

Technical notes

GET electronics samples data analysis



J. Giovinazzo^{a,*}, T. Goigoux^a, S. Anvar^c, P. Baron^c, B. Blank^a, E. Delagnes^c, G.F. Grinyer^b,
J. Pancin^b, J.L. Pedroza^a, J. Pibernat^a, E. Pollacco^c, A. Rebi^a, T. Roger^b, P. Sizun^c

^a Centre d'Etudes Nucléaires de Bordeaux Gradignan (CENBG) – UMR 5797, CNRS/IN2P3 – Université de Bordeaux, Chemin du Solarium, F-33175 Gradignan, France

^b Grand Accélérateur National d'Ions Lourds (GANIL), CEA/DRF – CNRS/IN2P3, Bvd Henri Becquerel, 14076 CAEN, France

^c IRFU, CEA, Université Paris-Saclay, F-91191 Gif-sur-Yvette, France

ARTICLE INFO

Keywords:

GET electronics

Data analysis

TPC

ABSTRACT

The General Electronics for TPCs (GET) has been developed to equip a generation of time projection chamber detectors for nuclear physics, and may also be used for a wider range of detector types. The goal of this paper is to propose first analysis procedures to be applied on raw data samples from the GET system, in order to correct for systematic effects observed on test measurements. We also present a method to estimate the response function of the GET system channels. The response function is required in analysis where the input signal needs to be reconstructed, in terms of time distribution, from the registered output samples.

1. Introduction

The General Electronics for TPCs (GET) [1] has been developed to address new needs in nuclear physics instrumentation. In particular, it has been designed for a new generation of time projection chambers (TPC) and active targets. Several devices of that type have been successfully used in the last decade. For example, the MAYA active target [2] was used in various reaction studies [3,4]. Another case is the TPC device [5,6] built for the direct observation of 2-proton radioactivity of ⁴⁵Fe [7] and ⁵⁴Zn [8] and other exotic decay mode studies [9]. Time projection chambers currently under development, such as ACTAR TPC [10] in Europe, AT-TPC [11] in the USA, S π RIT [12–14] and HypTPC [15] in Japan, are using this readout electronics. In addition, the GET system has been designed in a versatile way, in order to be used with other types of detectors, such as semiconductors and scintillators.

The GET electronics is highly integrated, and can equip detection devices with up to 30 000 signal channels. The detector channels are connected to the AsAd (ASIC and ADC) front-end boards, each of them housing 4 AGET chips. The AGET chip [16], based on ASIC technology, contains the electronics circuits for the processing of the input signals from 64 channels of the detection setup. The analogue signal of each channel is sampled in time, and stored in a circular memory buffer (a switched capacitor array, SCA). This circular memory is filled continuously, until a write stop is received, when an event is accepted. The content of the memory is then digitized by the 12 bits ADCs from the

AsAd board. The data, resulting from the processing by the AGET chip of the input signals, and their digitization by the AsAd board, are transferred to concentration boards (the CoBo modules), held in a μ -TCA crate, connected to the data acquisition workstations. The crate also contains a trigger module, MuTANt (MuLtiplicity, TrIGger And Time), which can manage 3 levels of triggers for the whole system.

In this paper, we focus on the AsAd board and the AGET chip that are in charge of generating the digital output data from the analogue input signals (the processing of the numerical data and the trigger issues are outside the scope of the present work). The output samples are the data stored by the acquisition system that have to be analysed for the experiment purpose, such as charged-particle trajectory reconstruction in a TPC.

The samples have to be analysed, on a channel-by-channel basis to retrieve the information from the input signal coming from the detector: the total charge deposit, or the charge distribution in time. In order to improve the accuracy of the channel signal analysis, some systematic effects coming from the GET electronics should be corrected. This is presented in Section 3.

Due to the channel signal processing by the AGET chip, the output samples are not a direct digitization of the input charge distribution. It is nevertheless possible to reconstruct the input signal by deconvoluting the registered data from the response function of the channel electronics. An empirical method is proposed to estimate this response function for each channel in Section 4.

* Corresponding author.

E-mail address: giovinaz@cenbg.in2p3.fr (J. Giovinazzo).

2. GET test-bench

To perform the test measurements, we used a *reduced-CoBo* module, developed at NSCL/MSU and IRFU, which is a simplified version of CoBo that manages only one AsAd board (instead of 4). The *reduced-CoBo* module is based on a commercial ML507 board, and an adapter card has been designed to connect it to the AsAd board with a VHDCI cable. The test-bench software developed at IRFU is used to load the configuration of the electronics and to readout and store the events.

The test system thus contains 4 AGET chips, each containing 64 signal channels. The signal from each channel is processed by a 4-stage electronic chain before being stored in the SCA buffer memory: a preamplifier (charge sensitive amplifier, CSA), a pole zero cancellation (PZC) stage, a shaper (Sallen-Key filter) and an inverting $\times 2$ gain stage.

In addition to the signal channels, each AGET chip has 4 additional channels, the *fixed pattern noise* (FPN) channels, that can be used for the reduction of some of the coherent noise of the signal channels. For these FPN channels, the input processing only consists in the inverting $\times 2$ gain, without CSA, PZC and shaper.

For all channels (including the FPN), the processed signal is stored in the analogue memory buffer, with a write frequency F_W ranging from 1 to 100 MHz. The buffer contains 512 memory cells (or time buckets). When an event is triggered, the writing of the memory is stopped, and the memory content is digitized by the AsAd board. As a result, the output data for each channel is a time sample with up to 512 digital values. In this paper, we adopt the following definitions:

- the *circular buffer memory* is the SCA (one for each channel), in which the analogue signal is written continuously;
- a *cell* (or *time bucket*) is a single element of the SCA;
- a *sample* refers to the set of N (up to 512) digitized data of the buffer memory of one channel: for example $s[k_t]$, where k_t is the time index of the sample value (in the range $[0; N - 1]$), corresponding to the instant $t = k_t/F_W$ in the full time window of the sample (the bracket notation indicates the discrete character of the signal);
- an *event* is the readout of the samples for all channels, resulting from a trigger (either external or from slow control).

2.1. Test measurements

The GET system is highly configurable at all levels of the electronics. For physics experiments, only a few parameters are really relevant at the analysis stage. The key parameters are:

- the sampling frequency F_W (from 1 to 100 MHz)¹;
- the peaking time τ (in ns), which is the shaping time of the amplification chain;
- the gain parameter g which corresponds to the maximum input charge (in pC or fC), selected by the input capacitor C_g , (in pF or fF);
- the AsAd pulse generator amplitude V_{in} (in V) when used;
- the trigger delay d_{trg} that sets the delay between an event trigger and the effective stop of the SCA memory; it is tuned for example with generator measurements to set the pulse at a given position in the time window of the samples.

For the measurements presented, the data were stored in *full readout* mode, which implies that all data from all channels were written to the output files, regardless of any threshold or pedestal. In our analysis, mainly two types of measurements have been performed:

- Runs in *functionality test* mode. The pulse generator embedded on

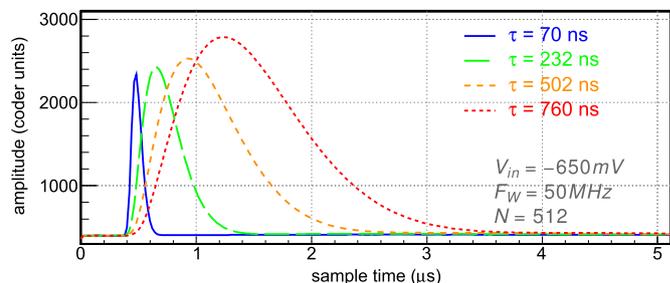


Fig. 1. Example of output sample, for one channel, in the case of a measurement with the AsAd pulse generator (*functionality mode*). In this plot, the measurement is performed with $N=512$ values in the sample and a write frequency $F_W = 50$ MHz, thus covering a time window of $10.24 \mu\text{s}$ (the displayed signals are zoomed to a $[0; 5.12 \mu\text{s}]$ time window). The different curves correspond to different values of the peaking time τ , all other parameters being constant. Each curve is an average of 1000 pulse generator events.

the AsAd board is triggered by the slow control. It generates a voltage step on a 100 fF capacitor at each channel input, which results in an almost point-like charge deposit. A write stop of the SCA is issued to process the event. In this mode, the FPN channels input is directly the generated pulse.

- Runs in normal *external* mode. An external trigger is sent to the reduced CoBo module, and the signal from all channels is processed. In this mode, there is no input signal for the FPN channels.

2.2. GET output samples

The data stored for the measurements presented here have been acquired from the full buffer memory, $N = 512$. Thus, the raw data are, for each channel, samples of N values ranging from 0 to 4095, from the 12-bits ADC. The time window covered by the samples depends of course on the write frequency F_W in the analogue memory. An example of such data is presented in Fig. 1, in the case of a *functionality test mode* measurement.

The registered data present distortions of the signal at the beginning and the end of the samples. These artifacts, due to the electronics behaviour, are illustrated in Fig. 2. The effect observed at the beginning and the end of the sample are due to the on-off switching of the write mode in the analogue buffer memory, resulting in a signal distortion in the order of 10 coder units. In addition, readout induces an additional artifact on the first data that causes an underestimation of this value of more than 50 coder units. The identification of the origin of these problems is discussed in Section 3.2. These effects are reproducible, event by event.

The first solution could be the truncation of the sample, but this option has not been considered here, since for further processing (see Section 4), it is important to keep samples with a size $N = 2^k$. Nevertheless, for the corrections proposed in Section 3.1, the truncation remains a valid way to avoid these problems at the analysis level. These effects can also be corrected by extrapolating the neighbouring values, but we will see further a more natural solution to this problem (see Section 3.1.1).

3. Raw data corrections

In this section, we analyse some systematic distortions of the measured signal, and we propose some correction procedures. This is achieved by analysing the *baselines* (or pedestal), which correspond to the data collected while no signal is present at the channels input.

3.1. Baseline analysis

In an ideal case, with no intrinsic noise from the electronics, we would simply expect a constant offset value. This is obviously not the

¹ The write frequency is limited to 6.25, 12.5, 25, 50 and 100MHz by the software controlling the reduced CoBo configuration.

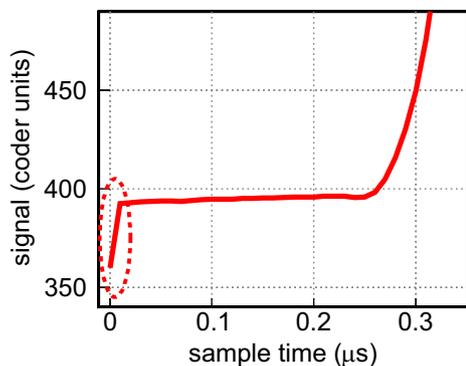


Fig. 2. Measured sample artifacts: the registered values for the first data of the sample and for the end of the sample exhibit a systematic effect. These distortions originate from the switching of the write and/or read modes of the electronics (see text).

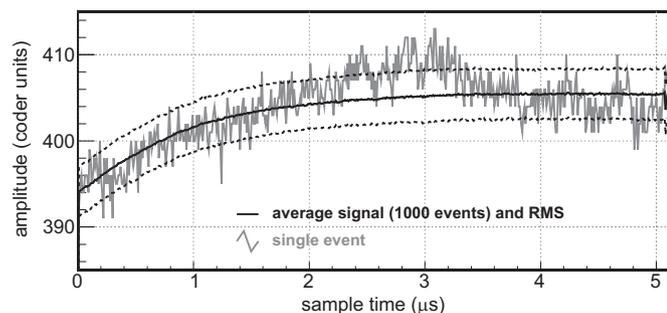
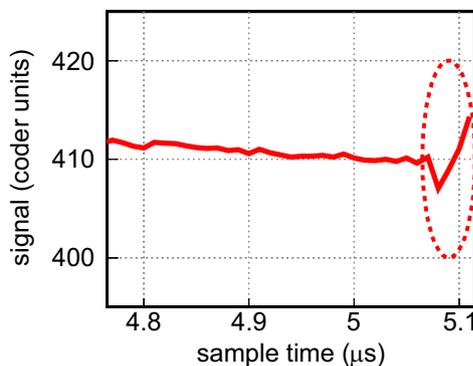


Fig. 3. External baseline estimate, resulting from the average of 1000 events measured with an external trigger and no input signal (full line black curve). The dashed curves show the standard deviation (RMS) of the events with respect to the mean sample. The grey curve shows a single event to illustrate the intrinsic noise.

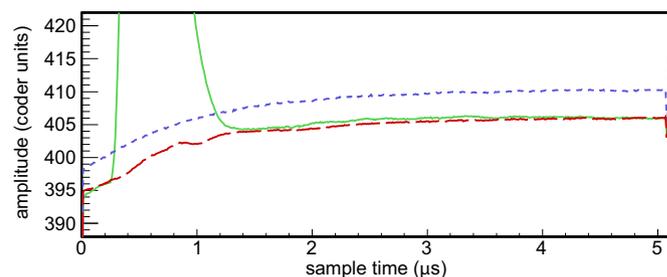


Fig. 4. Run with AsAd pulse generator compared to baselines. The full line curve (green) shows the average signal measured for AsAd pulser events (here with an amplitude $V = -50$ mV and channel parameters $\tau = 232$ ns and $g = 120$ fC). The small dashed line curve (blue) is the *external baseline* (as defined in Section 3.1.1). The long dashed line curve (dark red) is the *pulser baseline* estimated from pulse generator runs (see Section 3.1.2). (For interpretation of the references to colour in this figure caption, the reader is referred to the web version of this paper.)

case, and we present two effects that should be corrected for. The first one is the average baseline shape, as a function of sample time, that can trivially be corrected in the case of normal measurement conditions (measurement of the signal coming from a detector connected to the GET electronics). The second effect is related to the position at which the write stop occurs, in the SCA analogue memory. This effect is called here a *phase effect* since the memory is a circular buffer and the correction depends on where the last data of the measured sample is stored in this buffer.

3.1.1. External baseline

A baseline event can be obtained by sending an external (random) trigger to the reduced-CoBo module, while the channel entries are not connected: such a sample is noted $s_{evt}^{in=0}[k_r]$. Such an event also measures some intrinsic noise of the electronics channel. When averaging over a large number of events, the contribution of the noise can be significantly reduced. The baseline sample resulting from the average over 1000 events, $b_{ext}[k_r] = \langle s_{evt}^{in=0}[k_r] \rangle_{evt}$, is illustrated in Fig. 3.

The baseline is clearly not a constant pedestal. The readout of the buffer memory induces a distortion of the sample. The baseline position and shape may slightly vary from one channel to another.

The first correction to apply on a measured sample is thus to subtract the baseline estimated for the corresponding channel. Note that the samples problems shown in Fig. 2 are present on both the measurement and the baseline. As a consequence, the *baseline correction* proposed here also corrects this problem of the first and last sample data.

3.1.2. Pulser baseline

In the case of test or calibration measurements with the pulse generator of the AsAd board, the external baseline defined previously does not match precisely the expected correction (see Fig. 4). Our interpretation is that the pulse generator is able to generate an input

pulse for AGET through different values of capacitors in series with non-ideal switches that cause an offset voltage at the ASIC output.

Thus, in the particular case of measurements with this generator, instead of using the baseline as described in Section 3.1.1, it is better to define a specific *pulser baseline*. This one is obtained by measuring the signal in the pulse generator mode (the *functionality test mode*), with a zero amplitude.² This *pulser baseline* is noted $b_{pul}[k_r]$ in this paper.

3.2. Buffer memory readout phase effect

In addition to the distortion coming from the buffer memory readout, another effect coming from the buffer itself can be addressed. In standard running mode, the event trigger will stop the memorization of the analogue signal in the circular buffer (stop of the SCA write mode). This may occur while writing any of the N memory cells. The last written cell (or time bucket), l_c , thus defines the end of the measured sample. Then the SCA readout and coding starts at the next cell $l_c + 1$, and is achieved for all cells of the circular buffer: from $l_c + 1$ to $N - 1$, and then from 0 to l_c .

A systematic effect has been observed that depends on the cell position (labelled from 0 to $N - 1$) in the buffer. It is referred to as the *phase effect* because of the circular character of the memory: the effect is repeated periodically while the writing process is looping over the N memory cells. Due to the random position of the SCA write stop, such a systematic effect is shifted with respect to the measured sample time, as shown in Fig. 5.

² With reduced-CoBo, it is not possible to run the *functionality test mode* with a zero pulse amplitude. The *pulser baseline* can nevertheless be built from the average of two runs with opposite pulse amplitudes, for example +1 and -1 mV. It should be noted that for negative pulse amplitude, the measured signal is positive, and that due to the baseline offset, the GET system can measure also negative signals, as long as the output is not lower than 0.

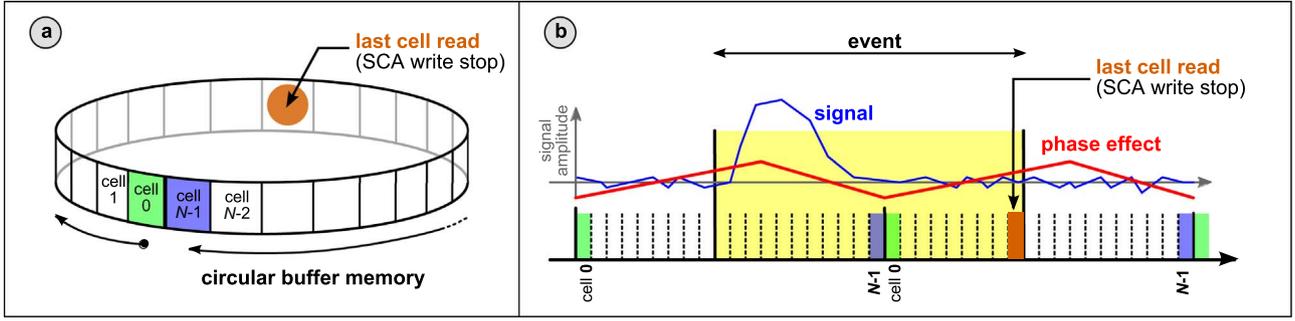


Fig. 5. Illustration of the SCA circular buffer memory. The signal is written continuously in the SCA memory, and when the end of the memory is reached (cell $N - 1$), the writing continues at the beginning (cell 0), and the memory is overwritten cell by cell. In standard readout mode, the event trigger induces the stop of the SCA write mode: the last written cell is shown with the spot in the schematic representation of the circular memory, (a). The readout starts in the next cell and loops until the last written cell is read. The event that is read thus corresponds to the memory content when the SCA write stop occurs, as shown on the chronogram in (b). For a systematic distortion of the signal that depends only on the memory cell position (phase effect), the effect on the signal is randomly distributed in time, because of the

It is nevertheless possible to correct this effect since the absolute index of the last written (which is also the last read cell) cell information is available in the output data. We propose here a way to estimate this systematic distortion of the signal as a function of the cell number, and a solution to correct for this effect with the *last cell read* information. In order to measure the *phase effect*, we used the ability of the system to read and convert data from cell 0 to cell $N - 1$ instead of reading from where the write stop occurred (the *read from column 0* mode).

The upper part of Fig. 6 shows the baselines measured in standard readout mode $b_{ext}[k_i]$ and in readout from column-0 mode $b_{ext}^{(0)}[k_i]$. Both curves result from an average of 1000 events. The pattern of the *phase effect* corresponds to the difference between these two curves (lower part of Fig. 6):

$$\varphi[k_i] = b_{ext}^{(0)}[k_i] - b_{ext}[k_i]$$

A single event (in read from column-0 mode) is displayed, showing an artificial oscillation caused by the SCA write stop. This effect is randomly distributed in the sample, for each event, since the trigger is asynchronous with respect to the full sampling period. A first consequence is that, for the same number of events, it induces more important fluctuations for the baseline in readout from column-0 than for the baseline in standard mode. For the baseline in standard readout mode, this effect is always located at the end of the sample, for each event. Then a second consequence is that, on average, this effect is visible at the end of the average standard baseline, while it disappears on the average column-0 baseline. When estimating the *phase effect* (difference of the two baselines), this effect causes an artificial oscillation on the last data points. A simple solution to correct this problem is to extrapolate the end of the estimated pattern from the previous points, since on average it varies very smoothly. In addition, to reduce the effect of increased fluctuations (due to the column-0 baseline, as mentioned previously), we considered two solutions. The first one was to increase the number of events for the average baseline: this would require more than 5 times more events to get approximately the same fluctuations than for the standard baseline. The other solution was to apply a smoothing filter to the data: this is the option illustrated in Fig. 6.

The problem for the first sample data mentioned in Section 2.2 (and shown in Fig. 2) is present on both baselines (standard and readout from column-0). So when making the subtraction to compute the *phase effect*, this problem disappears and no specific correction is needed here.

3.3. Raw measurement corrections

In this section, we present how to correct the raw samples to take into account the systematic effects described previously. The correction

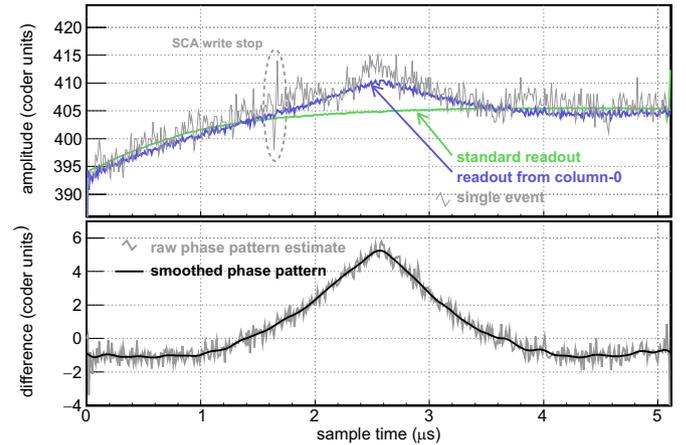


Fig. 6. Phase effect pattern. The upper picture shows the baseline measured in standard readout mode (the readout starts after the last written cell) (green), and in readout from column-0 mode (see text for more details) (blue). These curves correspond to an average of 1000 events. The grey curve shows a single event (readout from column-0): the oscillation observed (dashed ellipse) corresponds to an artifact caused by the SCA write stop. The lower picture shows the pattern of the *phase effect* (grey line) resulting from the difference of the previous baselines. The black curve is the *phase effect* after correction of the last data and smoothing (see text for details).

with the help of the FPN channels is the natural correction to be applied for standard measurements. Nevertheless, when running system calibrations in *functionality test mode* (see Section 2.1), this cannot be applied anymore and an alternative correction is proposed. The estimate of the channels response function requires such kind of signal correction.

In order to quantify the effect of these raw signal corrections, we first analyse the intrinsic fluctuations of the measured signals. This is achieved by studying the standard deviations (RMS) observed for a large number of events (typically 1000), with the same input signal. For one channel, this RMS sample is:

$$r_S[k_i] = \sqrt{\langle (s_{evr}[k_i] - \langle s[k_i] \rangle_{n_{evr}})^2 \rangle_{n_{evr}}}$$

where $s_i[k_i]$ is the sample measured for event i (for one channel), and the $\langle \rangle_{n_{evr}}$ notation indicates the average over all events of the measurement.

Since the fluctuations amplitude is relatively constant with time index in the sample, we define the average RMS for a single channel (averaging over the time index in the sample), and the corresponding standard deviation:

$$R_S = \frac{1}{N} \sum_{k_i=0}^{N-1} r_S[k_i] = \langle r_S \rangle_N \Delta R_S = \sqrt{\frac{1}{N} \sum_{k_i=0}^{N-1} (r_S[k_i] - \langle r_S \rangle_N)^2}$$

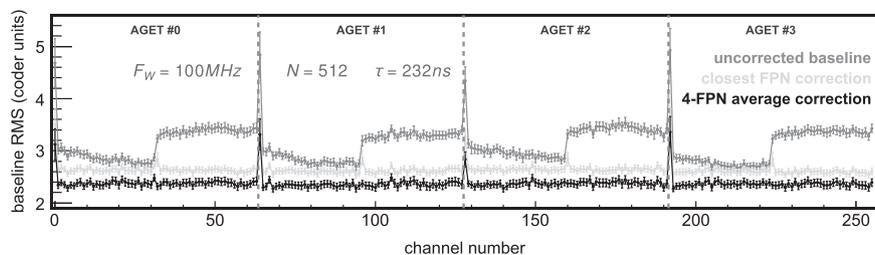


Fig. 7. Effect of FPN correction on intrinsic fluctuations. The medium grey points correspond to the RMS measurement R_S for each signal channel on the AsAd board for a baseline measurement run (see Section 3.1.1). The light grey points represent the same information for samples corrected with the closest FPN channel while for the black points, the correction is applied with the average of the 4 FPN channels of the AGET chip. The error bars are the standard deviation ΔR_S along the sample cells (see text for details).

3.3.1. Fixed pattern noise correction

The FPN channels have been included in the AGET chip in order to measure the coherent noise of the electronics, after the amplification and shaping stages, which includes the SCA memory.

The correction is applied by subtracting the sample of the FPN channels from the sample of the signal channels:

$$s_{i,j}^{cor}[k_t] = s_{i,j}[k_t] - s_{i,FPN}[k_t] \quad \text{with } k_t = 0, \dots, N-1$$

where $s_{i,j}$ is the raw sample for signal channel j of an event i , and $s_{i,FPN}$ is the FPN sample for the same event.

The 4 FPN channels of a chip are located regularly between the 64 signal channels, at channels 11, 22, 45 and 56 (all channels – FPN and signal – being numbered from 0 to 67). We considered the correction of a channel sample either with the closest FPN channel, or with the average of the 4 FPN channels. The resulting RMS in baseline measurement (intrinsic noise) for a full AsAd board is presented in Fig. 7. Due to the symmetries in the routing of the AGET chip, we also considered a correction of the channels samples with the average of the 2 first FPN channels for signal channels 0–31 and of the 2 last FPN channels for signal channels 32–63. The coherent noise is better reduced with the use of the 4 FPN channels average.

In addition, for uncorrected samples, the intrinsic noise is larger for the second half of the channels than for the first half, on each AGET chip. The RMS for corrected samples is quite constant for all channels, indicating that the difference between the first and second halves of the AGET channels arises primarily from a coherent source of noise.

Since the baseline (see Fig. 3) for each channel is not exactly the same, for samples corrected with FPN channels, a residual baseline remains, as illustrated in Fig. 8.

In addition to the event by event FPN correction, this remaining systematic distortion must be corrected for by subtracting this *FPN corrected baseline* for all signal channels.

3.3.2. Baseline and phase effect correction procedure

In the case of runs using the AsAd pulse generator (in *functionality test mode*), the FPN channels measure the input voltage pulse. They can no longer be used to correct for the baseline and the phase effect due to the write stop position in the memory buffer.

It is thus necessary to define the corrections to be applied, using the results of the analysis of Section 3.1. The average *baseline* (see Fig. 3) correction corresponds to a readout distortion that does not depend on the write stop of the SCA memory. The same sample has to be subtracted for all measured events. For the *phase effect* (see lower part of Fig. 6) correction, the pattern must first be shifted according to the last written cell in the circular memory, and then subtracted from the data sample.

As a consequence, they cannot be applied as a single correction and have to be considered sequentially. For each channel, the correction procedure is the following (see Fig. 9):

- the *baseline* sample (the *external baseline* for standard measurements or the *pulsar baseline* for runs using the AsAd pulse generator), as defined in sections 3.1.1 or 3.1.2, is subtracted from

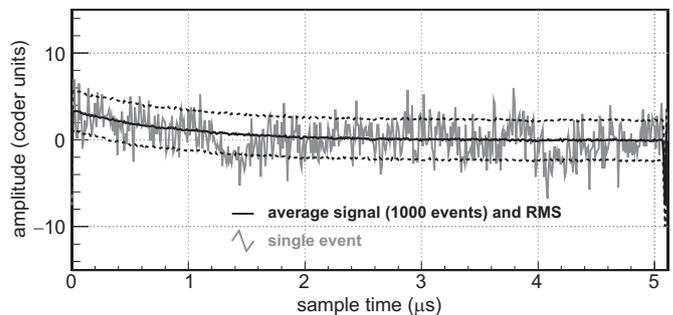


Fig. 8. Samples average after FPN correction. The black curve shows the *FPN corrected baseline* for a baseline measurement with 1000 events. The dashed curves show the standard deviation of the events with respect to the mean sample. The grey curve shows a single event to illustrate the intrinsic noise remaining after FPN correction.

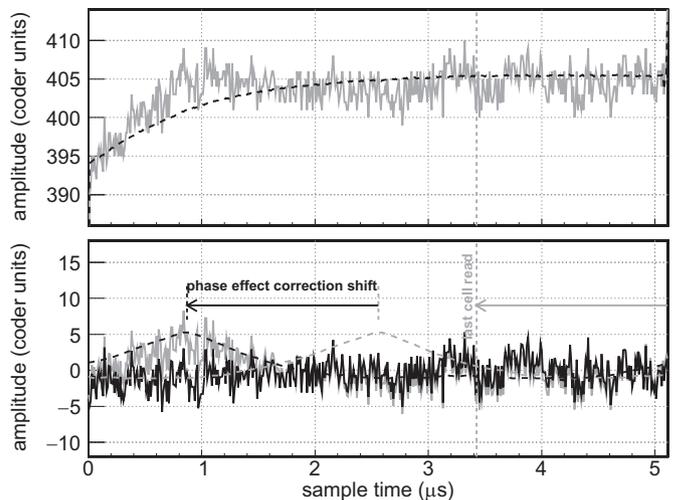


Fig. 9. Correction of events. The upper plot shows the raw data from one channel in a single event (a random external trigger, with no signal on the AsAd board inputs). The dashed line is the average *baseline* as described in Section 3.1.1. The lower plot shows the same sample, corrected for the *baseline* only (grey) and for the *baseline* and the *phase effect* (black). In this plot, the dashed grey curve is the *phase effect* as estimated in Section 3.2, and the black dashed line is the same, but shifted according to the position of the SCA write stop in the circular memory for the displayed event (indicated here by the vertical dashed line).

the raw data sample $s_{evr}[k_t]$:

$$s_{evr}^b[k_t] = s_{evr}[k_t] - b_{ext/pul}[k_t]$$

- the *phase effect* sample, as defined in Section 3.2, is first shifted according to the last written cell in the circular memory for the processed event, and then subtracted from the result from the *baseline* correction:

$$s_{evr}^c[k_t] = s_{evr}^b[k_t] - \varphi[k_t + \Delta k_t - eN]$$

where N is the sample size, $\Delta k_t = l_c + 1$ is the shift for the last cell

position, and ϵ keeps the time index in the range $[0; N - 1]$: 0 if $k_i + \Delta k_i < N$ and 1 if $k_i + \Delta k_i \geq N$.

3.3.3. Effect of the corrections on intrinsic fluctuations

In this section, we give a quantitative analysis of the effect of the raw samples corrections. We measured, for baseline runs (no input signal), the intrinsic fluctuations from the electronics output.

For a single channel, we first consider the standard deviation of the signal with respect to the average value $b_{cst} = \frac{1}{N} \sum_{k_i} b_{ext}[k_i]$ (mean value of the baseline over the whole sample time):

$$r_{cst}[k_i] = \sqrt{\langle (s_{evt}^{in=0}[k_i] - b_{cst})^2 \rangle_{evt}}$$

The RMS obtained when considering a constant pedestal is given for illustration (Fig. 10, dotted grey curve), but due to the shape of the baseline (see Fig. 3), its behaviour is governed by systematic deviation more than by fluctuations. It will not be considered anymore.

As an intrinsic fluctuation measurement, we rather consider the standard deviation from the average baseline (Fig. 10, dotted black curve):

$$r_b[k_i] = \sqrt{\langle (s_{evt}^{in=0}[k_i] - s_{ext}[k_i])^2 \rangle_{evt}}$$

When applying both the *baseline* and the *phase effect* corrections on an event by event basis, the resulting RMS $r_{b+\phi}[k_i]$ (Fig. 10, full black curve) appears to be 25% smaller.

For a single channel, the average deviations $r_b[k_i]$ and $r_{b+\phi}[k_i]$ are quite constant for all times in the sample. We can then consider the average values $\bar{r}_b = \frac{1}{N} \sum_{k_i} r_b[k_i]$ and $\bar{r}_{b+\phi} = \frac{1}{N} \sum_{k_i} r_{b+\phi}[k_i]$ for a quantitative comparison. This is illustrated in Fig. 11 for all the signal channels of an AsAd board.

The coherent noise correction applied either from the SCA phase pattern or from the FPN channels (as described in Section 3.3.1) gives a similar result in terms of intrinsic fluctuations reduction. Nevertheless, a systematic study of these corrections for various peaking times and various sampling rates (SCA write frequency) shows

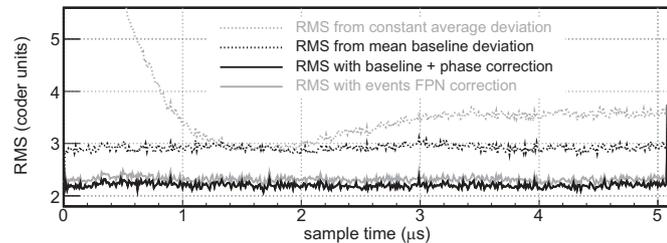


Fig. 10. This picture illustrates the effect of samples corrections on the intrinsic fluctuations of the measured signals for a single channel. The dotted grey curve is the deviation from a constant baseline (taken as the average value of the *external baseline*): $r_{cst}[k_i]$, and mainly reflects the systematic effect of the shape of the baseline. The dotted black curve is the mean deviation of the signal from the average baseline $r_b[k_i]$: it measures the fluctuations (RMS) when applying only the *baseline* correction. The full black line curve is the signal RMS when applying both the *baseline* and *phase effect* corrections: $r_{b+\phi}[k_i]$. The reduction of the intrinsic noise is almost the same as in the case of the FPN correction (see Section 3.3.1), plotted as full grey line.

that the *baseline + phase* correction is slightly better. In addition, in the case of an experiment with only few channels hit per AGET chip, the *phase pattern* correction does not require to store the 4 FPN channels for every event, which may represent a non-negligible fraction of the data volume.

3.4. Partial readout limitations

The raw signal corrections presented in this section have been analysed in full readout mode of the AsAd board. In the case of partial readout (only the channels selected by the trigger levels of the *GET* systems are read), which is in most cases the standard way of running, it appears that both the *baseline* (see Section 3.1) and the *phase effect* (see Section 3.2) may vary significantly with the number of channels that are read in a given AGET chip.

Under such conditions, the precision of the correction procedures presented here (see Section 3.3) is degraded. Nevertheless, an acceptable solution is to apply the FPN correction (see Section 3.3.1), despite the fact that the residual noise (fluctuations) is larger than with the corrections proposed for full readout mode (see Fig. 12).

As a conclusion, we suggest to apply corrections as presented in Section 3.3 for calibrations and characterizations of the electronics channels, as presented in the next section concerning the input signal reconstruction, and to apply only the correction from the FPN channels in standard measurements in partial readout mode.

4. Input signal reconstruction

Since the signal on each channel of the *GET* system is sampled in time, this should in principle allow a shape analysis to be performed on the input signal. In the case of a gas detector such as a TPC, the charge collection creating the input signal corresponds to the energy loss of the charged particles in the active gas volume. Depending on the incidence of the particle with respect to the collection plane, the charge distribution may vary, as illustrated in Fig. 13a and b. In addition,

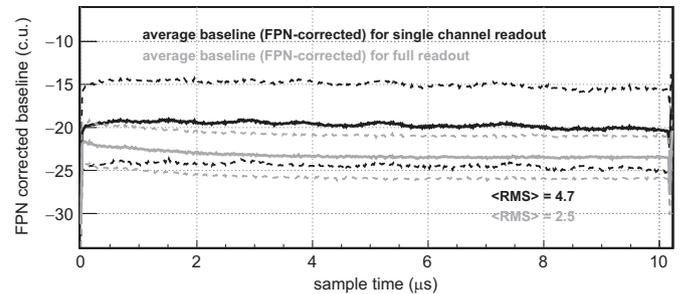


Fig. 12. Comparison of the baseline resulting from the FPN correction in full (grey) and partial (black) readout mode. The full lines correspond to an average of 200 events, and the dashed lines show the standard deviation. The example here is obtained for $F_W = 50$ MHz, and $\tau = 232$ ns. Despite an increase of the fluctuations, the correction for the partial readout mode gives an acceptable result. The remaining offset of about 20 coder units is due to the absence of a residual baseline correction (see Section 3.3.1 for details).

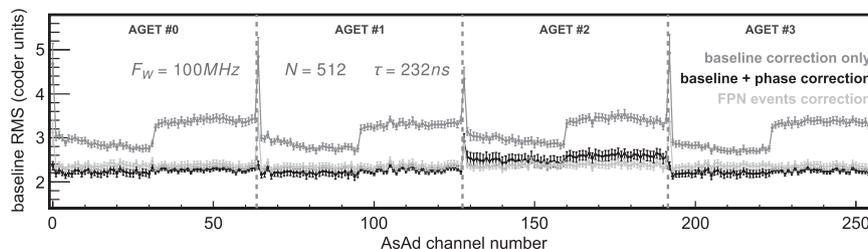


Fig. 11. Mean values of the signal RMS for the samples when considering the baseline correction only (dark grey points), both baseline and phase effect corrections (black points), or the FPN correction (light grey points). Each point corresponds to the average of the RMS for one channel (Fig. 10), summing on the sample time, and the error bars is the corresponding standard deviation. The plots show the values for all channels of the tested AsAd board.

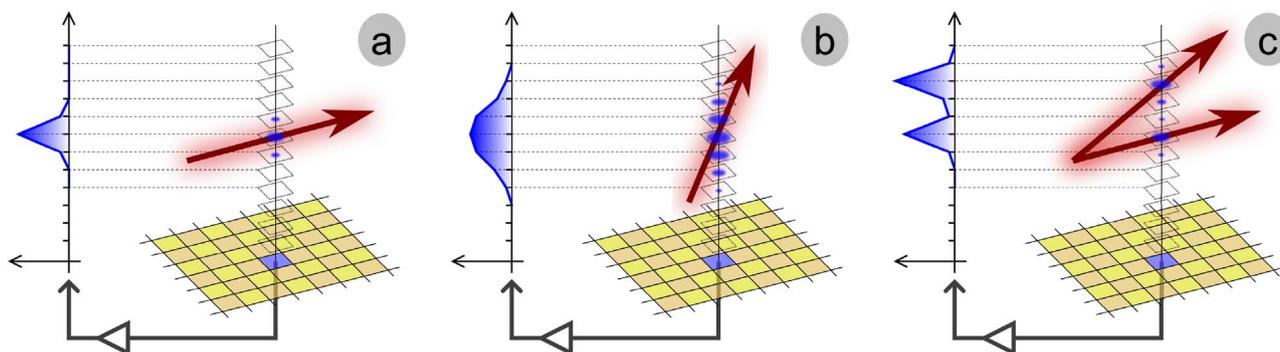


Fig. 13. Illustration of the variation of the shape of the charge collection signal in the case of a TPC detector. The arrows represent the charged particle(s) moving in the active gas volume. The ionization signal is collected on the bottom pad plane due to the presence of an electric field. On the left side of each picture, the distribution of the signal collected for a single pad (dark) of the collection plane is shown (the readout electronic channel is represented by the bold grey arrow). The ionization drift time corresponds to the vertical dimension and the time sampling is equivalent to a sampling along the vertical direction. In the case of an almost horizontal track (a), the signal is a narrow distribution, while for a track close to a vertical direction (b), the distribution should be much broader. The last picture illustrates the case where more than one particle track contribute to the signal: the sampling can, to some extent, allow signals from multiple tracks to be disentangled.

the time sampling should allow, in the case of a gas detector, to identify the contributions from different tracks to the total signal distribution, as shown in Fig. 13c.

A similar shape analysis could also be used in the case of other types of detectors to analyse random pile up. This is outside the scope of this paper.

In order to test the capabilities of the GET electronics with respect to the reconstruction of the input charge distribution, we performed tests with an external pulse generator (voltage) applying signals $v_{in}(t)$ on a capacitor C_{in} connected to a GET channel input. The input current for the channel is $i_{in}(t) = C_{in} \cdot \frac{dv_{in}(t)}{dt}$.

This section describes a simple procedure to build an estimate of this input current from the measured samples.

4.1. Reconstruction principle

The measured output sample $s_{out}(t)$ results from the convolution of the input signal $i_{in}(t)$ and the electronics response function for the channel $h(t)$. If the response function is known, it is possible to deduce the shape of the input signal from the output sample. Since the output sample is measured in *coder units* (c.u.) and the input signal is a current (nA), the response function considered here is expressed in c.u./nA.

In order to deal with convolutions/deconvolutions, we use the Fourier transforms of the signals:

$$I_{in}(f) = \frac{S_{out}(f)}{H(f)}$$

where the capital letters denote the Fourier transforms of the corresponding functions in lower case, and f is the frequency.

Since we work with finite digitized samples, we need a digitized estimate of the response function, and we reconstruct a sampled estimate of the input signal:

$$I_{in}[k_f] = \frac{S_{out}[k_f]}{H[k_f]}$$

where the bracket notation indicates the sampling in frequency space³: the frequency is $f = k_f \cdot \Delta f$ where k_f is an integer in the range $[0; n/2]$ and $\Delta f = 1/(n \cdot \Delta t)$.

³ For a real sample with n values and a sampling time Δt , the frequency digitization step of the FFT (fast Fourier transform) is $\Delta f = 1/(n \cdot \Delta t)$ and the maximum frequency is $f_{max} = 1/(2 \cdot \Delta t)$. The FFT (discrete) components for $f=0$ and $f=f_{max}$ are pure real numbers and the $n/2 - 1$ components for frequencies $f = k_f \cdot \Delta f$, with $k_f=1$ to $k_f = n/2 - 1$, are complex numbers.

In the analysis presented here, we used the Danielson–Lanczos algorithm [17] for the fast Fourier transform (FFT). For this FFT algorithm, the sampling size must be a power of 2 (which is achieved with the GET system). The use of a FFT has several consequences, because it considers all signals as periodic, while this is not the case of a real measurement:

- a signal at the end of an input sample may induce a response at the beginning of the output sample;
- despite the corrections applied to the measured samples described in previous sections, a discontinuity may exist between the end and the beginning of the sample, resulting in artificial high frequency components of the frequency spectrum.

4.2. Response function estimate

As mentioned, in order to reconstruct the effective input signal from the measured output sample, one needs to know the response function.

The first option is to derive the response function from the electronics design. An approximate analytical function is given by the chip developer (P. Baron):

$$h(t) = A \cdot \exp\left(-3\frac{t}{\tau}\right) \cdot \left(\frac{t}{\tau}\right)^3 \cdot \sin\left(\frac{t}{\tau}\right)$$

where A is related to the amplification gain and τ is the peaking time parameter. Unfortunately, this analytical response function is only a rough approximation, and it is not accurate enough when performing the deconvolution of output samples. This function will not be discussed further.

4.2.1. Empirical response function

Another alternative is to build an estimated response function from a known input signal. In Fourier space: $H_{est}[k_f] = \frac{S_{out}[k_f]}{I_{in}[k_f]}$. Such a response function is distorted by the noise of both the output and the input signals. In order to reduce this effect, the estimated response functions presented in this analysis are the result of average signals over 1000 events (see Fig. 14):

$$H_{est}[k_f] = \frac{\langle S_{out}[k_f] \rangle_{n_{evt}=1000}}{\langle I_{in}[k_f] \rangle_{n_{evt}=1000}}$$

It should be noticed that there is some residual noise in the estimated response function. But since this noise is significantly reduced with respect to a single event and this function is used in the deconvolution of output samples on an event-by-event basis, the noise after deconvolution

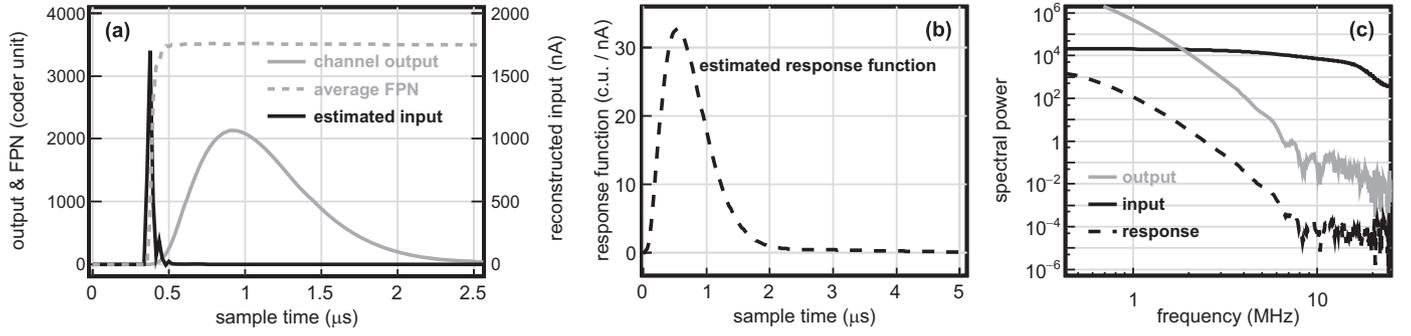


Fig. 14. Construction of the response function estimate from the AsAd pulse generator. This example is built with a write (sampling) frequency $F_W = 100$ MHz, a peaking time $\tau = 502$ ns, an input dynamic range of 120 fC and a pulse amplitude $V_{in} = -650$ mV. The channel input results from the derivative of the FPN channels. The response function (b) is the deconvolution of the output sample from the input signal (a). The samples are averaged over 1000 events. For the illustration, the horizontal scale is different for left and centre plots. Plot (c) shows the power spectrum (see Section 4.4 for more details) of the different samples to illustrate the residual noise at high frequencies.

lution is dominated by the noise level of the single output sample. This will be discussed in Section 4.3, but at this stage, the average on 1000 events is sufficient.

The most efficient way to build the estimated response function for all channels of an AsAd board is to perform measurements in *functionality test mode*:

- the AsAd pulser provides a step pulse on a 100 fF capacitor for each channel;
- in this mode, the FPN channels can directly measure the input pulse, bypassing the shaping elements of the electronics (CSA, PZC and SK filter, see Section 2).

The input signal can be estimated as the derivative of the average of the 4 FPN channels (see Fig. 14a) for one AGET chip (64 channels):

$$i_{in}(t) = \frac{C_{in} \cdot V_{in}}{F_{max}} \cdot \frac{df(t)}{dt}$$

where C_{in} is the test input capacitor (100 fF in functionality test mode), V_{in} is the AsAd generator pulse height, $f(t)$ is the average of the FPN channels, and F_{max} is the asymptotic value of the measured FPN signals, which reflects the voltage to coder units conversion factor. The resulting response function $h_{AsAd}(t)$ is shown in Figs. 14b and 15.

It is also possible to estimate the response function with an external pulse generator signal. This requires to design a circuit connected to the AsAd board input, embedding the capacitor in order to inject the charge to the requested channels. In this case, the run is performed in a standard acquisition mode, and there is no measurement of the input signal.

The pulse applied to the capacitor is a voltage step, with a rise time of 10 ns (10–90% of the amplitude). For the deconvolution of the measured output samples, we use the following function for the input signal:

$$V_{in}(t) = \frac{V_0}{2} \cdot \left(1 + \text{Erf} \left(\frac{t - t_0}{\delta t} \right) \right)$$

where V_0 is the pulse amplitude, t_0 the pulse start time (with respect to the sample time window) and δt the width of the transition. The pulse rise time does not correspond exactly to δt , but since it is smaller than the sampling period, the resulting response function is relatively insensitive to the exact δt value. The corresponding input current is then:

$$i_{in}(t) = \frac{V_0 \cdot C_{in}}{\delta t} \cdot \exp \left(-\frac{(t - t_0)^2}{2 \cdot \delta t^2} \right)$$

The resulting response function $h_{Ext}(t)$ is compared to the response function built using the AsAd pulser in Fig. 15. Since the capacitor value is not precisely known in the injection circuit we used with the

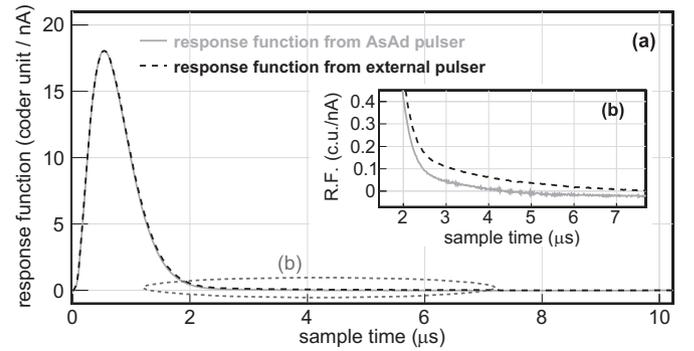


Fig. 15. Comparison of the response functions obtained with the AsAd pulse generator measurement $h_{AsAd}(t)$ (full grey line) or with an external pulser $h_{Ext}(t)$ (dashed black line). The inset is a zoom on the tail of the response functions that shows a slight discrepancy in the baseline restoration. The difference is less than 0.5% of the maximum amplitude. The response functions plotted here result from measurements with a write frequency $F_W = 50$ MHz, a peaking time $\tau = 502$ ns and a dynamic range of 120 fC (which corresponds to the maximum amplification from the preamplifier).

external pulser, the $h_{Ext}(t)$ response function has been scaled to match the $h_{AsAd}(t)$ function amplitude. Despite a small discrepancy in the tail of the response functions (less than 0.5% of the maximum amplitude), there is a good agreement in the resulting shapes.

4.2.2. Response function decomposition

The response function can be decomposed into 4 terms, corresponding to the main components of the amplification and coding chain:

$$h(t) = \frac{1}{C_g} \cdot r_{shaper}(t) \cdot 2 \cdot K$$

where $\frac{1}{C_g}$ is the amplification from the CSA (preamplifier) with gain capacitor C_g , r_{shaper} is the response function from the shaping stage, the factor 2 is for the $\times 2$ gain amplifier and K is the amplitude to coder units conversion factor from the ADC (in c.u./V). The only selectable parameter for a channel is the gain capacitor C_g .

It should then be possible, in principle, to define a *reduced* response function that is gain independent:

$$h_0(t) = r_{shaper}(t) \cdot 2 \cdot K = C_g \cdot h(t)$$

The *reduced* response function is illustrated in Fig. 16 for different values of the gain capacitor. Since the gain capacitor values are known with a precision of about 10%, the functions have been scaled to match their amplitude. A small discrepancy can be observed, on the order of 2% of the maximum amplitude, and 7% of the integral.

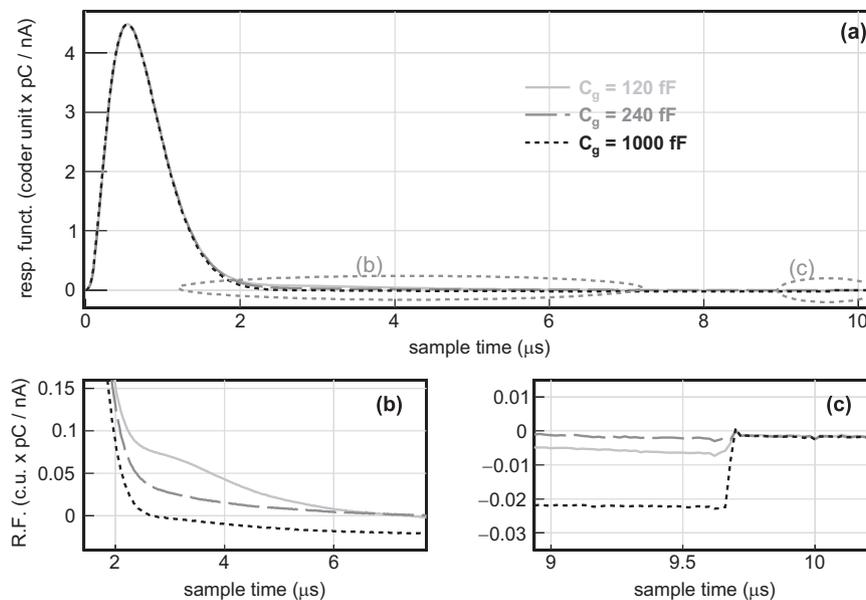


Fig. 16. The top figure (a) shows the reduced response functions built using an external pulse generator for different gain values of the preamplifier (with CSA input capacitors $C_g=120$, 240 or 1000 fF). The bottom pictures are zoomed regions of the top plot. Plot (b) shows a discrepancy between the response functions. The difference is of the order of 2% of the maximum amplitude. Plot (c) shows the effect of the discontinuity between the end and the beginning of the measured samples. This is due to an imperfect baseline restoration in the amplification and shaping chain. The functions plotted here result from measurements with a write frequency $F_W = 50$ MHz, and a peaking time $\tau = 502$ ns.

4.2.3. Calibration issues

For a full system with many channels, the response function should be estimated for every channel with a well-controlled input pulse and a precisely known input capacitor. Ideally, the same capacitor should be used for each channel, in order to ensure good gain matching across all channels.

This can be achieved easily with an external pulse generator and an external capacitor. But this requires separate measurements for each channel.

The alternative is to use the AsAd pulse generator, in functionality test mode, but the capacitor precision is of the order of 10%, and may change from one channel to another. As a result, this procedure leads to only an approximate gain matching.

In the following analysis, since we focus on processing at a single channel level, it will be performed using the response function estimated from an external pulser signal measured with the same channel gain as the signal to be analysed.

4.3. Signal reconstruction

The reconstructed input signal $i_{rec}(t)$ (current) is estimated on an event-by-event basis from the measured output sample $s_{mes}(t)$ (a single detector channel for the processed event). The deconvolution is performed on the Fourier transforms in the frequency space:

$$I_{rec}(f) = \frac{S_{mes}(f)}{H(f)}$$

The high frequency components of the response function being very small (see Fig. 14c), the division in frequency space induces an increase of high frequency noise for the reconstructed signal estimate. As a consequence, it is necessary to introduce a low-pass (numerical) filter $\Phi(f)$ in the reconstruction process:

$$I_{rec}(f) = \frac{S_{mes}(f)}{H(f)} \cdot \Phi(f)$$

Such a filter has to be adapted to the effective noise level of the measurement. The consequence of the low-pass filter is a *smoothing* of the signal, because high frequencies that are responsible for sharp variations are attenuated, as shown in Fig. 17. For this work, we tried

low-pass filters of orders 4–8.⁴ The higher order filters result in a sharper attenuation of high frequencies. A detailed study of the filtering is out of the scope of this paper. In particular, it depends on the expected shape of the signals to be measured.

4.4. Reconstruction capabilities

As mentioned previously, the reconstruction of the effective input charge distribution requires to filter the deconvoluted signal. The quality of the result depends on several factors, either on the electronics side (such as the write frequency and the peaking time of the shaper), or on the input signal side (such as the amplitude or the total charge deposit).

The following characterization has been performed with a 50 fC charge deposit (a 12.5 mV pulse amplitude and a 4 pF input capacitor). This approximately corresponds to the expected signal for a 2 mm track portion (the size of a pixel in the ACTAR TPC detector) for a 1 MeV proton in P10 gas at 1 bar, assuming a multiplication factor of 500 of primary ionization electrons.

4.4.1. Reconstruction fluctuations

The filtering should be studied with data taken with the detector in order to perform measurements in real noise conditions. Nevertheless, we propose here a simple characterization of the reconstruction quality for a low-pass filter.

The injected signal is a Gaussian shape current, with a constant integral (the total charge Q_0) and a width w_0 (FWHM), from a programmable pulse generator. The deconvolution of the response function and the filtering are applied on the measured output sample. The resulting reconstructed signal is fitted with a Gaussian function.

The average deviation from the fit (over the N samples of the signal) gives a quantity related to the reconstruction fluctuations:

⁴ The transfer function for a 1st order low-pass filter is $H(f) = \frac{1}{1+j\frac{f}{f_c}}$ where f_c is the cut frequency, and for a 2nd order filter: $H(f) = \frac{1}{1 - \left(\frac{f}{f_c}\right)^2 + j\frac{1}{Q}\frac{f}{f_c}}$ where Q is the filter quality (we used $Q = \sqrt{2}/2$ which is the highest value for a filter without resonance). Higher order filters are built by combination of 1st and 2nd order filters.

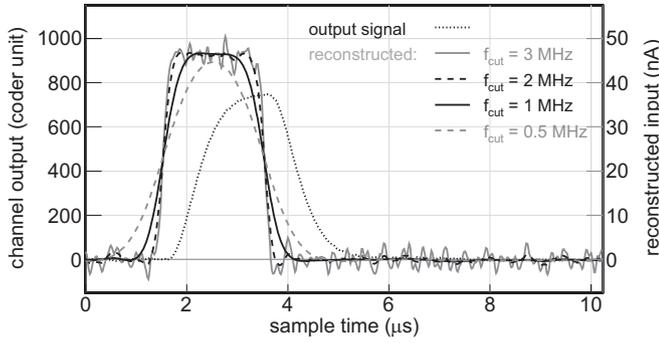


Fig. 17. Comparison of the effect of filtering on the reconstructed signal. The input signal is a 2 μs square current, and the output signal from the GET channel (dotted black line) has been measured with a write frequency $F_W = 100$ MHz, a peaking time $\tau = 502$ ns and a channel input capacitor $C_g = 240$ fF. The reconstructed input signals (full and dashed lines, grey and black) have been obtained with a 6th order filter with different cut frequencies. The curves correspond to an average of 100 events.

$$\Delta i = \sqrt{\frac{1}{N} \sum_{k_l=0}^{N-1} (i_{rec}[k_l] - i_{fit}[k_l])^2}$$

We define the corresponding *charge fluctuations* over the sample: $\Delta Q = T \cdot \Delta i$ where $T = N \cdot \Delta t$ is the total sample time (Δt is the sampling period), and the ratio:

$$R_Q = \frac{\Delta Q}{Q}$$

This ratio measures the relative importance of the *charge fluctuations* with respect to the signal total charge. It is independent of the shape of the charge distribution: this ratio is the same for a narrow signal with a large amplitude or a broad signal with a small amplitude.

The reconstruction method requires a compromise between the

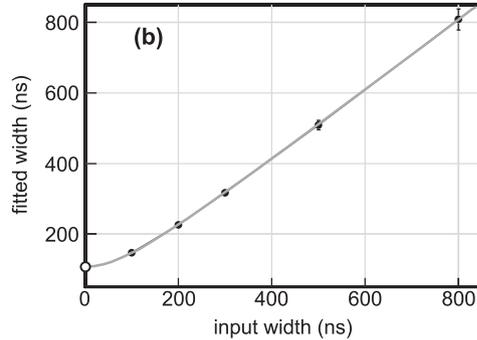
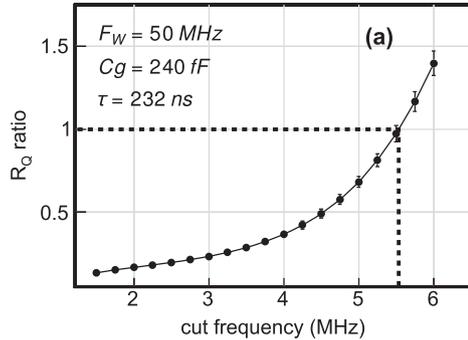


Fig. 18. The left figure (a) shows the behaviour of the R_Q ratio (that does not depend on the Gaussian input signal width) when varying the cutoff frequency f_c (in the case of a 2kth order filter with $k=3$). The points correspond to the average for 100 events, and the error bars indicate the dispersion of the results (standard deviation). The dashed line illustrates the filter parameter f_c for $R_Q=1$. With the resulting filter parameter, the reconstruction procedure is applied for events with various input widths (each point is the average of 100 events). The fitted width, on the right plot (b), can then be extrapolated for a signal with a null width.

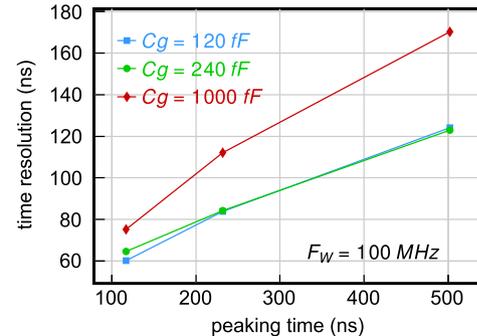
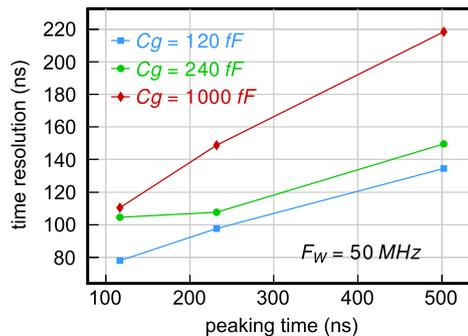


Fig. 19. Summary of the estimated time resolutions for various configurations of the electronics. The resolutions presented here have been estimated with the procedure described in the text (Section 4.4.2) for a total input charge $Q \approx 50$ fC and a condition for charge fluctuations $R_Q < 1$.

high frequency noise and the distortion of the real signal. In this section, we define an upper limit for R_Q to set the filtering parameters, and estimate the reconstruction characterization under these conditions.

The numerical filter used is a low-pass filter of order $2k$. We used an even order filter since a numerical filter of the first order $\Phi_1(f, f_c) = \frac{1}{1+jf/f_c}$, with cut frequency f_c , induces a shift of the signal. A filter such as $\Phi_{2k}(f, f_c) = \Phi_k(f, f_c) \cdot \Phi_k(f, -f_c)$ solves this problem (the transfer function has no imaginary component any more).

4.4.2. Time resolution: signals separation capabilities

In this section, we denote as time resolution the minimum time difference between two input signals (instantaneous charge deposits) that still allow a separation of the two signals in the reconstructed input sample. The time resolution that can be achieved in the reconstruction procedure is estimated here with the condition $R_Q < 1$. This condition defines the f_c parameter of the filter (see Fig. 18-left).

The filter is then applied to the reconstruction of input events with variation of the injected signal width w_{in} . The reconstructed signal is computed and fitted for each event. Fig. 18-right shows the fitted width w_{rec} as a function of the input width. The time resolution for the reconstruction is then the extrapolation of the fitted width for $w_{in} = 0$. The fit function is $w_{rec} = \sqrt{w_0^2 + w_{in}^2}$, where the parameter w_0 is the reconstructed signal width for $w_{in} = 0$.

This procedure has been applied for various configurations of the electronics: write frequency, shaper peaking time, gain capacitor. The results are summarized in Fig. 19. As expected, the reconstruction is better (smaller time resolution) with higher gain values (lower gain capacitors), and with smaller peaking time. This latter is not straightforward since two effects are related to the peaking time:

- a larger peaking time induces smaller high frequency components of

the response function, and thus larger high frequency noise in the deconvolution, which creates the need for more filtering;

- the amplitude of the output signal is at first order constant when changing the peaking time of a channel; thus a larger peaking time gives a larger output signal integral, which may increase the signal-to-noise ratio.

This figure shows that a time resolution of about 100 ns can be achieved with a write frequency of 50 MHz, and almost down to 50 ns at 100 MHz, depending on the selected peaking time. This resolution corresponds to approximately 5 time buckets.

An illustration of the separation of two point-like charge deposits is given in Fig. 20.

4.4.3. Timing precision

In a standard detector electronic chain, the preamplifier signal is used on the one hand as input of a fast amplifier to get a precise timing from the discriminator, and on the other hand as input to a spectroscopy amplifier (and shaper) for a precise measurement of the amplitude. In the case of a sampled signal, such information is computed from the analysis of the sampled data. A commonly used algorithm to get the signal time is a digital CFD (constant fraction discriminator). The information registered is then a single amplitude value and a single time value for a signal that is supposed to be very short.

Since the purpose of the signal reconstruction procedure presented here is to build an estimate of the *time distribution* of the input charge, the relevance of a *single* time information is limited. The purpose of this section is then not to perform a full study of this timing precision, but only to provide estimates for the precision of the timing information that can be obtained when using the reconstruction procedure (resulting in a time distribution), by comparing it to digital CFD timing (resulting in a single time value).

In both cases, the precision is estimated from the fluctuations of the signal timing measured for a set of similar events. When using an external pulse generator for such a measurement, there is no absolute time reference. Therefore, we performed two types of tests: one with the AsAd pulse generator in *functionality test* mode for which the FPN channels signal, reproducing the input pulse, is used to estimate the effective time of the pulse, and one with an external pulse generator, without time reference.

Timing with AsAd pulse generator. With the AsAd pulse generator in *functionality test* mode, the average of the FPN channels signal is fitted with an error function:

$$f_{FPN}(t) = \frac{a}{2} \times \text{Erfc}(t - t_{FPN}) + b$$

where t_{FPN} is the estimated pulse time.

For each event, the time t_{REC} from the reconstruction procedure is the centre of a Gaussian fit to the reconstructed signal, and time t_{CFD}

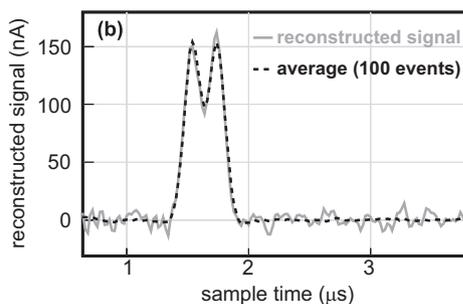
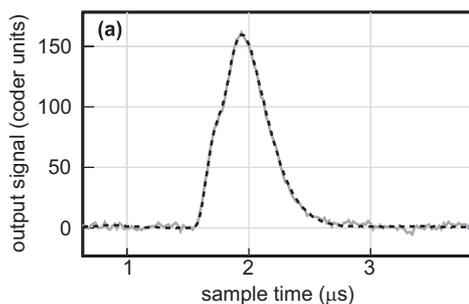


Fig. 20. Illustration of the separation of two pulses in the signal reconstruction: (a) the output signal from the tested channel (including corrections presented in Section 3); (b) the reconstructed input. The grey curves correspond to a single event, while the dashed black curves are the averages over 100 events. The sample is measured with a 232 ns peaking time, a write frequency $F_W = 50$ MHz and a gain capacitor $C_g = 1$ pF (most unfavourable illustration case). The samples length is 10.24 μ s, but the plots are zoomed on the interesting region. The total charge deposit is approximately 50 fC (25 fC for each pulse), and the pulses separated by 200 ns are generated with a 2.5 ns rise time for the input generator.

is obtained by applying the CFD algorithm to the output signal (see Fig. 21). The CFD algorithm parameters (constant fraction and delay) have been set approximately in order to minimize the fluctuations, but no full optimization procedure was performed.

Since in this case, the pulse is very short (few ns rise time) and the signal-to-noise ratio is large, the measurement gives an estimate of the intrinsic timing precision. The comparison of the timing precision (dispersion of the estimated signal time) of the reconstruction procedure and the CFD algorithm is illustrated in Fig. 22, for a $F_W = 50$ MHz write frequency and a $\tau = 502$ ns peaking time. Despite the results from both procedures have a quite similar FWHM, the RMS is better for the CFD algorithm in this case (short pulse, low noise).

Timing with a signal distribution from an external pulse generator. The same comparison has been performed with a signal from an external pulse generator, in order to perform the measurement in more realistic conditions for a signal from a TPC. The injected signal is a Gaussian distribution, with a total charge of 50 fC.

Since with the external pulser, there is no access to any time reference, the fluctuations in the measured time include the deviation due to the jitter of the system trigger with the asynchronous external trigger. Assuming that the pulse is randomly distributed during the write cycle period, a deviation of $\Delta t/\sqrt{12}$ is expected, where Δt is the write period, which is ~ 2.9 ns for a write frequency $F_W = 100$ MHz or ~ 5.8 ns for $F_W = 50$ MHz. This deviation due to the trigger jitter is summed quadratically to the effective timing precision.

Fig. 23 gives an example of the behaviour of the measured timing

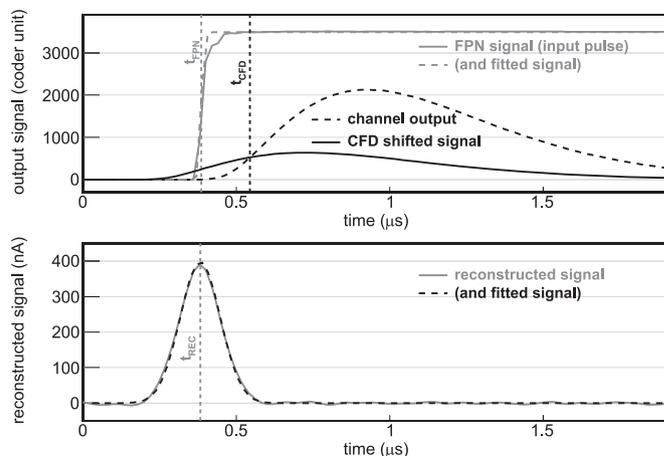


Fig. 21. Signal time measurement with the AsAd pulse generator. In the upper plot, the reference time t_{FPN} is estimated from the fit (dashed grey line) of the FPN channel signals (full grey line). The output signal (dashed black line) is scaled and shifted (full black line) for the CFD algorithm (for the illustration the modified signal is not inverted and the zero crossing is replaced by the intersection of the initial and modified signals) that give the CFD timing t_{CFD} . The lower plot shows the reconstructed input signal (full grey line) and the Gaussian fit (dashed black line) used to estimate the signal time t_{REC} .

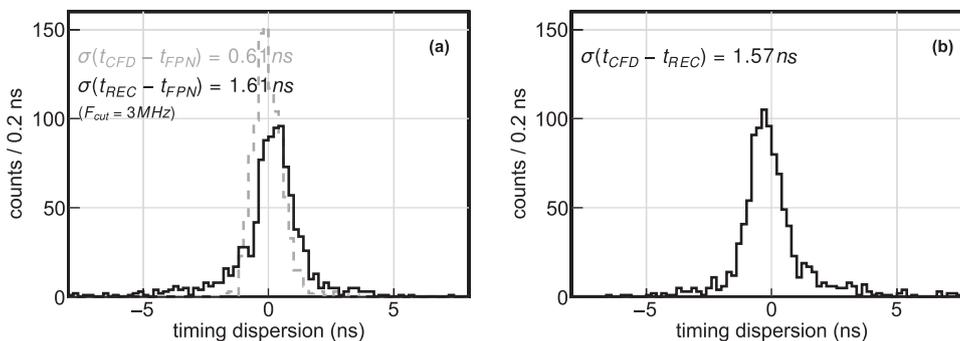


Fig. 22. The left plots (a) illustrates the dispersion of the events time estimate with respect to the FPN timing considered as reference, for the reconstruction procedure (black) and for the CFD algorithm (dashed grey), for a measurement with a write frequency $F_W = 50$ MHz and a peaking time $\tau = 502$ ns. The right plot (b) shows the dispersion of the time difference for the two time estimate methods (reconstruction and CFD). A cut frequency $f_{cut} = 3$ MHz was used for the filter in the reconstruction of the input signals. The quoted values correspond to root mean square of the distributions. If the fluctuations for the different times are not correlated, this leads to intrinsic timing fluctuations: $\sigma(t_{REC}) = 1.53$ ns, $\sigma(t_{CFD}) = 0.35$ ns and $\sigma(t_{FPN}) = 0.50$ ns.

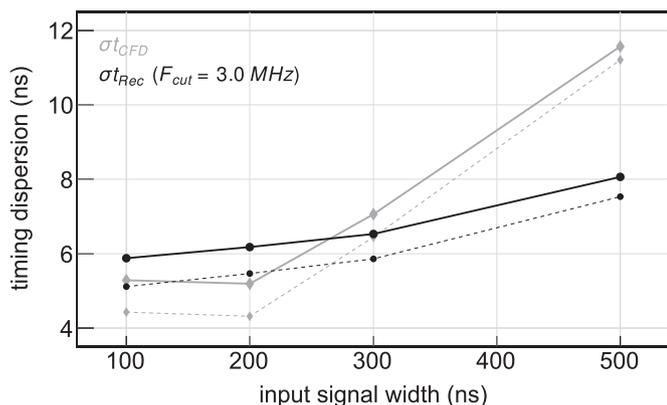


Fig. 23. Timing precision for events with a Gaussian input signal for various widths (FWHM) of the signal. This plot is for measurements with $F_W = 100$ MHz and $\tau = 232$ ns. The full lines are the observed timing dispersion for the reconstruction procedure (black circles) and for the CFD algorithm (grey diamonds). The dashed lines correspond to the observed timing dispersion, corrected for the jitter of the internal trigger with respect to the asynchronous external pulser signal.

fluctuations with the width of the input signal distribution. As expected, when the input charge distribution becomes larger, the reconstruction procedure becomes more appropriate than the CFD algorithm, which works ideally with instantaneous signals.

Remarks about timing precision measurement. As mentioned previously, the timing precision is mentioned here despite the fact that a comparison between the CFD algorithm and the reconstruction procedure is limited, because the first one is well suited for an instantaneous charge deposit while the latter is proposed for a charge deposit with a time distribution. The behaviour has been presented in few configurations, but many parameters play an important role in the results one can obtain:

- the electronics configuration parameters such as the write frequency and the peaking time;
- the input signal parameters: the amplitude (or signal-to-noise ratio) and the width;
- the reconstruction parameters such as the filter characteristics for the reconstruction and the CFD parameters (delay and constant fraction).

For the reconstruction procedure, a lower filter cut frequency will result in a more accurate time precision. Since a lower cut frequency induces a larger distortion of the reconstructed signal (see Fig. 17), the analysis parameters may result from a balance between the signal-to-noise ratio and the reconstructed distribution precision, depending on the purpose of the analysis.

4.4.4. Reconstruction precision

The precision on the time distribution extracted from the reconstructed signal is another important issue in the reconstruction procedure. In order to illustrate this, we injected a square signal in the channel input: the pulse generator was set to provide a ramp with a constant slope during a time w_{in} corresponding to the signal width. This signal was applied on the input capacitor, resulting in a square charge at channel input. The pulse amplitude was kept constant in order to inject a constant total charge Q_0 (which is approximately 100 fC). Fig. 24a shows examples of single-event reconstructions for different input signal widths.

The reconstructed signals are fitted with the convolution of a square and a Gaussian functions, $f_{rec}(t)$:

$$f_{rec}(t) = \int i_0(t - t_0, Q, w) \cdot \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(t - t_0 - t')^2}{2\sigma^2}\right) dt' + b$$

where $i_0(t - t_0, Q, w)$ represents the effective input signal (centred at t_0):

$$i_0(t - t_0, Q, w) = \begin{cases} Q/w & \text{if } -w/2 < t - t_0 < w/2; \\ 0 & \text{otherwise} \end{cases}$$

and t_0 , w and Q are the central time, the width and the total charge deposit of the square signal, σ is the Gaussian smoothing width and b is a constant offset.

Measurements were performed with different input signal widths. A fit procedure was applied on an event-by-event basis. Figs. 24b and c show the average fitted values for the total charge Q and for the ratio of the fitted width and the input width w/w_{in} .

In the case of the total charge Q , the expected result is a constant value. The observed deviations are of the order of 1%. For the w/w_{in} ratio, the expected value is 1. As shown in Fig. 24c, a precision of about 1% is also achieved, except for the lowest width values. The largest deviation at lower width can be explained by several factors. First the fit function, with a Gaussian smoothing, is only a simple approximation. Indeed, the filtering in the discrete Fourier space induces a small $\sin(t)/t$ effect. As a result, a small negative component is present on each side of the square signal that is not taken into account with this function. In addition, the short width corresponds to fast changes in the pulse generator signal. Due to the bandwidth of the pulse generator, a small distortion is observed for a few ns at the beginning and end of the square pulse, which results in a slightly larger signal. Nevertheless, the small signal broadening is symmetrical with respect to the centre of the charge distribution that is not affected, as shown in Fig. 24d: the variations are in the order of 2–3 ns. This effect becomes significant enough for short pulses.

The same analysis has been performed for a write frequency $F_W = 100$ MHz, for various values of the channel gain ($C_g = 120$ fF,

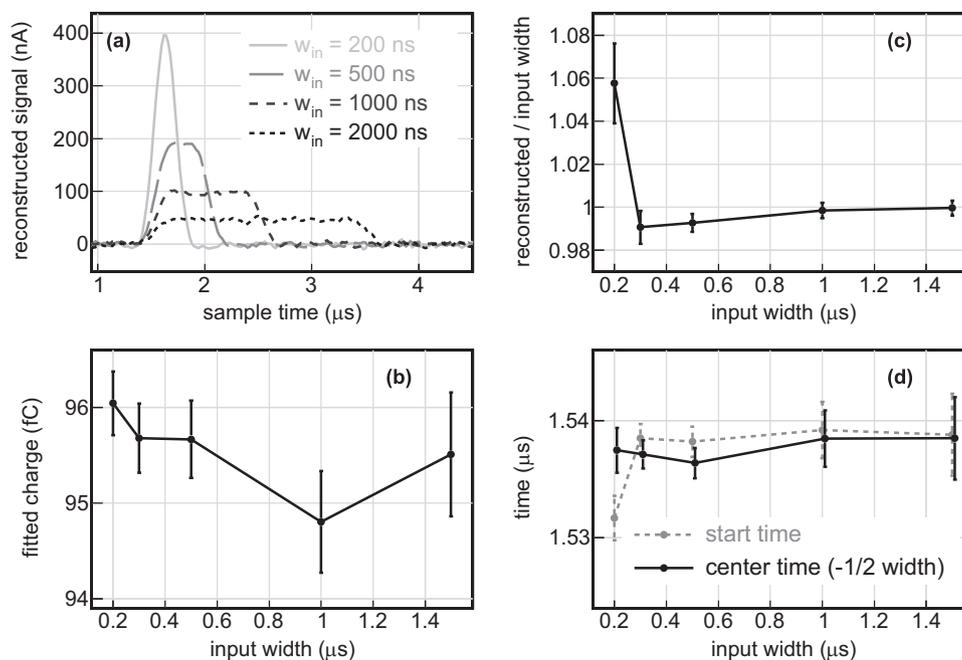


Fig. 24. The plot (a) shows some reconstructed events for a square input signal with different input widths. For each displayed event, the injected total charge is the same. The reconstruction has been performed with the procedure described in Section 4.4.2. The plots (b) and (c) show the average values of the total charge Q and the ratio w/w_{in} of the fitted and input widths (see text for the description of the fit function). The plot (d) shows the estimated start time t_0 in grey. The lower value of the first point ($w_{in} = 200$ ns) corresponds to the larger value of the estimated width in plot (c). The width of the distribution is overestimated, but the centre is anyway constant (see text for details). The points result from the average of 100 events for each value of the input width w_{in} , and the uncertainties are the dispersion of the results. The figures correspond to measurements with a write frequency $F_W = 50$ MHz, a gain capacitor $C_g = 1$ pF and a peaking time $\tau = 232$ ns.

240 fF and 1 pF) and of the peaking time ($\tau = 117, 232$ and 502 ns). In all cases, the same effects are observed and the precision of the fit parameters of the reconstructed events is of the order of 1%.

5. Conclusions

We have analysed a series of output data samples from the GET electronics channels in order to evaluate the required processing of these data before they may be used for interpretation of a real detector signal.

We have shown that a first correction should be applied in order to improve the intrinsic resolution of the measured signals. This has to be done in two steps: to correct for the average baseline distortion and for the systematic variation depending on the cells of the switched capacitor array. Two alternative methods can be used, either using the information from the *fixed pattern noise* channels, or defining a correction pattern for each channel. The latter gives slightly better results but the improvement on the intrinsic noise is of the same order of magnitude.

We also propose a simple method to build an empirical estimate of the response function of the GET channels, using pulse generator events. This allows us to reconstruct the effective input signal, which may be used for signal shape analysis. A very basic filtering procedure has been applied in the reconstruction process, in order to estimate the quality of the reconstructed input. As a result, an intrinsic time resolution of about 5 sampling time steps can be achieved. The analysis of the signal shape, based on an event-by-event fit to the reconstructed signal, shows that a precision of about 1% can be achieved with both a simple filtering method and a simple fit function.

The analysis procedures presented in this paper have been developed in standard modes of the GET electronics. They may also be applied for the 2P-mode (short-lived implantation and decay events, implemented for very exotic radioactivity studies), in which the 512-cells SCA memory is separated into two 256-cells memory halves. This requires some slight modifications of the procedures described above.

Acknowledgement

The GET project (General Electronics for Time projection chambers) was funded by the *Agence Nationale de la Recherche* (ANR) (Grant no. ANR-09-BLAN-0203) in France, on “Programme blanc” edition from 2009 for the development of AGET chips, AsAd boards and MuTANt. The research leading to these results have received funding from the European Research Council under the European Union’s Seventh Framework Program (FP7/2007-2013)/ERC grant agreement no 335593 and from the Conseil Régional d’Aquitaine (grant no 2014-1R60402-00003319).

References

- [1] E. Pollacco, et al., *Phys. Procedia* 37 (2012) 1799.
- [2] C.-E. Demonchy, et al., *Nucl. Instrum. Methods A* 583 (2007) 341.
- [3] M. Caamaño, et al., *Phys. Rev. Lett.* 99 (2007) 062502.
- [4] T. Roger, et al., *Phys. Rev. C* 79 (2009) 031603 (R).
- [5] B. Blank, et al., *Nucl. Instrum. Methods B* 266 (2008) 4606.
- [6] B. Blank, et al., *Nucl. Instrum. Methods A* 613 (2010) 65.
- [7] J. Giovinazzo, et al., *Phys. Rev. Lett.* 99 (2007) 102501.
- [8] P. Ascher, et al., *Phys. Rev. Lett.* 107 (2011) 102502.
- [9] L. Audirac, et al., *Eur. Phys. J. A* 48 (2012) 179.
- [10] G.F. Grinyer, et al., Active Targets and TPC for Nuclear Physics Experiments Workshop, MSU, 2015 (<https://indico.fnal.gov/contributionDisplay.py?contribId=27&sessionId=4&confId=8976>).
- [11] D. Suzuki, et al., *Nucl. Instrum. Methods A* 691 (2013) 39–54.
- [12] T. Isobe, et al., Active Targets and TPC for Nuclear Physics Experiments Workshop, MSU, 2015 (<https://groups.nslc.msu.edu/hira/sepweb/presentation/isobe.pdf>).
- [13] M. Kurata-Nishimura, et al., Active Targets and TPC for Nuclear Physics Experiments Workshop, MSU, 2015 (<https://indico.fnal.gov/contributionDisplay.py?contribId=14&sessionId=7&confId=8976>).
- [14] T. Isobe, et al., in: 5th International Symposium on Nuclear Symmetry Energy NuSYM15, Krakow, 2015 (http://nusym15.if.edu.pl/contributions/contribution_67.html).
- [15] H. Sako, et al., Active Targets and TPC for Nuclear Physics Experiments Workshop, MSU, 2015 (<https://indico.fnal.gov/contributionDisplay.py?contribId=15&sessionId=7&confId=8976>).
- [16] S. Anvar, et al., in: Proceedings of the IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), vol. 745, 2011.
- [17] G. Danielson, C. Lanczos, *J. Frankl. Inst.* 233 (1942) 435.