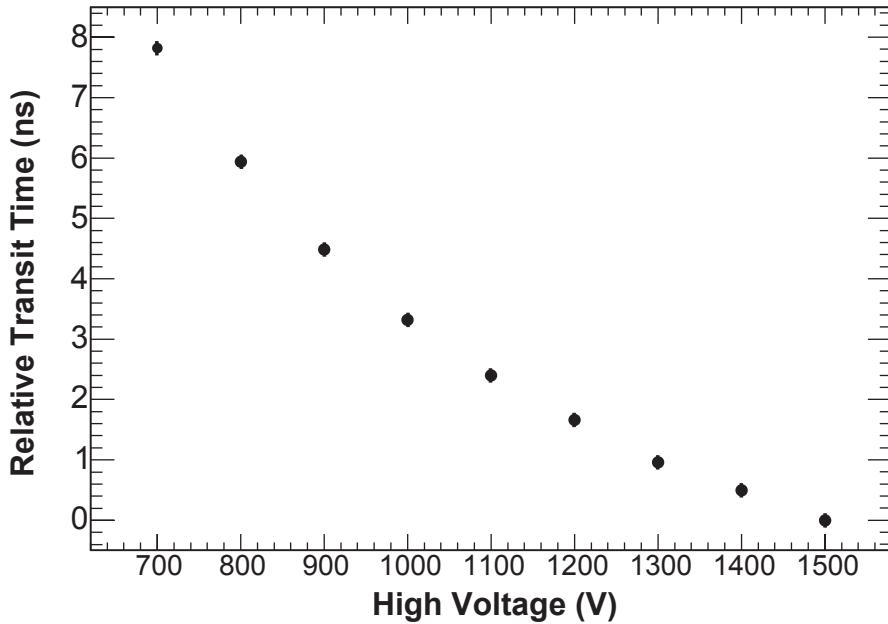


Figure 6.1: PMT relative transit time for a Hamamatsu R11920 PMT, as a function of the high voltage applied [9]

and simulate the trigger. In both cases a trigger region of 3 clusters (21 pixels) was considered, with the Level 1 threshold fixed to have a camera NSB accidental rate of 10 kHz, comparable to what is expected during observation. To simulate the different transit times of the PMT output pulses in the camera, the Cherenkov photon arrival time at each pixel was shifted by adding different sets of delays, gaussian-distributed with a 0 mean and a standard deviation ranging from 0.25 to 2 ns.

The results of these simulations are presented in figure 6.2, which shows the LST collection area ratios which come about after normalizing the effective area obtained with a given time shift, with respect to a non-shifted effective area. From the figure, one can see that the drop in effective area (i.e., in telescope performance) with an increasing transit time dispersion is very small for γ -rays with energies above 100 GeV, due to the brighter and broader shower images. However, for energies smaller than 100 GeV the effect becomes noticeable, and at 20 GeV (at the LST threshold) the drop in performance is close to 30% for a delay dispersion of 1 ns. Fortunately, reducing the transit time dispersion down to 250 ps or less provides performances closer than 5% with respect to ideal ones at 20 GeV. Additionally, it is shown that the drop in performance with increasing transit time dispersion is more pronounced in the sum trigger than in the majority trigger. This was expected, as the sum trigger relies on the analog sum of all analog pulses, and the different delays between the pulses can ruin the total sum.

The results of the simulations show that, in order to achieve the best performance in the low energy range, the transit times introduced by the PMTs must be compensated until equalizing them with an accuracy of at least 0.25 ns. The following sections describe two systems designed to compensate these delays.

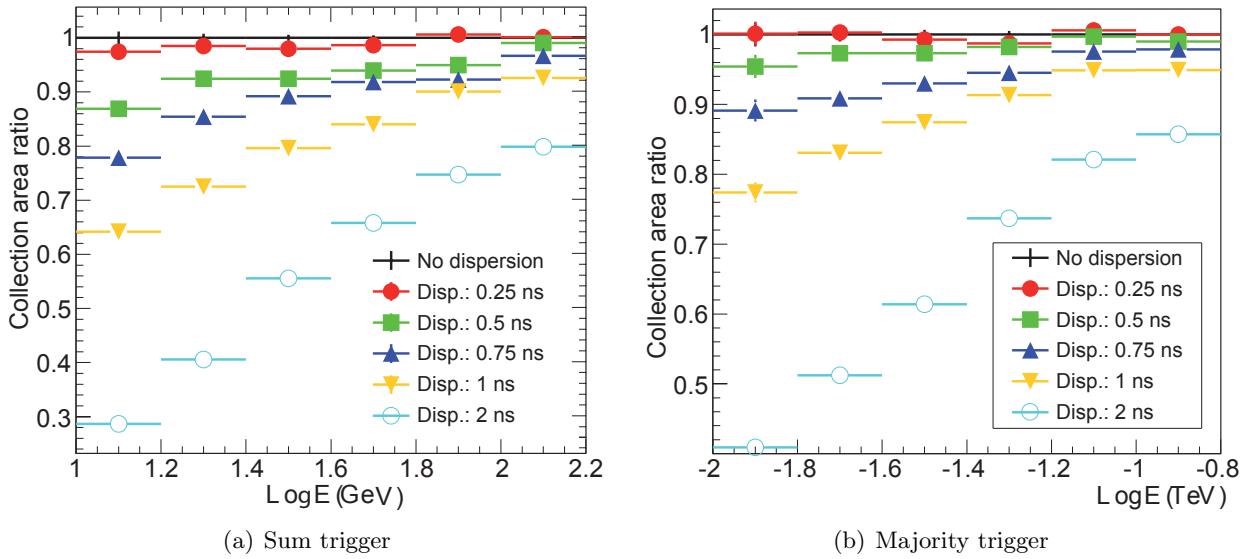


Figure 6.2: Ratio of effective area as a function of energy, for different maximum delay dispersion between pixels, with sum and majority trigger strategies[9]. The plots correspond to LST simulations, but for MSTs the effect is also noticeable.

6.2 Delay compensation with a reference

The delay dispersion of the PMTs can be compensated by adding a specific delay to each channel, which would be optimized in a calibration process. All the IACTs have calibration systems which can illuminate the whole camera with light pulses as short as few ns (simulating Cherenkov light), of approximately the same amplitude and arriving to all the pixels synchronously. Additionally this calibration box, as explained in section 3.2.1.7, can generate an electrical pulse corresponding to each light pulse. If this electrical pulse is distributed to all the camera clusters isochronously, arriving to all of them at a time close to the one at which the trigger corresponding to the light detection is expected, it is possible to measure the difference between the detected light and the reference. The delay differences would be caused by the different transit times, so setting a different delay in each pixel to align its trigger output to the electrical reference pulse would eliminate the dispersion.

The first requirement for this scheme to operate, is to be able to distribute the electrical reference to all the clusters of the camera at the time at which the Level 1 trigger due to the calibration light pulse is expected. If the calibration box knows very accurately when the light pulses will be generated, it can send the reference signal just at the time required to be distributed and arrive at the clusters at the correct time. The calibration light source is usually placed at the center of the reflector, so the time required by the electrical signal to reach the clusters is always longer than the time required by the light to reach the camera and the trigger to be generated. Therefore the electrical signal must be generated some time before the lights are switched on to arrive at the right time.

Sending the reference in these conditions can be relatively simple if the calibration light source consist of an array of fast LEDs as it is currently done in HESS [136]. This section describes a delay compensation system designed to work in these conditions and figure 6.3 shows a general diagram

of how it is implemented. However, if a powerful laser is used instead of the LEDs, there may be an uncertainty of up to $1\ \mu\text{s}$ in the time at which the light pulse is emitted, so that it will be impossible to generate the reference signal a fixed time before the laser is fired (because it is not possible to know *a priori* when it will be fired). A delay compensation system designed to work in this second situation is described in section 6.3.

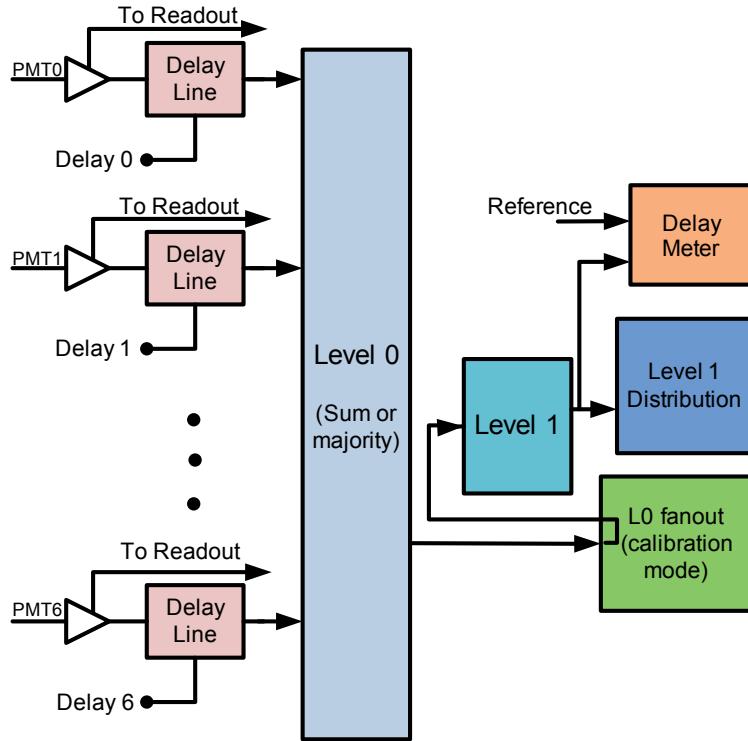


Figure 6.3: Delay compensation system architecture using a reference signal

Provided that there is an electrical pulse reference arriving to the camera at the right time, the next challenge is to be able to distribute this reference with an accuracy better than $250\ \text{ps}$ throughout the camera. Fortunately, the Level 1 distribution system does exactly this (see section 3.2.6). So, simply replicating the system which distributes the Level 1 trigger should be enough to distribute the timing reference precisely.

The second key element of the system is an adjustable delay line scheme. One delay line is required for each PMT and it must be placed before the Level 0 addition with the aim of adding up the signals properly. The technical aspects of the adjustable delay lines are described in section 6.2.1.

Once it is possible to adjust the delay of every PMT, another element is required to measure how well the detected pulses are aligned to the reference. This measurement must be done for every PMT. However, in order to reduce costs, in this implementation there is only one delay meter per cluster at the output of the Level 1 trigger subsystem.

The process of calibrating the delay of an individual pixel in each cluster is implemented as follows: first, all the pixels in the cluster except the one under calibration are fully attenuated by means of the attenuators and the clipping circuits implemented in the Level 0 subsystem (see section

3.2.3.2), so that the output signal of Level 0 only corresponds to that pixel. Then the Level 0 fan-out module in the backplane is set to calibration mode, so that the Level 0 output of a given cluster is simply fed into Level 1 of that cluster using a switch, instead of being replicated. Finally, the Level 1 compares its input with a threshold (configured high enough to avoid NSB triggers) and generates a trigger signal which is sent to a delay meter in order to check how well it is aligned with the reference. The output of the delay meter is read by a digital control implemented on an FPGA which varies the delay until the optimum value is found.

This process is repeated 7 times, for each individual pixel in a cluster. It is worth mentioning that this delay calibration process can be performed in parallel for all the clusters of the camera, so that the time required to calibrate the whole camera is essentially the same as that required for a single cluster. Therefore this algorithm will be somewhat faster than some other methods currently in use [25] and of course much more convenient than rearranging the PMTs [136].

6.2.1 Delay lines

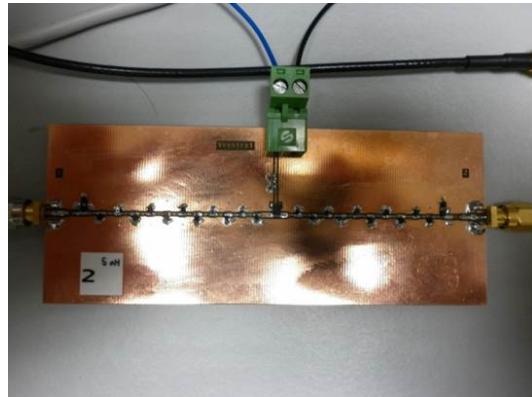
As it was shown in the simulations, an adjustable analog delay line capable of delaying the analog pulses up to 5.75 ns in steps of 250 ps is required for each pixel. There have been previous works to develop a similar system, like the one described in [137] for the MAGIC telescope. In that work the delay adjustment was implemented by means of a multi-step low pass filter with varactors which change their capacities with a DC voltage, thus allowing a very good delay accuracy in a small space. A prototype of a similar delay line was manufactured by the author of this thesis at UCM (figure 6.4(a), in order to assess this kind of lines. It had 23 steps with SMV1247 varactor diodes [138] and 10 nH inductors, connected as shown in figure 6.4(b)).

Unfortunately, the measurements of the prototype as well as the experiences in MAGIC [25] showed that it is difficult to obtain a flat attenuation response and good matching in a large bandwidth for a wide range of delays. The measurements of the prototype presented in figure 6.5 show how the output pulse can be greatly distorted when long delays are selected.

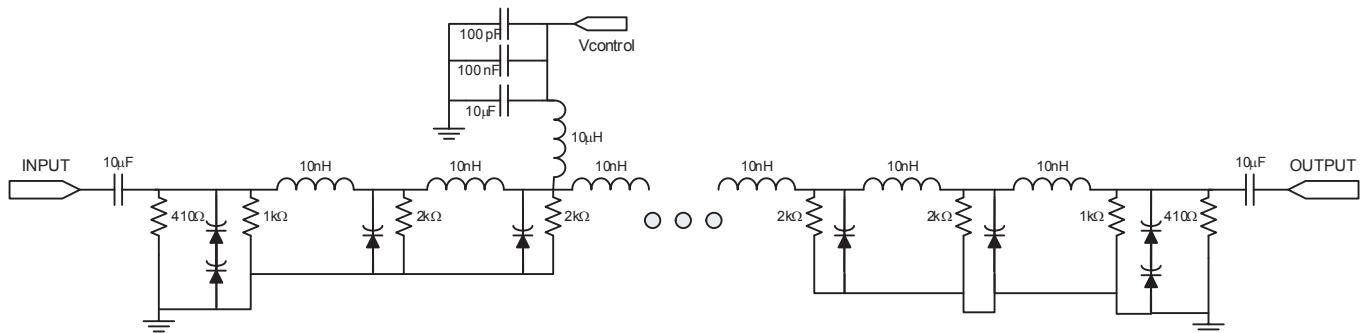
These poor results, together with the fact that simulations show that 250 ps accuracy is enough for a good trigger performance, led us to develop a different system based on lines of different lengths selectable with switches as shown in figure 6.6. In principle, the implementation of the delay lines as tracks in a PCB requires more space than the varactors, but it was greatly reduced by distributing the lines among different layers separated by ground planes in a multilayer board. The lines were designed carefully in order to have a controlled $50\ \Omega$ impedance and similar lengths in all the channels.

The delay lines, developed by the IFAE team, were integrated in the Level 0 sum trigger mezzanine before performing the addition. The level 0 receives a differential signal coming from the PMTs as an input, which is then processed by a differential amplifier [98]. As processing single-ended signals is much more convenient in order to perform the Level 0 operations, the first stage is a converter from differential to single-ended similar to the ones shown in figure 3.18(a), in chapter 3. The amplifier used for this conversion has enough fan-out to supply two $50\ \Omega$ matched lines. These lines are routed with a length difference of 2 ns between them. Thus, by choosing one of the two inputs coming to the first switch, it is possible to control a first delay step of 2 ns.

After the first switch there is a variable attenuator used to equalize the gain of the different channels. Then, two switches SP4T (single pole, 4 throw) and 4PST (4 pole, single throw) allow



(a) Photograph



(b) Schematic

Figure 6.4: Prototype of a delay line with varactors, manufactured and tested at UCM

the signal to be routed through one of 4 possible paths with a relative delay of 0, 0.5, 1 or 1.5 ns. The next two switches work in exactly the same way as the previous ones, but only a delay of 2 ns can be switched in this step. Finally, a clipping stage and an amplifier are used to clip the signal and adjust the gain if necessary (see section 3.2.7.4). This amplifier, like the first one, can drive two output signals which are routed with a 0.25 ns delay difference between them. The last switch allows the desired one to be selected. In total, a 5.75 ns delay can be controlled in steps of 250 ps.

The longer the delay, the larger the attenuation of the signal. However, by means of the variable attenuator, the different attenuations can be equalized, and the system can adjust the delay with virtually no gain degradation.

Figure 6.7 shows the measurements at the output of the adjustable delay line for a few different delay values, when there is a pulse of 2.6 ns width and 200 mV amplitude at the input. The pulse on the left-hand side corresponds to 0 ns, while the following ones are obtained after applying delays of 1.75, 4, 4.75 and 5.25 ns respectively. Many more delay values can be achieved by selecting different paths, but they have not been represented in figure 6.7 to maintain clarity. Anyway, it can be seen clearly that the pulse distortion is much lower than in the prototype with varactors.

In figure 6.8(a), the amplitude of the output pulses for several different delay combinations has been represented. It can be seen that the amplitudes are almost identical after adjusting the variable attenuator. However, in figure 6.8(b) the width of the pulses for different delays is shown to increase

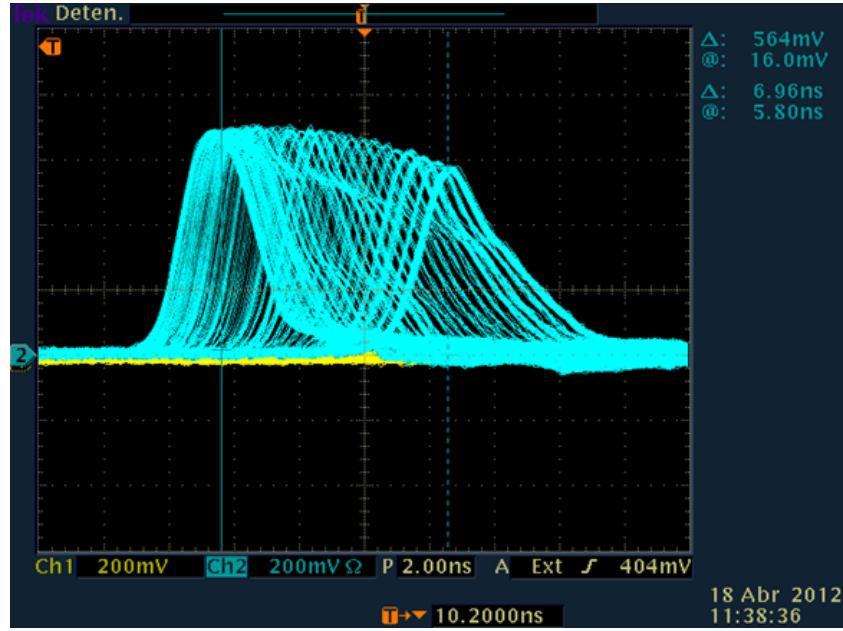


Figure 6.5: Output pulses from the prototype delay line with varactors, for different control voltages

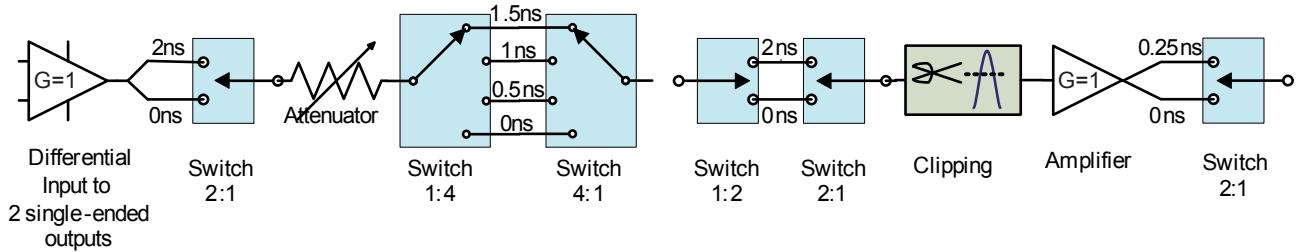


Figure 6.6: Scheme of the delay line based on selectable paths with different lengths

slightly with the delay up to 3 ns for the longest delays. It is difficult to reduce this undesired pulse widening due to the many stages that the signal has to go through. However, the effect of this widening is not expected to be very harmful for the trigger performance provided that the pulse width remains under 3 ns.

6.2.2 Delay meter

Once we can introduce variable delays into every pixel path, a delay meter is required to measure the time difference between the Level 1 output pulses and the time reference, and then minimize this difference. As it has to be sensitive enough to detect arrival time differences of about 250 ps, it is difficult to implement it directly in an FPGA or with current commercial chips. A complete TDC module can carry out this function, but the cost and space required are not affordable. Instead, a custom design has been developed by the author of this thesis, based on the idea represented in figure 6.9.

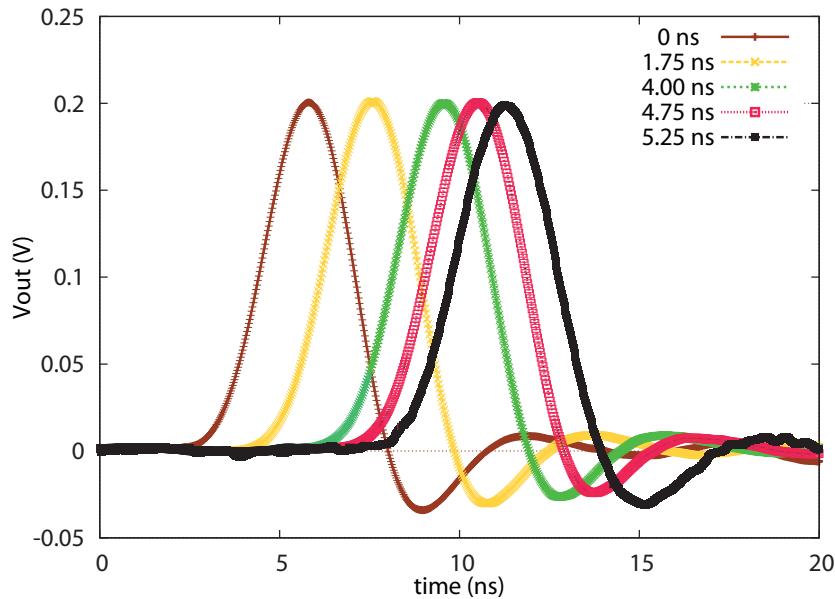


Figure 6.7: Measured pulses at the output of the delay lines with selectable paths

6.2.2.1 AND gate

The simplest function to know whether two active positive pulses are coincident is an AND gate. As both the reference and the output of the Level 1 are LVDS signals, we need a very fast LVDS AND gate which must only generate a positive output while the two pulses are overlapping, with an error of few picoseconds. Commercial gates to do this function fast enough and managing LVDS standard are scarce and expensive, but this is a perfect opportunity to use an AND gate of the differential logic family developed [7] and described in section 4.9. In this way, the AND gate corresponding to the schematic shown in figure 6.10, was designed.

The active components used are the same as those required for the OR gate of the Level 1: Avago HSMS2855 zero-bias Schottky diodes [123] and ADCMP604 comparators [102]. However, contrary to the OR gate, the resistors in the AND gate are actually needed to speed up the transition from “0” to “1” giving a low impedance path in parallel to the reversed diodes, while the transition from “1” to “0” is always fast, flowing the charge through active diodes. These means that, for the AND gate, the higher the resistance the shorter the pulse. Making the resistance lower, the pulse becomes wider until being as wide as the time during which the two inputs have a logic “1”². The highest resistance value which achieves output pulse widths as wide as the time during which the two inputs are set to “1” was 1.8 kΩ, and for that reason that was the chosen value. Selecting a lower value would increment the current flowing through the resistors when one input is set to “1” and the other is set to “0”, without adding any advantage.

²Obviously, it can not be wider.

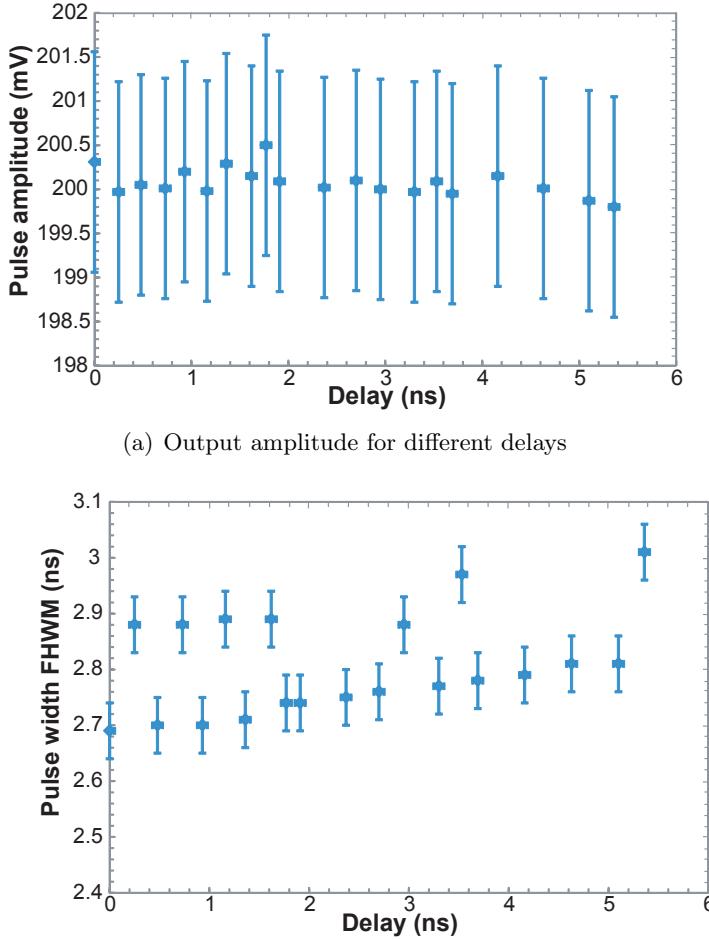


Figure 6.8: Characteristics of the output pulses for different delays

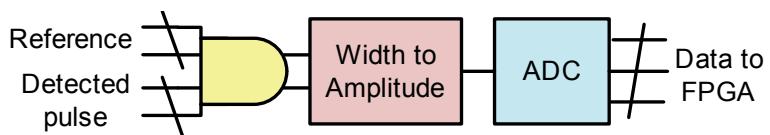


Figure 6.9: Delay meter general scheme [9].

6.2.2.2 Width to amplitude converter

The output from the AND gate is an LVDS pulse lasting between 0 ns when the signals do not overlap, and 3 ns for maximum overlapping time. As this time lapse is difficult to measure, the circuit in figure 6.11 has been used to transform the duration of the pulse into the amplitude of a 1 μ s constant width pulse, which is much easier to handle.

The diodes in the circuit in figure 6.11 are of the zero-bias Schottky type, as are those used in the

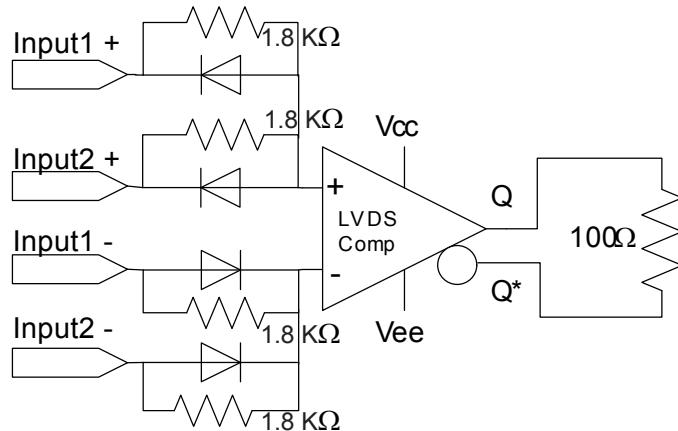


Figure 6.10: 2 inputs LVDS AND gate for the delay meter

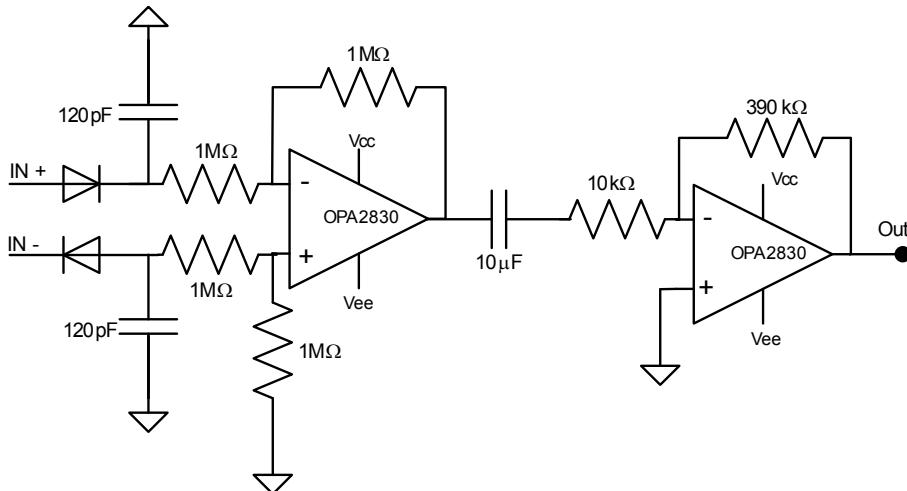
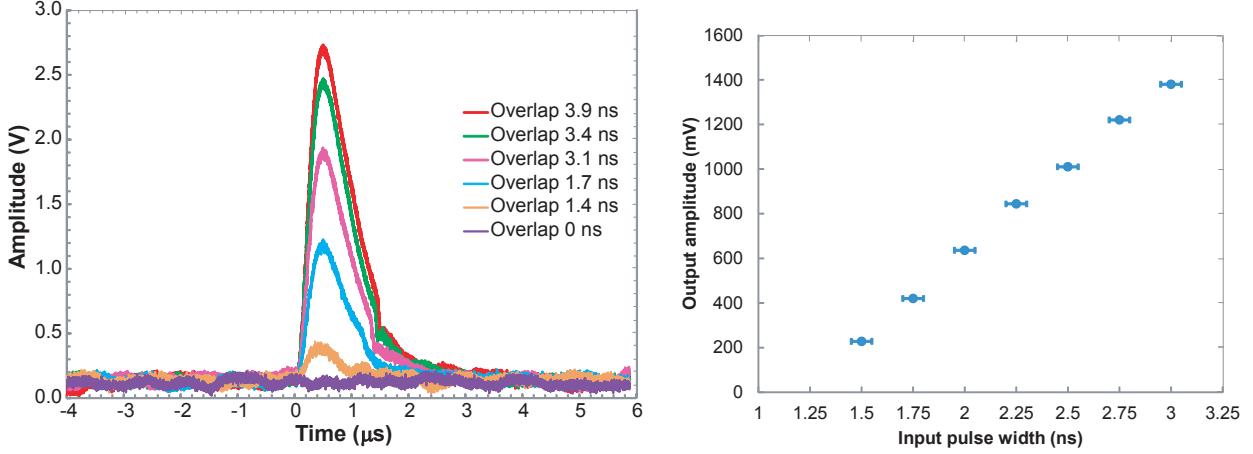


Figure 6.11: Schematic of the width to amplitude converter [9]

AND gate. When a positive differential LVDS pulse arrives, they let the capacitor in the positive branch get charged from 1 V to 1.4 V and the one in the negative branch gets discharged from 1.4 V to 1 V, in both cases through the small equivalent resistance of the diodes in the forward state. On the other hand, when the positive pulse finishes, the diodes change to reverse, showing a high resistance which does not let the capacitors get discharged (or charged, for the negative input) in a short time. In this way, a voltage level proportional to the length of the input pulse is retained in the capacitors, which get discharged slowly, forming a pulse of about $1 \mu\text{s}$ width.

Then, the positive pulse is subtracted from the negative one removing the offset of 1V typical from LVDS standard and the common mode noise, thus generating a larger and less noisy single-ended pulse. The input impedance of the subtractor is designed to be very high compared with the one of the diodes in reverse, so it does not contribute to reduce the discharge time. Finally, the large output pulse from the subtractor is inverted and amplified in the output amplifying stage, obtaining output pulses with amplitudes proportional to the input widths as shown in figure 6.12(a). With this scheme it is possible to obtain a sensitivity of 768 mV/ns, with a very linear response, as can

be seen in figure 6.12(b).



(a) Pulses at the output of the width to amplitude converter, (b) Output amplitude as a function of input pulse for different overlapping times. 4 ns width pulses have been used to characterize the full operating range, although real pulses generated by the PMTs do not last more than 3 ns

Figure 6.12: Measurements of the width to amplitude converter

6.2.2.3 Sample, hold and ADC reading

The amplitude of the pulse generated in the width to amplitude converter needs to be measured with an ADC. The problem is that the ADC needs to receive the digitizing request at the precise time to capture the voltage at the peak of the pulse. The FPGA in the front-end board that controls the delay system could solve this problem by sending the digitizing command a short time after the time reference arrives at the front-end board, but this would demand high clock frequencies in the FPGA. On top of that, any jitter in the generation of the digitizing request could produce sizeable errors in the amplitude measurement. Therefore, in order to make the digitization timing less challenging, a cheap and simple auto-triggered sample and hold circuit has been placed in front of the ADC, according to the scheme in figure 6.13.

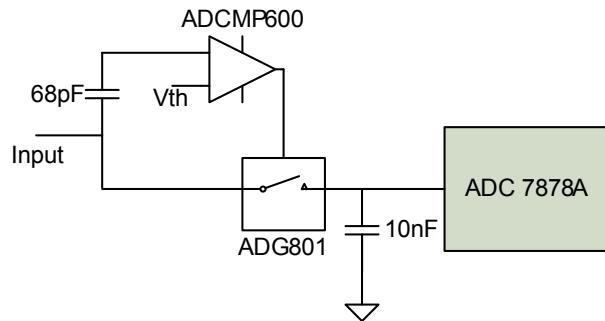


Figure 6.13: Block diagram of the sample, hold and ADC circuit

The way it works is very simple. The 68 pF capacitor differentiates the 1 μs input pulse (figure 6.14(a)) generating a fast positive peak on the leading edge of the input (figure 6.14(b)). This

narrow (≈ 300 ns) peak is detected by an ADCMP600 comparator [139], generating a TTL signal which closes the switch[140], shortcircuiting the input and the 10 nF capacitor only during the first hundreds of nanoseconds of the input pulse. In this way, when the input pulse starts to decrease, the switch is opened, keeping the maximum amplitude for a much longer time. Thus, the signal which has to be digitized by the ADC lasts several tens of ms (the time required to discharge the 10 nF capacitor through the parasitic resistances), as it is shown in figure 6.14(d). In this way, the effect of a possible error in the digitizing timing is negligible. As we can control when and how often the calibration pulses are generated, working with long pulses and low repetition frequencies does not mean any limitation.

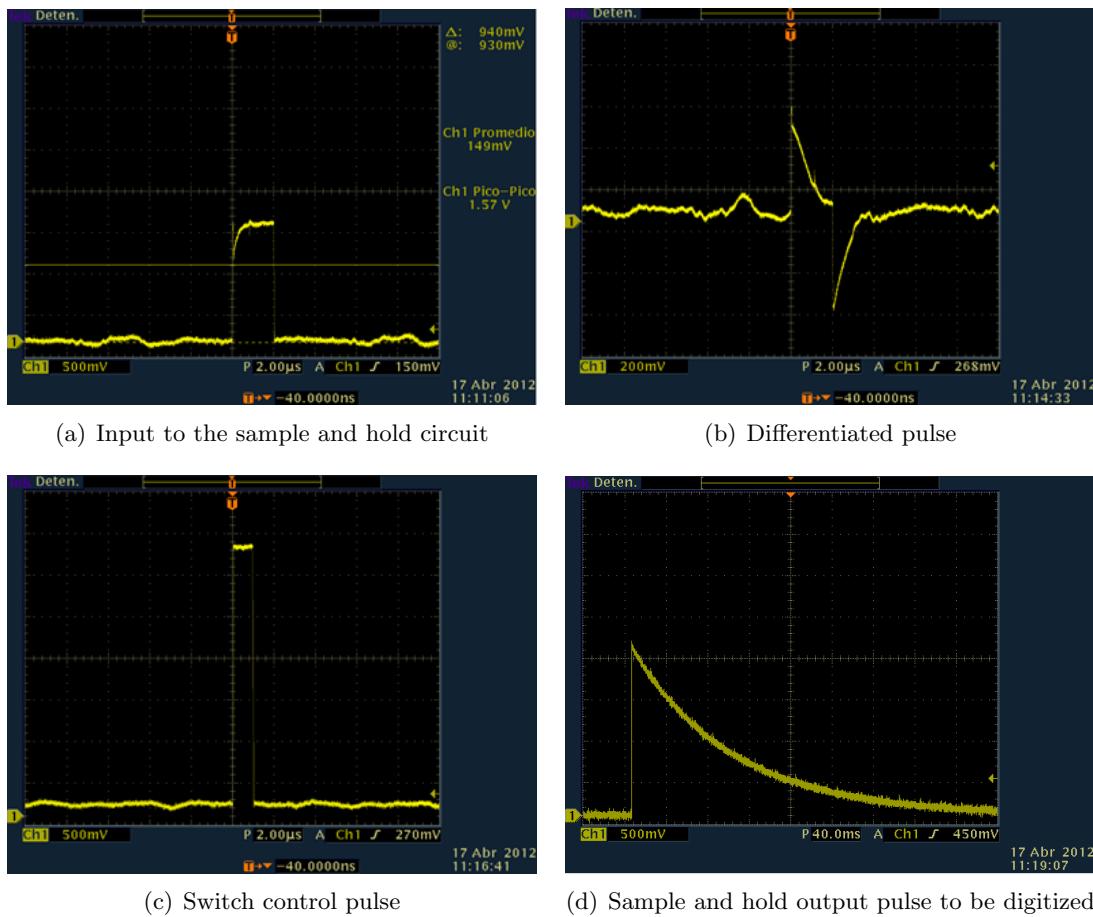


Figure 6.14: Waveforms at different points of the sample and hold circuit

Finally, the voltages measured by the ADC are read-out by the FPGA of the cluster front-end board. Figure 6.15 shows the voltages measured for different delays between the leading edges of the reference and the Level 1 output signals. For the purpose of this test, a large delay variation range has been used, by combining the delay lines and an additional cable length. As can be seen in the figure, the maximum voltage measured coincides with the best aligning, as was expected. In this way, the FPGA can change the delay corresponding to each pixel until achieving the maximum voltage in all of them, which happens when all the inputs will be aligned with the reference. In [141] it is possible to see the alignment algorithm at work.

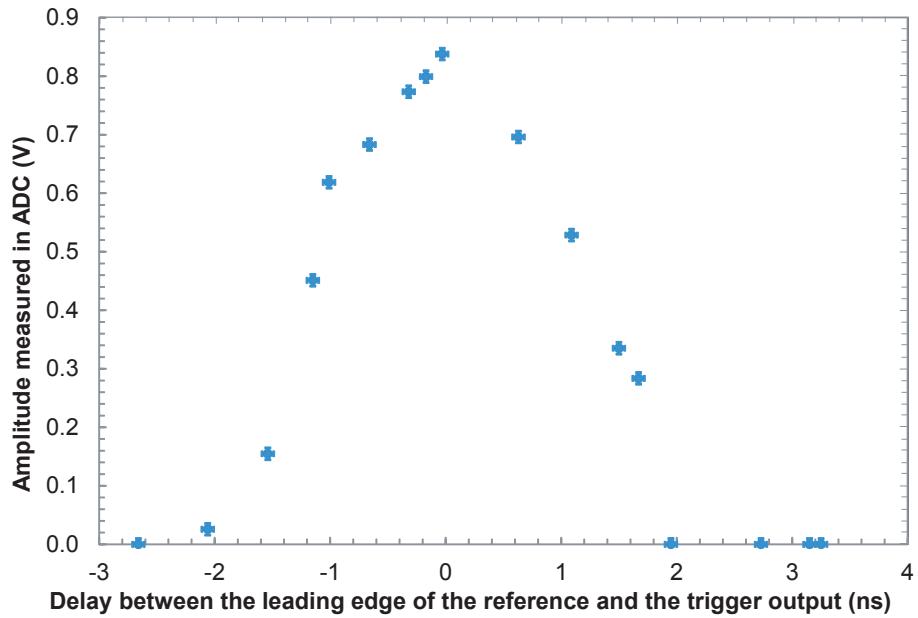


Figure 6.15: ADC measurement for different delays between the leading edges of the reference and the Level 1 output signal

6.2.2.4 Power saving

Except for the delay lines which are always connected, all the other circuits forming part of the delay compensation system are only used during calibration. With the aim of saving power, a calibration mode was implemented in the Level 1 system by setting $A_0 = "0"$ and $A_1 = "0"$. When the two mode control lines are not set to 0, none of the active components in the AND gate, the width to amplitude converter, the sample and hold or the DAC is powered. An AP2141 [142] power switch has been used to switch off the power. In this way, around 132 mW are saved.

6.3 Delay compensation without a reference

If the jitter in the calibration light emission makes unfeasible to generate a reference pulse synchronized with it, a different method from the one described in section 6.2 must be used instead. One possible solution, inspired in what it is currently implemented in MAGIC, requires to take one pixel as a reference, for instance the central pixel in the central cluster. By activating only this pixel and the central (or other) pixel of a neighbour cluster, and selecting a working mode which adds these two pixels, it is possible to know how well they are aligned with a rate scan. When the overlapping is optimal, the maximum added amplitude is obtained. Figure 6.16 illustrates how the alignment affects the rate scan of the addition. For similar input pulse amplitudes from the two pixels, it occurs that $B > A$. In this way, the delay of the second pixel can be adjusted with the delay lines described in section 6.2.1 until obtaining a perfect alignment.

In a second phase, the two pixels can be used to align other pixels in the neighbouring clusters. This procedure can be repeated until having one calibrated pixel in every cluster. All calibrated

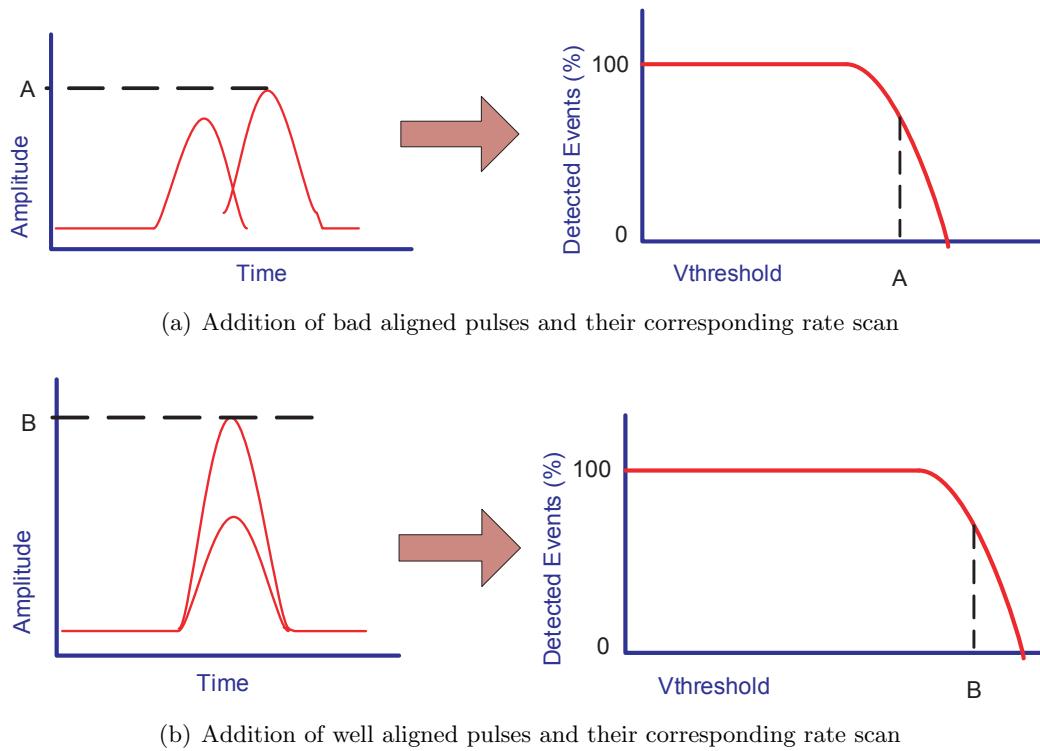


Figure 6.16: Sketch of the scheme to align pulses and based on rate scans

pixels can be used as references, so more pixels can be calibrated at the same time with each iteration. However, the selection of which pixels are being calibrated in each iteration must be done with care, because only 2 pixels should be added in each cluster at each step, corresponding to the reference and the pixel under calibration. If there are more pixels being added, finding the optimum alignment becomes much more difficult. Figure 6.17 shows the first 6 iterations of a possible algorithm to calibrate the delays in this way, assuming that the Level 1 is working in mode 2³.

Once there is a calibrated pixel in every cluster, the calibration mode can be selected and then the other pixels in the clusters can be calibrated with respect to the calibrated one. Choosing the calibration mode allows to calibrate one pixel in all the clusters at the same time, so after 6 iterations in this mode all the pixels in the camera would be adjusted.

This calibration scheme have some advantages but also several drawbacks. The main advantages are that:

- It can work with light sources with high jitter.
- It does not require additional hardware but the delay lines.
- It does not require to distribute a reference.

On the other hand, the drawbacks are that:

³This is the best selection to reduce the possibility of adding more than 2 clusters

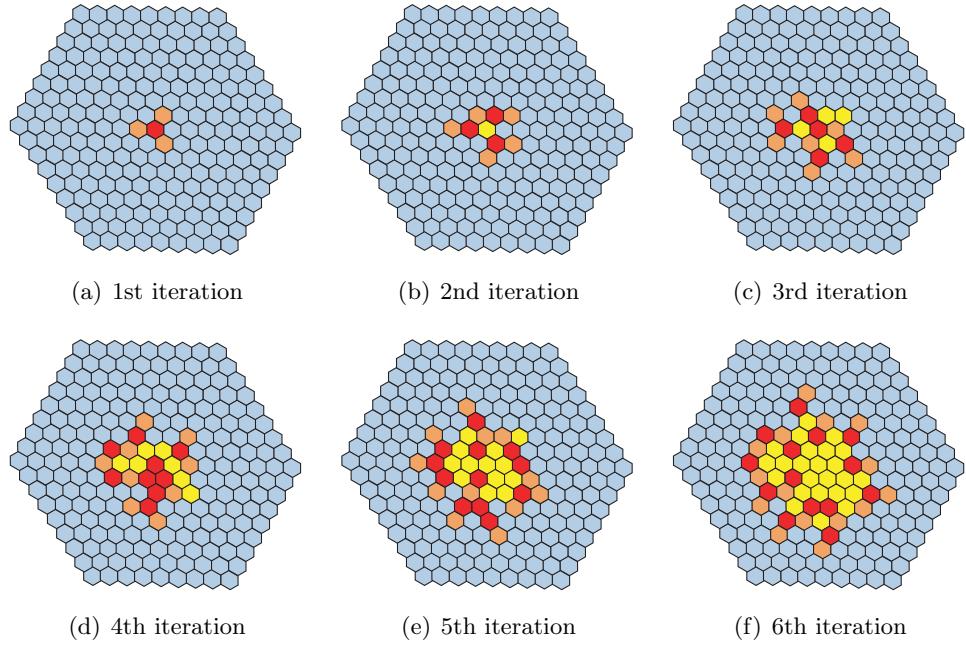


Figure 6.17: Possible first 6 iterations to calibrate 1 pixel in every cluster, each one represented as an hexagon in the figure. Orange clusters are under calibration, red clusters are being used as reference, yellow clusters are already calibrated clusters which are disabled to avoid adding more than 2 inputs in the clusters performing rate scans, and blue clusters are the ones not yet calibrated.

- As there is not a common reference, the inaccuracies are added in each iteration, so the last clusters to be calibrated can be quite delayed with respect to the first ones.
- This method requires much more time to calibrate the camera, requiring many rate scans for each pixel calibration.

Due to these reasons, the delay lines and also the delay meter have been included in the Level 1. The MSTs will probably use calibration lights developed with LEDs with small jitter, so the algorithm described in section 6.2 could be used, while for the LSTs, which are expected to use a laser as calibration light source, the second method would be used if finally the lasers have a large jitter. Anyway, as there is only one delay meter in each cluster, the over-cost will be very small.

6.4 Conclusions about the transit time delay compensation

The different transit times introduced by the PMTs degrade the performance of the trigger system in the LSTs and MSTs if the variations are higher than 250 ps. To avoid this degradation, each input to the trigger system must go through a variable delay line which must be adjusted to compensate the different delays. With the aim to know the delay which each delay line must add, a calibration system is also required. To make this calibration possible, the author of this thesis has developed an electronic system able to measure the time between two LVDS trigger pulses. This system has been included in the Level 1 trigger mezzanine and it is useful whenever a trigger

reference signal is available. In this conditions, the presented delay meter can achieve a very high accuracy. It must be remarked that one of the subcircuits which make up the delay meter, the width to amplitude converter, achieves a sensitivity of 768 mV/ns, which cannot be reached with any commercial width to amplitude converter circuit. On the other hand, if a reference is not available, a different algorithm should be used, like the one proposed in section 6.3, inspired in the current delay compensation system used in MAGIC.

Chapter 7

Trigger Interface Board.

The trigger interface board (TIB) was originally conceived with the name of stereo trigger board and with the only function of implementing the hardware stereo trigger algorithm in the LSTs, as described in section 7.2. This means that the board has to receive the local Level 1 trigger from the central backplane of the camera, perform certain operations with it and with the triggers coming from the neighbour telescopes and then send, using an specific line, the final trigger command to the central backplane to get it distributed and read out the camera.

However, the compliance with the stereo condition for a set of LSTs is only one of the possible sources that will produce a trigger signal that forces the readout of the camera. For instance it must be possible to trigger the camera on pedestal runs (to obtain the noise level from the NSB and the electronics), to take calibration readings, or to trigger the whole or part of the CTA array at the same time, using the so-called software triggers. There are several possible trigger origins and only one line in the central backplane to receive the trigger command, which is connected to the TIB. So, for the LSTs, the stereo trigger board assumed naturally the role of “trigger concentrator” becoming the trigger interface board. Other technical solutions could have been adopted, such as placing more lines in the central backplane to receive the different trigger origins and perform this function there, but the space for connectors in the backplane is very limited and it would have been mechanically complex. On the other hand, the TIB is not restricted to the cluster mechanics, as it is mounted in a 19” 1U rack box, with space enough to house several connectors for the different trigger origins. And last but not least, another reason to implement the trigger concentrator function in the TIB is that the development of this board has been scheduled three years after the first cluster prototypes, so the electronics in the cluster is a rather mature design, while the TIB still has flexibility to implement new functionalities.

Regarding the MSTs, in principle they can not participate in hardware stereoscopic trigger schemes (see sections 7.2.1 and 7.3.2), but they also need a system to concentrate the different trigger origins. As the MST cameras will have the same hardware elements and mechanical architecture than the LST ones, the TIB was also adopted to do the concentrator function in the MSTs, although the stereo trigger algorithm will be disabled in these cameras. This decision can seem a waste of resources, because it uses a board more powerful than strictly required for the MSTs, but in fact the TIB is rather cheap and using the same solution in MSTs and LST saves a great deal of development time and costs. Additionally, although not being stereo-triggered, the MSTs can

contribute to improve the LST stereo trigger in an easy way if the TIB is used, as it is explained in section 7.3.2.

Sections 7.1 and 7.2 describe the functions which are currently performed in the TIB, section 7.3 explains several functions and improvements which can be included in the future, section 7.4 describes the technical implementation, and section 7.5 shows the test results of the first prototypes.

7.1 Interfacing

The trigger interface board is placed in the camera of each LST and MST, interacting with other camera systems as shown in figure 7.1.

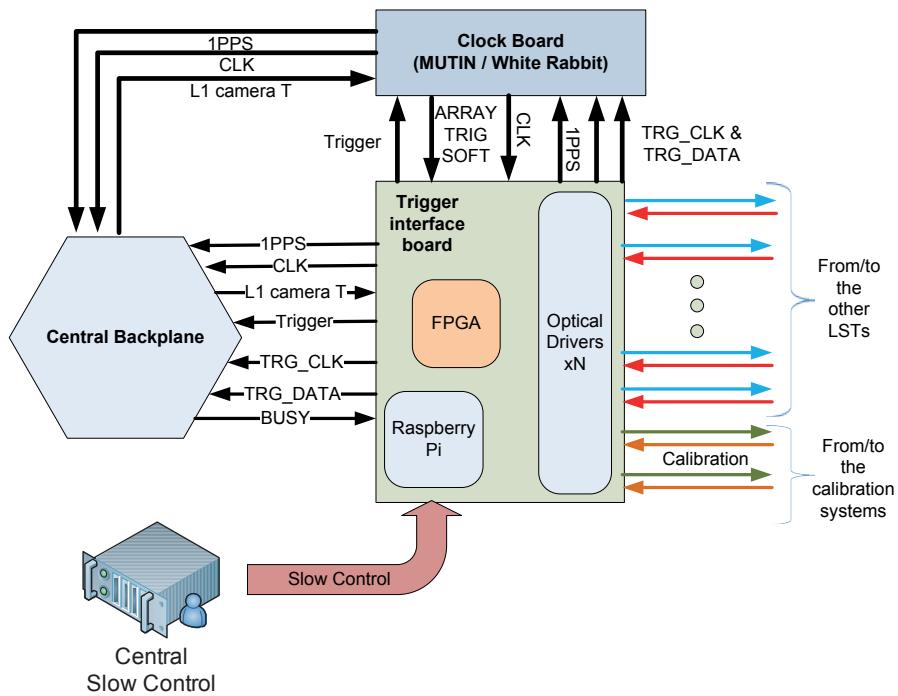


Figure 7.1: Scheme of the interfaces of the TIB with other systems in the camera

The systems which have common interfaces with the TIB are:

Backplane

The central backplane is a standard backplane as the one presented in section 3.2.1.4, with the only peculiarity of being placed at the center of the camera, so the Level 1 triggers generated at any cluster in the camera are sent to this central, as part of the Level 1 distribution scheme. The backplane has 3 RJ45 connectors (see figure 7.2): one of them is used for bypassing the Ethernet link of the front-end board, other is used to receive the clock and the 1PPS signals either directly from the clock board or through the TIB, and the third one, used for the trigger contains the following pairs:

- One pair to send the Level 1 trigger from the central backplane to the TIB.
- One pair to transmit the trigger command from the TIB to the central backplane, to get it distributed.
- One pair to transmit a BUSY signal from the central backplane to the TIB, which is useful to inhibit triggers if there is some cluster that can not accept it.
- One pair as SPARE.

Additionally, the TIB will be connected to other backplane different from the central one, in order to send to it the trigger type information which should be stored together with the data stream recorded in the front-end boards. Each event must have an associated trigger type; however, it is not required to distribute this information and store it in every cluster, but storing it in one cluster is enough. In the future, other links between the TIB and other clusters different from the central one could be defined, for instance to implement a topological trigger as the one described in section 7.3.5.

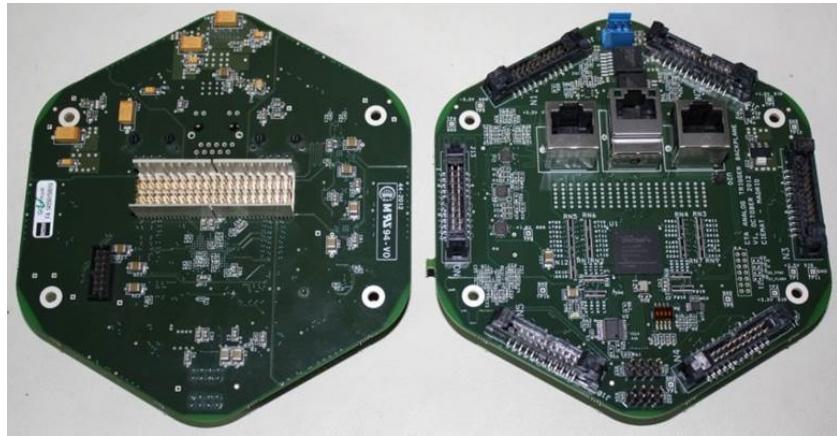


Figure 7.2: Photo of a backplane

TIBs in other LSTs

In order to perform the stereo trigger algorithm, all the TIBs of the LSTs need to receive the L1 trigger of the neighbouring LSTs. So a pair of optical fibers will be connecting each pair of LSTs (one fiber for each direction). In this way, every LST sends its local L1 trigger to all the others, while receives the Level 1 triggers of its neighbours.

Calibration box

The calibration box is placed typically in the center of the telescope reflector and is used to control the calibration light pulses (see section 3.2.1.7). These light pulses, implemented with powerful lasers or LEDs, are able to illuminate synchronously all the pixels of the camera with nearly the same light intensity. When the calibration procedure is taking place, the calibration box produces triggers coordinated with the light pulses, which should trigger the camera readout. In the same way, the calibration box also produce the so called “pedestal triggers” when the lights are switched off, to take measurements of the night sky background.

These signals are sent through optical fibers to the TIB which generates the corresponding trigger commands to read the camera.

Array trigger and clock board

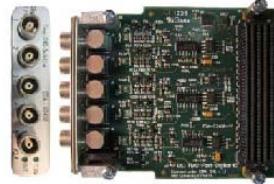
In order to reconstruct events observed by several telescopes in the array, it is very important to know very accurately when the samples were taken. To do so, a synchronized clock signal is required in each IACT, with a maximum difference of 1 ns between telescopes¹. There are two main developments competing to build this system, known in the collaboration as MUTIN and White Rabbit [99] (figure 7.3). Both developments consist of a central clock which is sent to all the telescopes by means of optical fibers, clock receptors in each camera, and calibration algorithms which measure the length of the links to take into account the propagation time and thus obtain a high accuracy.



(a) Central unit



(b) Clock receptors



(c) Time stamping mezzanine

Figure 7.3: Different components of the White Rabbit system [99].

In addition to distributing the primary clock used to generate the sampling clocks, the clock receptors in each camera (hereinafter clock boards) are also useful to time stamp the camera triggers, which are subsequently sent to the central array trigger unit to perform the software array trigger procedure, described in section 3.2.8.2. With this aim, copies of the camera trigger command pulse and of the trigger type are sent from the TIB to the clock board. Moreover, the highly accurate clock of the clock board can be used to program special so-called software triggers at a certain time, so the clock board can also be used as a trigger origin (signal ARRAY_TRIG_SOFT in figure 7.1). Finally, the clock board also provides with a stable 10 MHz clock signal to the TIB, which uses it as FPGA clock after frequency multiplication.

Central slow control unit

The central slow control unit consists of a server computer connected to the camera Ethernet network switch. The TIB, as well as other “intelligent” devices of the camera supporting Ethernet connectivity, communicate directly to this central unit through the network. The central slow control unit controls certain parameters and, in turn, is also controlled by the central control of the CTA array.

¹According to CTA specification.

7.1.1 Trigger types

Once presented the different systems related with the TIB, it is time to summarize the different trigger types that the TIB can handle:

- Local triggers from the Level 1 of some cluster, which can trigger the camera if working in mono (as will be the case of MSTs).
- Stereo triggers when the local Level 1 trigger together with some neighbour telescope fulfills the stereo trigger condition.
- Calibration triggers generated by the calibration box.
- Pedestal triggers generated by the calibration box.
- Scheduled software-triggers at a certain time, produced by the clock board.
- Triggers which can not be read because they occur during the camera dead time.

Some of these triggers can be interleaved, i.e. calibration triggers occurring during normal data taking producing camera triggers, so the information regarding the kind of trigger which fired the camera must be stored together with the data. To do this, the trigger type is sent to a backplane together with the camera trigger command. The timing restriction for the trigger type is not so constrained as for the trigger pulse, but the trigger type information must be available in the front-end FPGA before the data acquisition is finished (i.e. in less than $1 \mu\text{s}$), to include it with the rest of the event's data stream.

As the number of available differential pairs² is limited and high speed is required, a protocol with one line for the clock and using only two pairs is preferable. A slight variation of SPI protocol was defined, together with the DRAGON and NECTAr teams. This protocol is shown in figure 7.4. The only difference with SPI is that, instead of keeping a line with a negative value during all the data transmission, in this case the trigger pulse is used to warn the front-end FPGA about the data arrival. As there are only 6 possible trigger types, they can be encoded with only three bits, which will last 300 ns to be sent with a 10 MHz clock, fast enough to include them with the data. It is worth to mention that the backplane FPGA is transparent to the trigger type communication.

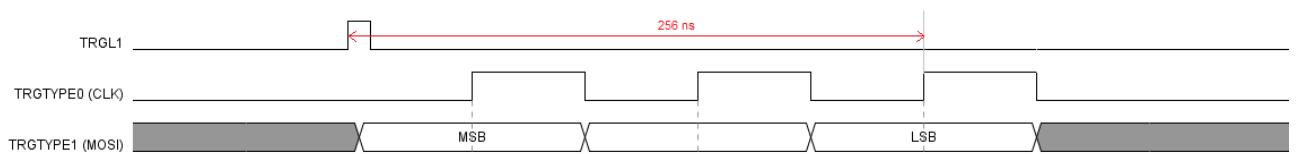


Figure 7.4: Protocol defined to transmit the trigger type information between the TIB FPGA and the front-end board FPGA

²The EM environment inside the camera can be quite noisy, so differential signals are mandatory. This is specified by CTA

Apart from sending the trigger type to the clusters to store it with the data, the trigger type must be also sent to the clock board, to include it with the event time stamp. The communication is done by means of another two differential pairs and with exactly the same protocol used for the transmission to the clusters.

7.1.2 Time stamp and event building

The time stamping of the triggers can be done in a double way: in the clusters and in the clock board.

The front-end boards receive from the clock board a 10 MHz clock (only in the case of Dragon) and a 1 PPS signal which is used to synchronize the internal oscillators used for taking the samples. These signals can be used to generate an absolute time stamp for each event, and even to know exactly when each sample was acquired, with an accuracy of 1 ns if sampling at 1 GHz. This is very useful for the event building, but it has a problem: the time stamp is not ready until the data packet is built.

With the aim of speeding up the generation of the time stamp and thus sending it sooner to the software array central trigger unit (see section 3.2.8.2, not to be confused with the LST hardware stereo trigger), and taking advantage of the clock board capabilities, the camera trigger signal is sent also from the TIB to the clock board to get time stamped. In this way, the event time stamp, together with the trigger type information are sent to the array central trigger, which uses this information to decide whether the data which will be stored in the camera server digital memories deserves to be stored to disk or not³. The sooner the time stamp is sent to the array central trigger, the smaller the memory size required in the camera server.

Additionally, there can be small time delays among the time at which the different clusters are read out (specially if a sliding window readout scheme like Colibri is used), so the time stamp of the trigger pulse in the clock board can be very useful to take it as the “event time stamp”, instead of having many slightly different time stamps for each cluster.

Apart from the time-stamp, an event number corresponding to each camera trigger command will be added to the data recorded by each cluster and also to the time-stamp generated in the clock board. Thus, the camera server will be able to identify the data packets corresponding to each event in an straightforward way, match them with the event time stamps, and in sum, to build the event in a simpler way. By using the event numbers, the time-stamping in each cluster is redundant if all the clusters are read synchronously. However, if a sliding readout window scheme is used, storing the time at which each cluster is read-out is mandatory.

7.1.3 Pin definitions

All the interfaces of the trigger interface board have been implemented with optical fiber pairs or with RJ45 connectors. The definition of the signals in the optical fiber pairs is obvious: one fiber to the output signals and other for the input ones. Regarding the RJ45 connectors, the one dedicated to the slow control is a standard Ethernet one. On the other hand, the differential pairs of the 2

³The trigger type information is very important for the array trigger, because coincidences with calibration, pedestal or programmed triggers should not be taken into account in the decision. The data from the telescopes which were triggered “artificially” must be stored, but not necessarily the data from the other telescopes in coincidence with them.