SLAC-PUB-14975

Analysis of Full Charge Reconstruction Algorithms for X-Ray Pixelated Detectors

A. Baumbaugh, G. Carini, G. Deptuch, P. Grybos, J. Hoff, P. Maj, P. Siddons, R. Szczygiel, M. Trimpl, R. Yarema

Abstract–Existence of the natural diffusive spread of charge carriers on the course of their drift towards collecting electrodes in planar, segmented detectors results in a division of the original cloud of carriers between neighboring channels. This paper presents the analysis of algorithms, implementable with reasonable circuit resources, whose task is to prevent degradation of the detective quantum efficiency in highly granular, digital pixel detectors. The immediate motivation of the work is a photon science application requesting simultaneous timing spectroscopy and 2D position sensitivity. Leading edge discrimination, provided it can be freed from uncertainties associated with the charge sharing, is used for timing the events. Analyzed solutions can naturally be extended to the amplitude spectroscopy with pixel detectors.

I. Introduction

The goal of this work was to study charge reconstruction algorithms in pixel detectors with nominal pitch of 60 microns. The typical RMS spread of the photon charge has a Sigma of 10 microns[1]. The magnitude of the charge deposited is of the order of 2200 electrons (Monochromatic X-rays of 8keV in silicon, in our application x-ray photon correlation spectroscopy[2], produce 2200 electrons.), although the ability to detect smaller charges is generally desirable, and of course larger signals can occur. This charge is spread over up to four pixels, and charge reconstruction is in the summing nodes. The issue of charge sharing in pixilated photon detectors has been studied previously[3]. Several charge position reconstruction algorithms were considered, the goal being to detect hit photons with very high-efficiency and produce very few false hits. These two requirements put heavy constraints on methods, charge sensitivity, and noise requirements. The different algorithms and results are reported here. A secondary but no less important consideration is that any algorithm created must fit inside the pixel area, as each pixel readout requires its own electronics.

Manuscript received November 10, 2011. This work was supported by Fermi Research Alliance, LLC under Contract No. De-AC02-07CH11359 with the United States Department of Energy. Work at AGH University of Science and Technology was supported by the Polish Ministry of Science and Higher Education, project no. N515 243 037 in years 2009–2011.

A. E. Baumbaugh, IEEE member, is with Fermi National Accelerator Laboratory, Batavia, IL 60510 USA (telephone: 630-840-4044, e-mail: baumbaugh@fnal.gov).

G. A. Carini is with SLAC National Accelerator Laboratory, Menlo Park, CA 94025 (telephone: 650-926-5009, e-mail: carini@slac.stanford.edu)

G. W. Deptuch IEEE senior member, J. R. Hoff, M. Trimpl IEEE member, R. J. Yarema IEEE Life member are with Fermi National Accelerator Laboratory, Batavia, IL 60510 USA (telephone: 630-840-4659,2398,4162,4817 e-mail: deptuch@fnal.gov, jimhoff@fnal.gov, trimpl@fnal.gov, yarema@fnal.gov)

P. Grybos. P. Maj, R. Szczygiel are with the AGH University of Science and Technology, Department of Measurement and Instrumentation, Al. Mickiewicza 30, 30-059 Cracow, Poland (tel: +48-12-6173299, e-mail: pgrybos@agh.edu.pl, piotr.maj@agh.edu.pl, robert.szczygiel@agh.edu.pl,).

D. P. Siddons is with Brookhaven National Laboratory, Upton, NY 11973 USA (telephone: 631-344-2738, e-mail: siddons@bnl.gov).

II. Simulation Methods

The basic method used to perform the simulations for the pixel detector was to create an 8 x 8 pixel array, with each pixel being 80 microns x 80 microns, and randomly deposit an x-ray charge cloud within the central 6 x 6 pixel array, in this way all the charge would be contained within the 8 x 8 array. The individual charge gain for each channel was randomly distributed using a Gaussian distribution with a sigma of 5%, these gain variations were calculated for each simulation run. For each simulated event, Gaussian noise was added to each pixel, and then a Gaussian distributed charge cloud was deposited in the array. Finally the charge for both noise and signal was adjusted by the gain variations in each pixel. In order to minimize computing time, we created a lookup table for x-rays hitting a pixel on a grid that was .5 microns x .5 microns. Since the charge cloud has a Sigma of 10 microns, this .5 microns scale has little if any effect on the charge distribution. Simulation runs were done for both purely white Gaussian noise with high bandwidth and for Gaussian noise smoothed by the band-pass filtering in the electronics. For simulation results a hit was said to be valid if the reconstructed position was within one pixel dimension of the actual hit position. Detected hits that were not within prescribed distance of the actual hit are considered “extra hits”. Simulations were run varying energy deposition, gain variation, noise level, summing node threshold, and pixel threshold. Generally we chose the gain variations to be 5% and assumed that offsets were negligible and were set to zero. The summing node signal was the sum of the charge in each of the 4 pixels associated with that node or corner.

III. Algorithms Tested

A number of different charge reconstruction algorithms were run in an attempt to find algorithms which would be both efficient at reconstructing real hits and that would not generate false hits. Initially all of these simulations were run using wide bandwidth white Gaussian noise for each pixel.

Work supported in part by US Department of Energy contract DE-AC02-76SF00515.

SLAC National Accelerator Laboratory, Menlo Park, CA 94025

A. C4P1 Algorithm

The first simulation method run was C4P1. This is an analog compare four ways, up down left and right (C4) to determine which of the neighboring pixels has the larger signal, and requiring at least one summing node be above threshold (P1) see Fig. 1.

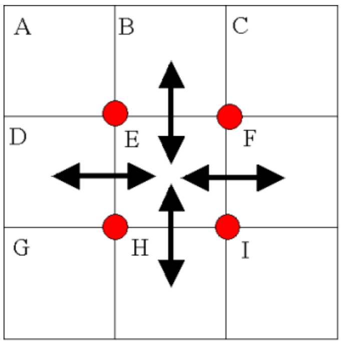

Fig. 1. C4P1 algorithm. Pixel E is said to be hit if it is greater than pixels B, D, F, and H, and if at least one of the red corner summing nodes is above threshold. Summing nodes exist at each junction of 4 pixels.

When simulated with no noise and no gain variations this method is nearly perfect, however any gain variation or noise causes this method to find many extra hits as can be seen in Fig 2. After studying many of the extra hit

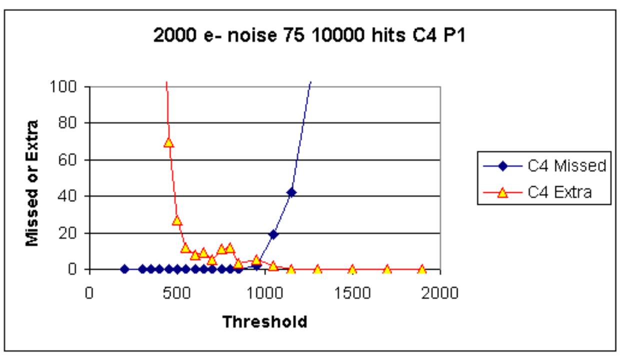

Fig. 2 C4P1 Summing node threshold scan. Note that there is no threshold that satisfies both no false hits and no missing hits. Lower thresholds cause extra hits and higher thresholds cause missing hits.

events it appeared as though requiring two summing nodes (P2) to be hit would increase the false hit rejection efficiency. However, an unanticipated effect of requiring two nodes to be hit, was to miss a large percentage of real hits. We therefore abandoned C4P1 and C4P2 methods.

0	0	0	0
0	2039	2163	0
0	2058	1626	0
0	0	0	0

78	-16	78	-25
1	113	9	91
28	107	1698	-82
-43	55	-32	-134

67	25	-122	-37
3	50	21	64
-49	46	1025	-99
-42	107	77	148

Fig. 3 Examples of extra hits for C4P1. Hits are shown as orange in the figure. Blue dots mark the position of the photon impact. The example on the left shows that even with no noise, the method has problems. Note most are on the diagonal which has no comparator.

As can be seen from the Fig. 3, the extra hits are caused by diagonal pixels.

B. C8P1 Algorithm

Failure of the C4P1 and C4P2 algorithms led to the second algorithm run, C8P1 shown in Fig 4. This algorithm requires a pixel to be larger than all its neighbors,

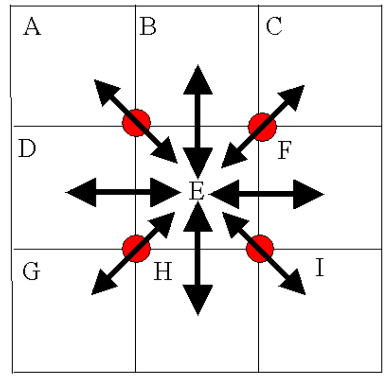

Fig. 4 C8P1 Algorithm. Pixel E is said to be hit if it is larger that all 8 of its neighbors AND at least one of the corner summing nodes is above threshold. If E is hit, then none of the neighboring pixels can be hit.

this requires eight comparators for the pixel and the pixel is also required to have at least one summing node above threshold. This method was studied extensively and shows promise at being both efficient for detecting real hits and proficient at rejecting false hits. As can be seen from Fig. 5, lowering the summing node threshold leads to producing extra hits. If the summing node threshold is raised too high real hits are missed. There is, however, a reasonable operating point for this method as long as the signal size is above about 1300 electrons, Fig 5.

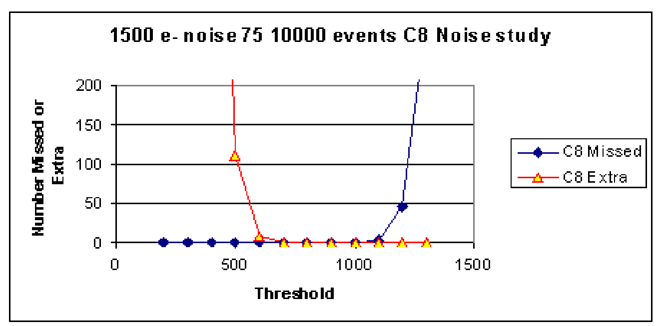

Fig. 5 C8P1 algorithm. Extra hits and missing hits as a function of summing node threshold. Above 700 e- and below 1000 e- is good. Below 700 e- we have extra hits and above 1000 e- we start missing real hits.

One concern with this method is the circuit area required for the four shared analog comparators for pixel-to-pixel comparisons, as well as the summing node comparators. It is being investigated whether the area within a pixel is sufficient to create this algorithm in a single pixel area. The maximum noise allowed for no missed hits and no extra hits as a function of threshold is shown in Figs 6 and 7. As can be seen from those figures, a signal of 1300 electrons with a threshold of 1000 electrons can tolerate a noise of about 100 electrons/pixel.

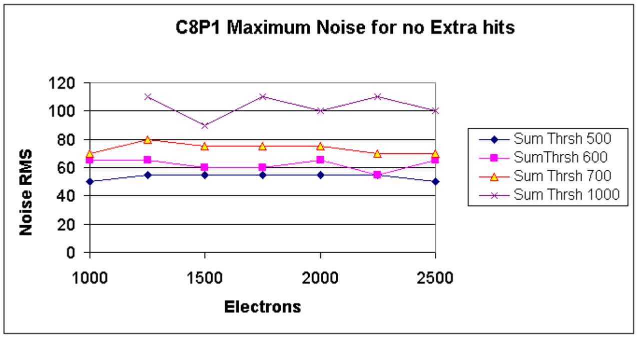

Fig. 6 Plot of maximum noise allowable for no extra hits as a function of signal size for different thresholds for C8P1 algorithm. For a threshold of 1000 electrons, we can tolerate about 100 electrons of noise.

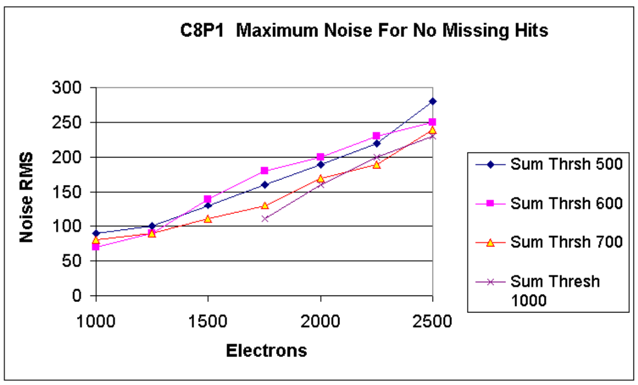

Fig. 7 Plot of maximum noise allowed for C8P1 to have no missing as a function of signal size for different thresholds. If we choose a threshold of 1000 electrons, we see that at a noise level of 100 electrons we need a signal size of about 1500 electrons to be fully efficient.

We also studied C8P2, which requires two summing nodes to be above threshold, but as before, this increases the number of missed hits and is therefore not desirable.

C. Summing Node pattern recognition P5 algorithm

We then considered a pattern recognition method, P5, based solely upon the pattern of summing nodes above threshold. As can be seen from Fig. 8, there is a limited number, 4, of possible summing node patterns for a single photon hit. While the pattern recognition algorithm is purely digital, the number of patterns and pixel to pixel data exchange may make this method too complicated to fit within a single pixel. It also appears that noise which can trigger a summing node or prevent a summing node from firing makes this pattern recognition algorithm inefficient for small signal sizes, see Fig 9.

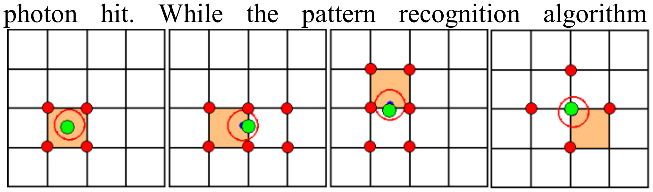

Fig. 8 P5 algorithm. Possible summing node patterns for single photon hits in ideal case. What we found when we simulated this was that noise would cause an extra node to fire or cause a node not to fire.

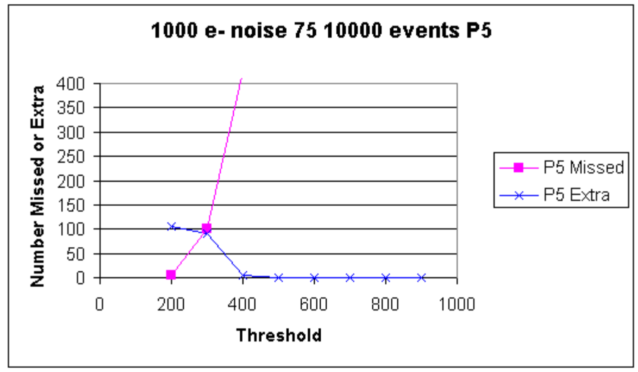

Fig. 9 P5 algorithm. Threshold scan for 1000 e- signal. Clearly there is no operating point. As we raise the threshold we start missing hits long before we reach the point where there are no extra hits.

However, at larger signal sizes this method does seem to have a valid operating point, see Fig. 10 and 11.

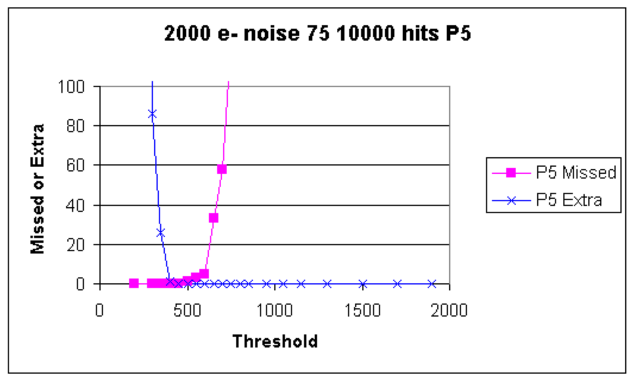

Fig. 10 P5 algorithm at a signal size of 2000 electrons has a narrow operating point at a threshold of about 450 electrons assuming a noise level of 75 electrons.

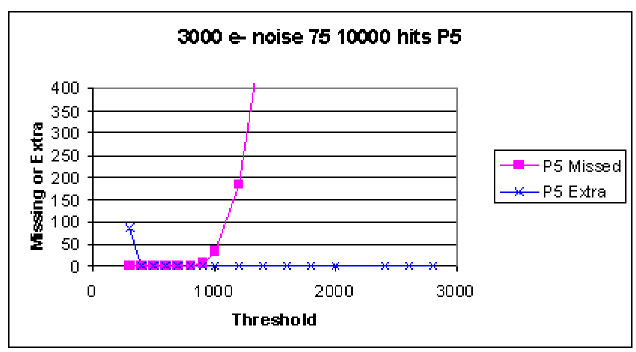

Fig. 11 P5 algorithm at a signal size of 3000 electrons and a noise of 75 electrons has a wide operating point from about 400 to 800 electrons for the summing node threshold.

We also studied an algorithm (P4) which looked only at 4 summing nodes in the 4 corners of a single pixel. P4 misses the hits that are shown in the rightmost of Fig. 8 where 5 nodes have fired giving the name to P5. While this was a simpler method to implement, it proved to miss too many real hits, and was discarded.

D. Pixel Pattern recognition algorithm PP

The final algorithm studied was a pixel pattern recognition algorithm, PP, which used a single comparator per pixel to determine that a pixel was above threshold as well as the aforementioned summing nodes. Fig. 12 shows possible pixel patterns which can be created by single photon.

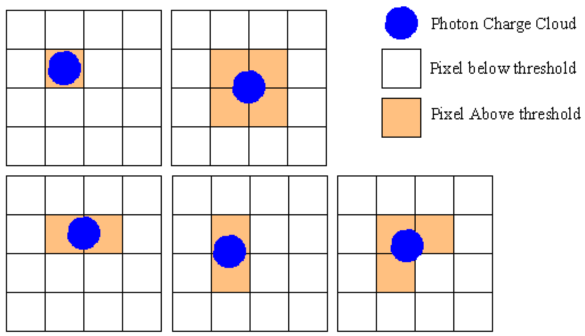

Fig. 12 PP algorithm possible pixel hit patterns. There are of course 4 different versions of the 3 pixel pattern only one is shown.

In this algorithm, the pixel is considered hit if it is above the pixel threshold and at least one of its summing nodes is above the summing node threshold. In studying the hit patterns for this pixel pattern method, we find that a sub set of patterns will cover all possible hit patterns. These are shown in figure 13. In this method the 2 parameters which must be studied are the summing node threshold and the individual pixel threshold.

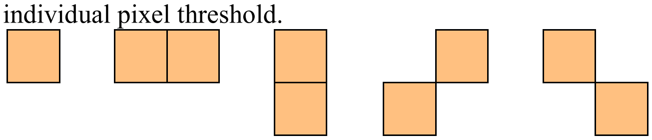

Fig. 13 PP algorithm these 5 patterns cover all possible hit patterns for a single photon. Note in all but the single hit pixel case, the position resolution is improved.

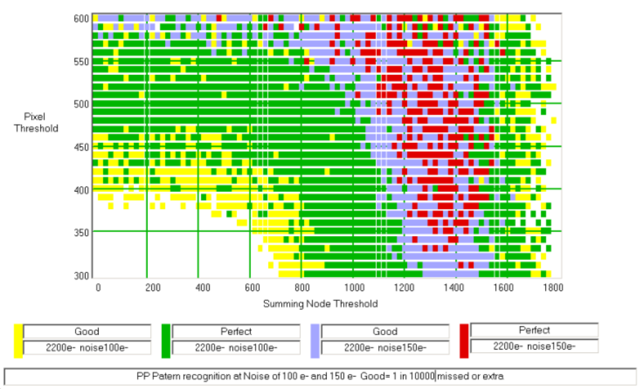

Fig. 14 PP algorithm efficiency scan for both pixel and summing node thresholds, signal size 2200 e-. Green and Yellow are perfect and good for 100 e- noise. Red and Blue are perfect and good for 150 e- noise. Good is for 1 part in 10,000 errors or better.

All possible hits are covered by one or more of the patterns shown here. This algorithm has a possible advantage in that the position resolution for some patterns is much better than non pattern methods, particularly for the diagonal patterns which can have the photon localized to a very small region. In studying this algorithm, we had to vary both the Summing node threshold and the individual pixel threshold. We also considered what was a "good" operating point and what was a "perfect" operating point. In the following graphs, "perfect" is missing no real hits and having no extra hits in 10,000 simulated events. "Good" is defined as having no more than 1 in 10,000 photons missed and no more than 1 in 10,000 events with an extra hit. We also did runs with this criteria changed to 1 part in 1000, both are shown. Only runs with signals of 1650 electrons and 2200 electrons, and for noise levels of 100 and 150 electrons are shown. Fig 14 shows the results for 2200 electrons with noise of 100 and 150 electrons. For a noise level of 100 electrons, there is a very broad operating point with a summing node threshold from 900 to about 1500 electrons. In these regions, virtually any pixel threshold from 300 to 550 electrons yields nearly perfect results. However, raising the noise to 150 electrons greatly reduces this region. With 150 electrons noise there is no region that is perfect and only a small region that is good at the 1 in 10,000 level. This smaller region has summing threshold from 1300 to 1450 electrons and pixel threshold from 450 to 550 electrons.

We also studied this algorithm assuming that we could tolerate 1 in 1,000 missed or extra hits. This relaxed performance requirement is shown in Fig 15.

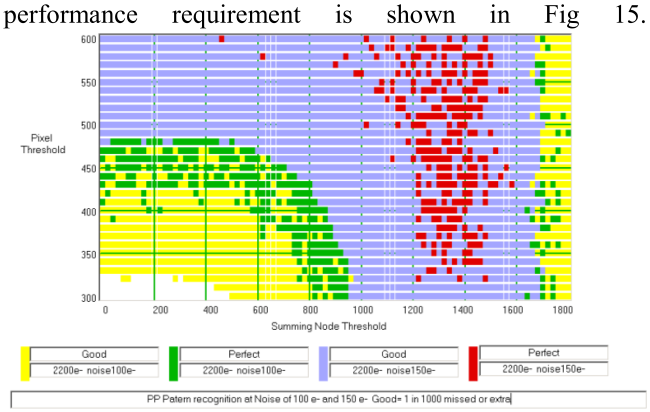

Fig. 15 PP algorithm efficiency scan for both pixel and summing node thresholds, signal size 2200 e-. Green and Yellow are perfect and good for 100 e- noise. Red and Blue are perfect and good for 150 e- noise. All are for 1 part in 10,000 errors. Red and Blue are perfect and good for 150 e- noise. Good in this case is for 1 part in 1,000 errors or better. Note the Green and Red do not change because they have no extra or missing in 10,000 and are perfect

Clearly, relaxing the performance requirement greatly increases the "good" region both for 100 electrons noise and 150 electrons noise. At 150 electrons noise the summing node threshold can be between 950 and 1650 electrons, and the pixel threshold can be just about any reasonable value. We also looked at the same method but at a signal of 1650 electrons and 1 part in 10,000 errors allowed. The results are shown in Fig 16.

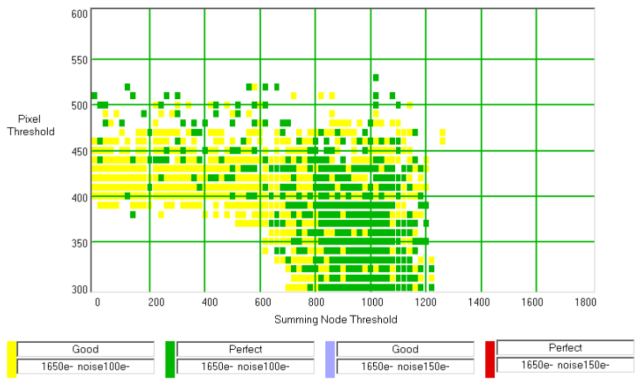

Fig. 16 PP algorithm efficiency scan for both pixel and summing node thresholds, signal size 1650 e-. Green and Yellow are perfect and good for 100 e- noise. Red and Blue would show perfect and good for 150 e- noise but no points exist. All are for better than 1 part in 10,000 errors. Notice that at 150 e- noise there is no valid operating point, and that at 100 e- noise the valid operating region is small.

If we then allow 1 part in 1,000 errors, the results are shown in Fig 17. Clearly there is an operating region at this error level; however, it is much narrower than at the higher signal level. It should be noted that at 1650 electrons signal and 100 electrons noise, there still is a small region that is perfect, but at 150 electrons noise there is no point where the device is perfect. Clearly, the noise is an issue which must be dealt with carefully.

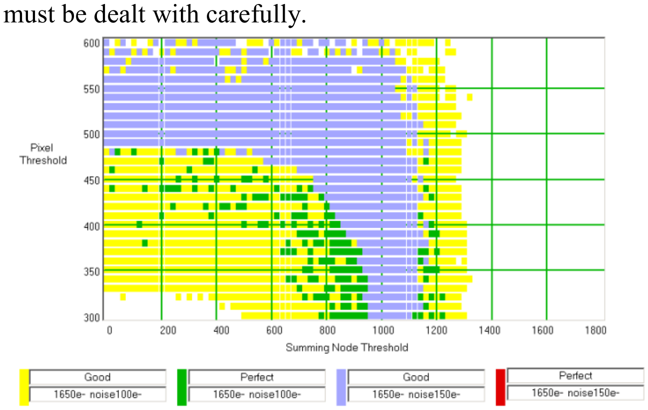

Fig. 17 PP algorithm efficiency scan for both pixel and summing node thresholds, signal size 1650 e-. Green and Yellow are perfect and good for 100 e- noise. Red and Blue are perfect and good for 150 e- noise. All are for better than 1 part in 1,000 errors. Notice that at 150 e- noise there is no perfect operating point. However, there is a large operating region for good at the 1 in 1,000 error level for both 100 e- and 150 e-.

This Pixel Pattern method shows promise as long as the noise can be controlled and the signal is large enough, however, it may be very difficult to implement within a limited pixel area.

IV. Summary of Results

Detailed results for C4P1 are shown in Fig. 2 and 3. The reader can see that as the thresholds increase there is no point where there are no extras and no missing hits. By not having the diagonal comparators, many single hits are seen as multiple hits, thus creating unacceptable false hits rates. While this algorithm is relatively easy to build in silicon, it just does not work well in the presence of noise and gain variations. Any method which requires more than one summing node to be above threshold is shown to have unacceptable missing hit rates. This is shown in Fig. 18, requiring 3 or 4 nodes is worse still.

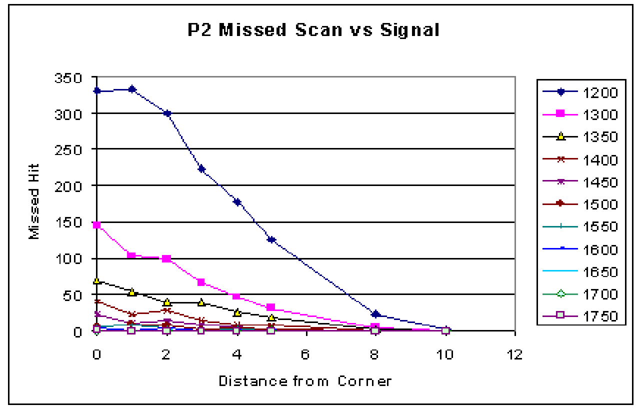

Fig. 18 P2 Scan (Requiring 2 summing nodes to fire) near the pixel corner as a function of signal size and distance from corner. Even for large signal sizes, the P2 algorithm has missing hit problems.

Even for large signals, hits are missing when they occur near a pixel corner. Although the P2 method does a good job of removing false hits, it is unacceptable due to the loss of real hits.

The C8P1 method was studied extensively and became the default design. We fully mapped the parameter space for signal size, noise, summing node threshold, and gain variation. This method works well for signal sizes above about 1300 electrons and for noise levels below about 120 electrons.

The pixel pattern recognition method was also extensively studied as it appeared to also be an efficient method. As the Fig. 14 shows, this method has a viable region of operation as a function of summing node threshold and pixel threshold versus noise and signal size. This method is a secondary design and is being implemented at lower priority. It has to be shown that it fits in the area of a single pixel.

V. More Realistic Noise Simulations

All of the work shown above used a purely Gaussian wide bandwidth white Noise signal for each pixel. However, a more realistic measure of the performance of the threshold comparators and algorithms would be to use noise that has passed through the amplifying circuitry and the front-end shaping. We therefore generated these noise signals based on white Gaussian high bandwidth noise at the first transistor and then propagated through the front end circuit. We repeated the simulations for both the C8P1 and the PP algorithms. Using a more realistic filtered noise signal, changes things in several ways. Examples of the white Gaussian Unfiltered Noise can be seen in Fig. 19 and then passing this noise through the filter produces a Filtered Noise signal shown in Fig. 20.

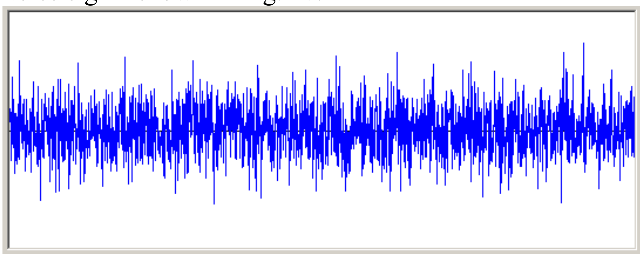

Fig. 19 2000 5 nanosecond time samples unfiltered or purely Gaussian white noise signal. RMS=100 electrons.

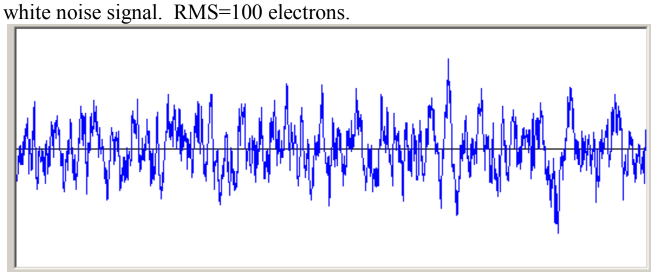

Fig. 20 2000 5 nanosecond time samples filtered noise. Input is Gaussian but then filtered by front-en bandwidth and shaping times. RMS=100 electrons.

The RMS of both types of noise are normalized to 100 electrons, however, the filtered noise does not have the high frequency of the unfiltered noise. The filtered noise will more closely match the real signals that will be driving the comparators in each pixel.

We created 64 of these filtered noise waveforms each having 23,000 time slices. Each pixel was randomly assigned a waveform and each waveform was randomly assigned a starting time between 1 and 10,000. The waveforms were normalized to produce the RMS noise level we were simulating. Then the simulations were run for 10,000 time slices and the number of extra hits or missed hits was determined. In this way each run was different and we could get a good measure of the performance of either algorithm for a given set of parameters.

A. C8P1 with new noise data

If we first look at the C8P1 algorithm comparing the Gaussian (Unfiltered) noise, Fig. 21, to the filtered noise, Fig. 22, we see that the acceptable noise level for having no extra hits goes up, thus improving the performance.

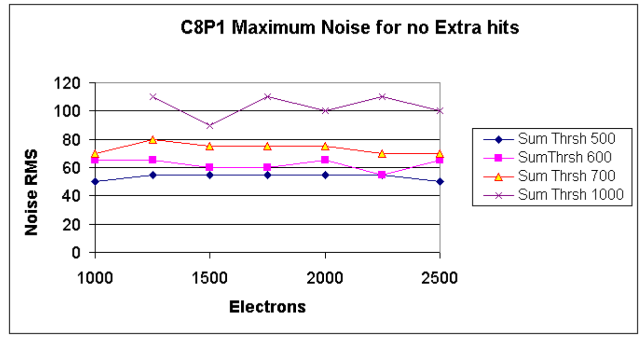

Fig. 21 Unfiltered Noise Performance for the C8P1 algorithm requiring no extra hits as a function of summing node threshold and signal size.

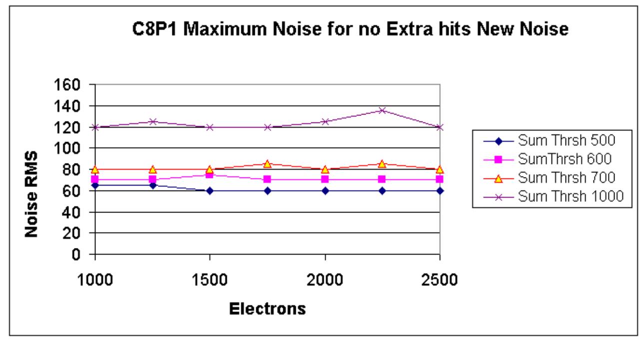

Fig. 22 Filtered Noise performance for the C8P1 algorithm requiring no extra hits. Level of allowable noise is increased giving more range for operating points.

Then looking at the noise level for missing no real hits for both unfiltered noise, Fig. 23 and filtered noise, Fig. 24, we see that the allowable noise for not missing real hits increases. Clearly using the more realistic noise for C8P1 gives a more realistic measure of the performance of the algorithm. While the change is not large, it does increase the margin for acceptable noise level in the front-end circuits.

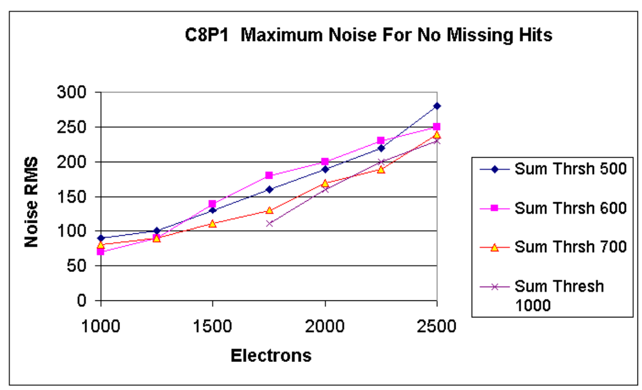

Fig. 23 Unfiltered noise performance for C8P1 algorithm requiring that there be no missing hits as a function of threshold and signal size.

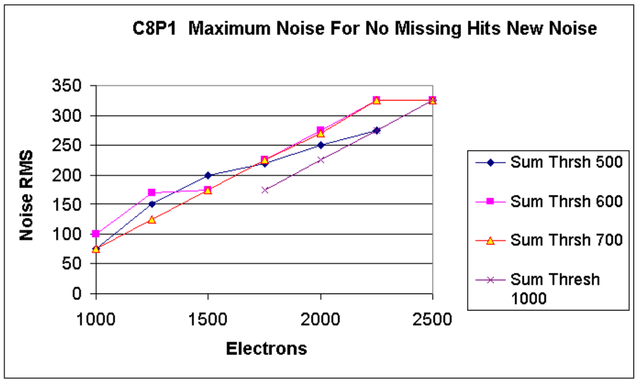

Fig. 24 Filtered noise performance requiring no missing hits. Again for a given threshold and signal size, the allowable noise to have no missing hits is increased. This improves the range for operation of the device. Note that the maximum noise level simulated was 325 electrons.

B. PP with new noise data

If we now look at the Pattern Recognition method (PP) using the new filtered noise waveforms, we see that using the more realistic noise gives a better representation of the true performance of the algorithm. The unfiltered noise matrix is shown in Fig. 25 and the filtered noise matrix is shown in Fig. 26, both are for 1 part in 10,000 extra or missing hits with a signal of 2200 electrons.

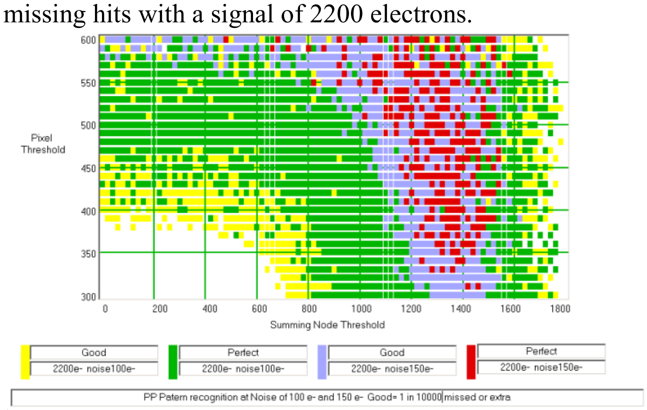

Fig. 25 Unfiltered noise PP algorithm performance matrix for 2200 e- signal- and 1 part in 10,000 missing or extra hits. Green is perfect and yellow is good for 100 electrons noise, red is perfect and blue is good for 150 electrons noise.

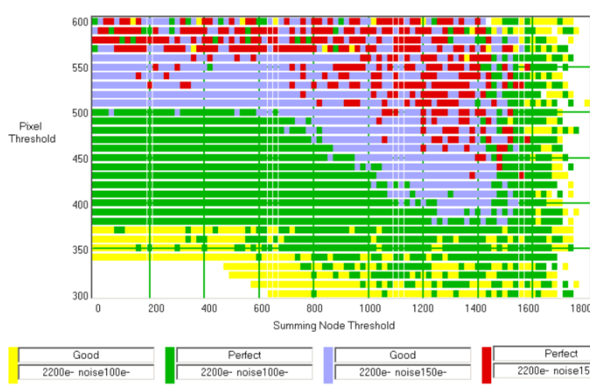

Fig. 26 Filtered noise PP algorithm performance matrix for 2200 e- signal- and 1 part in 10,000 missing or extra hits. Notice that the width of the green and blue horizontal bands has increased. This gives a broader region for operation and relaxes the noise requirement somewhat.

The operating point for the pattern recognition method is solid for pixel thresholds from 350 to 575 electrons and summing node thresholds from almost 0 to 1400 electrons. The summing node threshold seems to be redundant for this method. Note: comparing Fig. 25 and 26 it does appear that

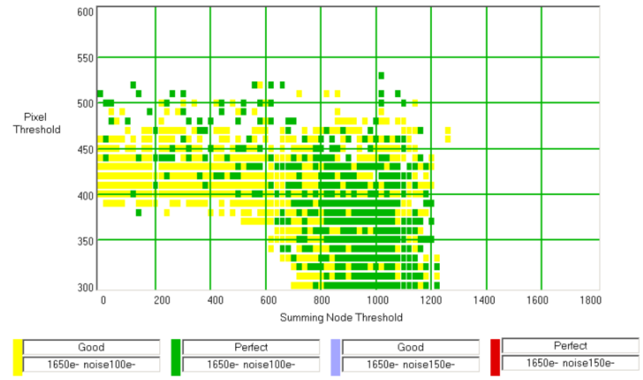

Fig. 27 PP algorithm. Performance with unfiltered noise at 1650 e-- and 1 part in 10,000 missing or extra hits. No points function at all at 150 e- noise.

using the more realistic noise, the efficiency at low pixel thresholds is reduced.

We also looked at the pattern recognition method with a signal of 1650 electrons. These are shown in Fig. 27 and

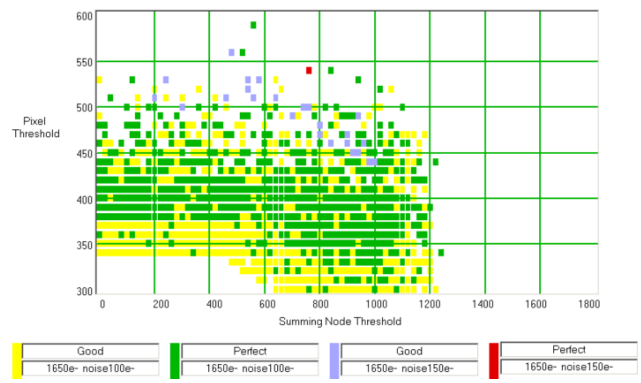

Fig. 28 Filtered noise PP algorithm performance at 1650 e- and 1 part in 10,000 missing or extra hits. Again the horizontal band for both good and perfect are wider and there are a few operating points where things work with 150 electrons noise.

We see that the filtered noise increases the area where things work and even some signals at 150 electrons noise are seen. This small signal size and noise region is difficult for the algorithm to resolve, but does show promise.

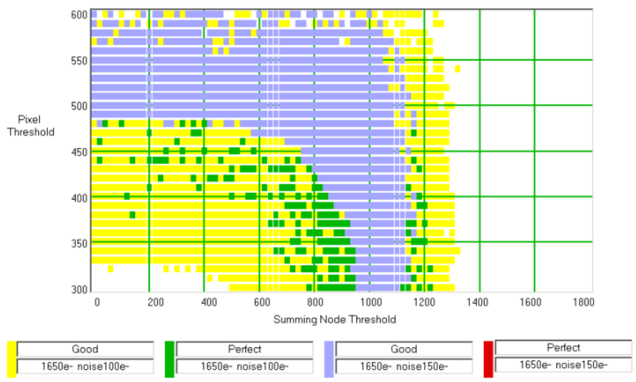

Fig. 29 Unfiltered noise PP algorithm performance at 1650 e- signal and 1 part in 1000 missing or extra hits.

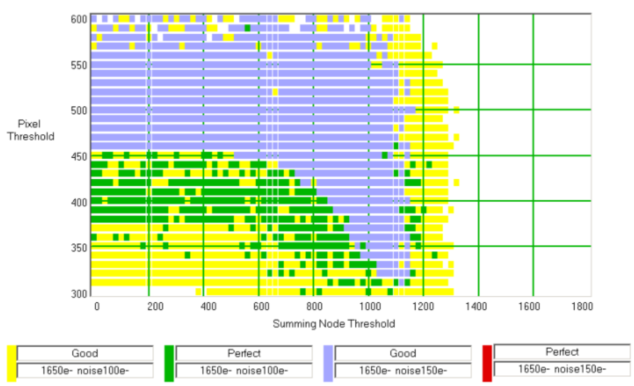

Fig. 30 Filtered noise PP algorithm performance at 1650 e- signal and 1 part in 1000 missing or extra hits. Now the region for good operation at 100 electrons noise is very wide and the operating region for 150 electrons noise is also wide enough to assure good functionality.

We also looked at the performance requiring only 1 in 1000 extra or missing hits. This is shown in Fig. 29 and Fig. 30. At the level of 1 part in 1000 errors we now have a larger operating region, see Fig. 30, for 1650 electrons and even a noise level of 150 electrons than in Fig 29 for the unfiltered case. Again, the more realistic filtered noise shows improvement in the performance of the algorithm over the raw unfiltered or purely white Gaussian noise.

VI. CONCLUSIONS

The C8P1 algorithm, which is the current default design, works well for photons which produce more than about 1300 electrons, provided that the noise level in each pixel is at or below about 110 electrons. C8P1 can tolerate noise levels of up to about 150 electrons for signals of 2000 electrons or above.

The PP or Pixel Pattern algorithm has difficulties with signals below about 1700 electrons and noise levels above about 125 electrons, but for low enough noise and larger signals does have a fairly broad operating range. In principle, PP also has the ability to give better position information than the C8P1. This is due to the extra information gained from adjacent hit pixels. This algorithm is still being studied to see if it can be made to fit in the current pixel area. It is not yet clear if this is possible, and if there is a way to extract the improved position information off chip.

Clearly the noise level in each pixel is critical. Every effort must be made to keep this noise level to a minimum. Simulations indicate that a noise level of about 75 electrons is achievable. As long as the noise level can be kept at or below about 110 electrons, the C8P1 and the Pixel Pattern recognition algorithm should both perform well.

The other important parameter is the signal size. Early hopes that a signal size of 1000 electrons could be detected efficiently were disproven by simulation results. It appears that at a minimum 1200 to 1500 electrons is needed. The targeted application providing around 2000 electrons hence appears easily achievable by either method.

REFERENCES

[1] Uher, J.; Harvey, G.; Jakubek, J.; "X-ray fluorescence imaging with the Medipix2 single-photon counting detector" Nuclear Science Symposium Conference Record (NSS/MIC), 2010 IEEE Page(s): 1067 – 1073

[2] O. G. Shpyrko, , E. D. Isaacs, J. M. Logan, Yejun Feng, et al; "Direct measurement of antiferromagnetic domain fluctuations", Nature 447, Pages 68 – 71, May 2007.

[3] C. Ponchut, "Correction of the charge sharing in photon-counting pixel detector data"., . Nuclear Instruments and Methods in Physics Research A, 591 Pages 311 - 313