

Piotr MAJ, Aleksandra DROZD

AGH AKADEMIA GÓRNICZO-HUTNICZA, KATEDRA METROLOGII I ELEKTRONIKI,  
Al. Mickiewicza 30, 30-059 Kraków

## Symulacje komputerowe efektu podziału ładunku w detektorach półprzewodnikowych z wykorzystaniem układu FPGA

Dr inż. Piotr MAJ

W 2005 roku ukończył studia magisterskie na Wydziale Elektrotechniki, Automatyki, Informatyki i Elektroniki Akademii Górniczo-Hutniczej im. Stanisława Staszica w Krakowie. Pracę doktorską obronił w 2008 roku i od tego czasu jest adiunktem w Katedrze Metrologii i Elektroniki. Jego zainteresowania naukowe to specjalizowane układy scalone do obrazowania cyfrowego oraz systemy kontrolno-pomiarowe.



e-mail: piotr.maj@agh.edu.pl

Mgr inż. Aleksandra DROZD

Absolwentka Matematyki Stosowanej oraz Międzywydziałowej Szkoły Inżynierii Biomedycznej na specjalności Elektronika i Informatyka Medyczna, na Akademii Górniczo-Hutniczej w Krakowie. Jej zainteresowania naukowe obejmują analizę ruchu człowieka, związane z tym systemy kontrolno-pomiarowe, a także modelowanie matematyczne i symulacje komputerowe.



e-mail: drozd@agh.edu.pl

### Streszczenie

Artykuł prezentuje wykorzystanie metody Monte Carlo do analizy efektu podziału ładunku w detektorach półprzewodnikowych. Są one niezbędne do podnoszenia jakości obrazowania medycznego, gdyż ze względu na pojawiające się na krawędziach pikseli rozmycie obniża dokładność pomiaru zarówno pozycji jak i energii. Wykonanie symulacji jest niezwykle istotne w pierwszym etapie projektowania dedykowanych układów scalonych, wymaga jednak dużych nakładów obliczeniowych, gdyż symulacje należy wielokrotnie powtarzać dla różnych wartości parametrów wejściowych. Zaproponowano zatem implementację przedstawionych algorytmów w układzie FPGA (Field Programmable Gate Array), pozwalającym na zrównoleglenie dużej ilości obliczeń i przez to na znaczące ich przyspieszenie.

**Słowa kluczowe:** efekt podziału ładunku, FPGA, metoda Monte Carlo.

### Computer simulations of charge sharing effect in semiconductor detectors using FPGA

#### Abstract

In recent years, there have been a lot of improvements in application of specific integrated circuits. New ideas have been also implemented in 2D X-Ray detectors which can be used in medical imaging. To understand the structure and the functionality of biological systems, there are required imaging systems of high spatial resolution. The aim is to improve the image quality but also to reduce the time of tissue exposure to the radiation. One of the major problems that needs to be solved while designing a detector with pixel dimensions less than  $200\ \mu\text{m} \times 200\ \mu\text{m}$  is compensation of charge sharing effect, presented schematically in Fig. 1. When a photon hits a detector close to the border between pixels, the resultant charge cloud is divided between up to four neighbor readout channels. Introduction of the compensation algorithm C8P1 [1] leads to a proper measurement of the spatial hit position as well as photon energy despite the fact that charge sharing occurred. Since the algorithm should be verified before hardware implementation, simulation using the Monte Carlo method was implemented. The simulation step is a crucial part in a design process, although it often takes much time to perform the simulations, as there are multiple parameters in the model that vary in specific ranges (Tab. 1). Taking the execution time into account, a new solution using the FPGA (Field Programmable Gate Array) was proposed (Fig. 4). Measurements of the operation execution time (Tab. 2) show that implementing simulations in an FPGA chip can accelerate calculations about 1000 times in comparison with the standard PC implementation (with Intel Core i7 processor), which is a significant improvement considering the integrated circuit design process.

**Keywords:** charge sharing, FPGA, Monte Carlo method.

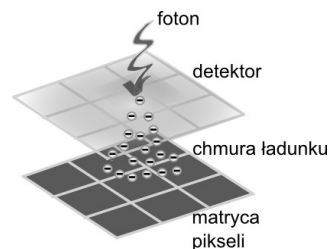
### 1. Wstęp

Stojące przed naukowcami problemy w rozwiązaniu zagadnień podlegających naturalnym, fizycznym ograniczeniom często spotykają się z koniecznością weryfikacji wielu różnych pomysłów. Przykładem może być tutaj proces projektowania układów scalonych do odczytu sygnałów z monolitycznego, krzemowego

pikselowego detektora promieniowania X, który ma szerokie zastosowanie w obrazowaniu medycznym. Aby zrozumieć funkcjonowanie i poznać strukturę systemów biologicznych, wymagane są technologie pozwalające na uzyskiwanie coraz lepszej rozdzielczości przestrzennej i energetycznej. Celem jest poprawa jakości obrazowania medycznego, a także minimalizacja czasu ekspozycji tkanki na promieniowanie.

W detektorze pikselowym uderzające w jego powierzchnię fotony powodują generację chmury ładunku, której bezwzględna wartość wyrażona w elektronach jest proporcjonalna do energii wpadającego fotonu. Wygenerowana chmura ładunku jako mały impuls prądowy stanowi sygnał wejściowy dla pojedynczego kanału odczytowego w matrycy pikseli. Wielkość pojedynczej komórki piksela zależy od jego funkcjonalności oraz użytej technologii i waha się od kilkudziesięciu do kilkuset  $\mu\text{m}$  [1-3].

Wpadające impulsy mogą być przetwarzane przez układ elektroniczny na dwa sposoby – mogą one być całkowane w zadanym oknie czasowym bez względu na ich amplitudę (CCD) lub zliczane mogą być jedynie impulsy o zadanej energii. Drugie podejście powoduje konieczność szybkiego przetwarzania kolejnych impulsów oraz porównywania ich amplitudy z pewnym zadanym progiem zwanym napięciem dyskryminacji. W układach tych błędami pomiaru amplitudy obarczone są te impulsy, które wpadają do detektora pomiędzy dwoma kanałami odczytowymi (rys. 1).



Rys. 1. Schemat prezentujący efekt podziału ładunku. Uderzający w detektor foton powoduje wygenerowanie chmury ładunku, która podzieli się pomiędzy sąsiednie piksele

Fig. 1. General idea of charge sharing effect. When incoming photon hits a detector, it generates charge cloud which might be divided between neighbor pixels

Wygenerowana w ten sposób chmura ładunku dzieli się pomiędzy dwa lub nawet cztery sąsiednie piksele, skutkując błędem w określeniu amplitudy przetwarzanego w torze elektroniki impulsu. Ilość cząstek podlegających opisywanemu efektowi podziału ładunku jest odwrotnie proporcjonalna do rozmiarów piksela, a zatem konieczne jest uwzględnienie tego efektu do precyzyjnego określenia amplitudy impulsu i miejsca zderzenia, zwłaszcza dla pikseli o rozmiarach rzędu kilkudziesięciu  $\mu\text{m}$ .

Stojące przed projektantami układu elektroniki odczytu wyzwania polegają na takim odczycie sygnału z detektora, aby wyeliminowany był negatywny efekt podziału ładunku. Poprawny pomiar całkowitej wartości wpadającego do detektora ładunku można osiągnąć przez implementację stosownego algorytmu przetwarzania sygnału. Niestety koszty produkcji układów, jak również czas poświęcony na jego projektowanie i późniejsze testy nie pozwalają na dowolną ilość podejść różniących się drobnymi szczegółami. Rozwiązaniem może być tutaj wykorzystanie numerycznego modelowania zjawisk fizycznych, symulacji komputerowych i włączanie do nich różnych algorytmów podlegających weryfikacji. Tego typu metody pozwalają na eliminację potencjalnych błędów oraz optymalizację algorytmu jeszcze przed etapem projektowania.

Aby zidentyfikować parametry mające wpływ na efekt podziału ładunku a także, aby przetestować rozwiązania układowe i algorytmy pozwalające na minimalizację tego efektu, przeprowadzono symulacje komputerowe metodą Monte Carlo. Ponieważ metoda ta wymaga wielokrotnie wykonanych obliczeń, a także ze względu na fakt, iż w symulacjach badano wpływ wielu parametrów wejściowych na efekt podziału ładunku, zaproponowano implementację przedstawionych algorytmów w układzie FPGA (Field Programmable Gate Array), pozwalającym na znaczące przyspieszenie obliczeń. W artykule zaprezentowano sposób implementacji w FPGA przykładowego algorytmu minimalizującego efekt podziału ładunku C8P1 [3].

## 2. Materiały i metody

### Modelowanie podziału ładunku

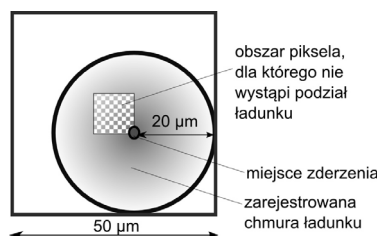
Opisany proces podziału ładunku można modelować znając zjawiska fizyczne mające wpływ na ładunek wygenerowany w detektorze. Chmura ładunku powstała w wyniku interakcji cząstki z detektorem dyfunduje w kierunku elektrody zwiększając swój rozmiar na skutek zjawisk elektrostatycznych oraz dyfuzji [4]. Rozwiązując równanie dyfuzji, przy założeniu zerowych warunków początkowych, można wyprowadzić model przestrzenny chmury ładunku, którą poprawnie przybliży funkcja Gaussa [5]. Równanie (1) opisuje trójwymiarowy model chmury ładunku.

$$f(x, y) = A \exp\left(-\left(\frac{(x-x_0)^2}{2\sigma^2} + \frac{(y-y_0)^2}{2\sigma^2}\right)\right), \quad (1)$$

gdzie  $x, y$  są pozycjami fotonu na płaszczyźnie matrycy,  $A$  jest współczynnikiem amplitudowym zależnym od energii fotonu,  $(x_0, y_0)$  – średnią pozycją (zwykle przyjmuje się  $x_0 = y_0 = 0$ ),  $\sigma$  – odchyleniem standardowym (ze względu na założenie o symetrii chmury ładunku, przyjmuje się  $\sigma_x = \sigma_y = \sigma$ ). Dodatkowo, założono brak korelacji pomiędzy  $x$  i  $y$ .

Badania eksperymentalne przeprowadzane dla detektorów o grubości warstwy krzemowej równej 300  $\mu\text{m}$  pokazały, że szerokość obszaru piksela, dla którego nastąpi podział ładunku wynosi około 20  $\mu\text{m}$  [3, 7]. W modelu przyjęto, że wystąpiło zjawisko podziału ładunku, jeśli mniej niż 99,7% ładunku zostało zarejestrowane przez jeden kanał odczytowy. Oznacza to, że w przypadku pikseli o rozmiarach 50  $\mu\text{m} \times 50 \mu\text{m}$  ponad 90% wpadających do detektora fotonów ulegnie efektowi podziału ładunku, co widoczne jest na rys. 2.

Symulując działanie torów odczytowych dla detektorów pikselowych należy wziąć także pod uwagę szumy poszczególnych elementów układów elektronicznych takie jak: rozrzut wzmocnienia, szum addytywny dla każdego piksela, szum dyskryminatorów i szum komparatorów. Wszystkie wymienione szumy są w symulacjach generowane jako losowe wartości z rozkładu Gaussa, o parametrach odpowiadających zmierzonym parametrom w rzeczywistych układach [1-3]. Przyjęte zakresy wartości dla celów symulacji zgromadzono w tab. 1.



Rys. 2. Piksel o wymiarach 50  $\mu\text{m} \times 50 \mu\text{m}$ . Czarno-biały kwadrat reprezentuje ten fragment piksela, dla którego nie jest obserwowany efekt podziału ładunku  
Fig. 2. 50  $\mu\text{m} \times 50 \mu\text{m}$  pixel. Black and white central square represents hit area not affected by charge sharing effect

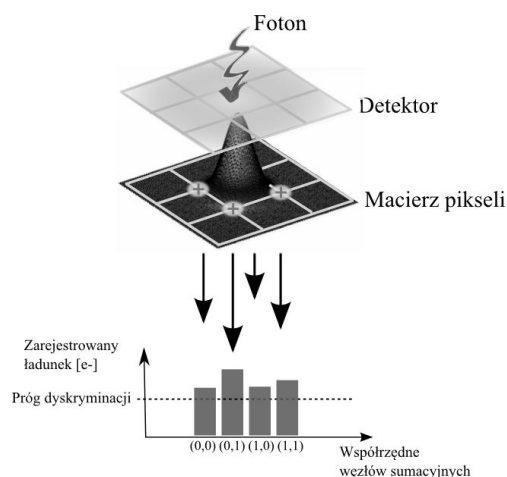
Tab. 1. Parametry symulacji – zakresy przyjętych wartości szumów  
Tab. 1. Simulation parameters – assumed noise ranges

Parameters	Parameters ranges
Szumy pikseli [e- rms]	0 - 200
Szumy dyskryminatorów [e- rms]	0 - 40
Szumy komparatorów [e- rms]	0 - 40
Rozrzut wzmocnienia [%]	0 - 10
Energia cząstki [e-]	0 - 6600

### Algorytmy wyznaczania miejsca zderzenia i energii cząstki

W standardowym podejściu, bez kompensacji efektu podziału ładunku, każdy piksel zlicza tylko zarejestrowany ładunek, a jako miejsce zderzenia przyjmuje się piksel o największej zarejestrowanej wartości. Takie podejście powoduje jednak błędy zarówno w wyznaczeniu pozycji zderzenia, jak i całkowitej energii cząstki.

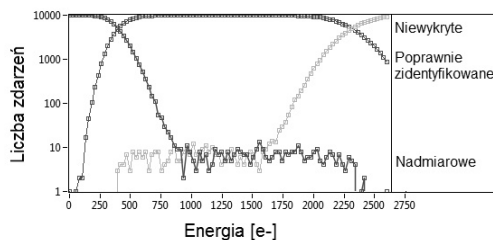
Alternatywnym podejściem, zaproponowanym w pracy [6] jest wykorzystanie algorytmu C8P1 do kompensacji efektu podziału ładunku [3]. Metoda ta polega na sumowaniu ładunku z każdego czterech sąsiednich pikseli oraz porównaniu amplitudy impulsu danego piksela z wszystkimi jego sąsiadami. Jeżeli sumowany ładunek przekroczy zadany próg dyskryminacji oraz będący w otoczeniu węzła sumującego piksel będzie przetwarzał impuls o amplitudzie większej niż każdy z jego sąsiadów, to jest on pikselem, w który uderzyła cząstka o danej, zsumowanej amplitudzie. Schemat działania opisanej metody zaprezentowano na rys. 3.



Rys. 3. Obrazowo przedstawiony schemat działania algorytmu C8P1. Wygenerowana chmura ładunku dzieli się pomiędzy sąsiednie piksele. Dla każdego węzła sumacyjnego sumowane są wartości 4 sąsiadów, a następnie dokonane jest porównanie z progiem dyskryminacji  
Fig. 3. General idea of C8P1 algorithm. Charge cloud is divided between neighbor pixels. For each summing node values of 4 neighbors are summed up and the sum values are compared to the threshold value

## Symulacje Monte Carlo

Aby zweryfikować poprawność algorytmu C8P1 przed etapem projektowania układu scalonego, został on zamodelowany w środowisku LabVIEW. Przeprowadzono symulację metodą Monte Carlo polegającą na wygenerowaniu 10 000 losowych pozycji zderzenia cząstki z detektorem o rozmiarze 6x6 pikseli. Zamodelowano generację chmury ładunku o rozkładzie gaussowskim o odchyleniu standardowym równym 10  $\mu\text{m}$  i średnich równych 0. Wygenerowana chmura została naniesiona na macierz pikseli w miejscu odpowiadającym zderzeniu, a następnie zsumowano ładunki dla każdego piksela. W ten sposób uzyskano model sygnałów odczytywanych z detektora. Następnie zamodelowano algorytm C8P1 pozwalający na wyznaczenie zarówno całkowitej amplitudy podzielonego impulsu jak i miejsca zderzenia, uwzględniając dodatkowe parametry takie jak: szumy oraz rozrzut wzmacnienia. Badano, w ilu przypadkach miejsca zderzenia z detektorem zostały poprawnie wyznaczone w zależności od poziomu progu dyskryminacji. Rys. 4 przedstawia przykładowe wyniki uzyskane w symulacjach. Zakres energii od około 700 do 2000 elektronów odpowiada poprawnej detekcji z nieznacznymi błędami wynikającymi z uwzględnienia szumów.



Rys. 4. Przykładowe wyniki symulacji dla 10 000 zdarzeń, energii cząstki = 2200 e-, rozrzutu wzmacnienia = 5%,  $\sigma$  szumu pikseli = 100 e-,  $\sigma$  szumu komparatorów = 10 e-,  $\sigma$  szumu dyskryminatorów = 20 e-  
 Fig. 4. Simulation results for 10 000 hits, photon energy = 2200 e-, gain spread = 5%,  $\sigma$  pixel noise = 100 e-,  $\sigma$  comparator noise = 10 e-,  $\sigma$  discriminator noise = 20 e-

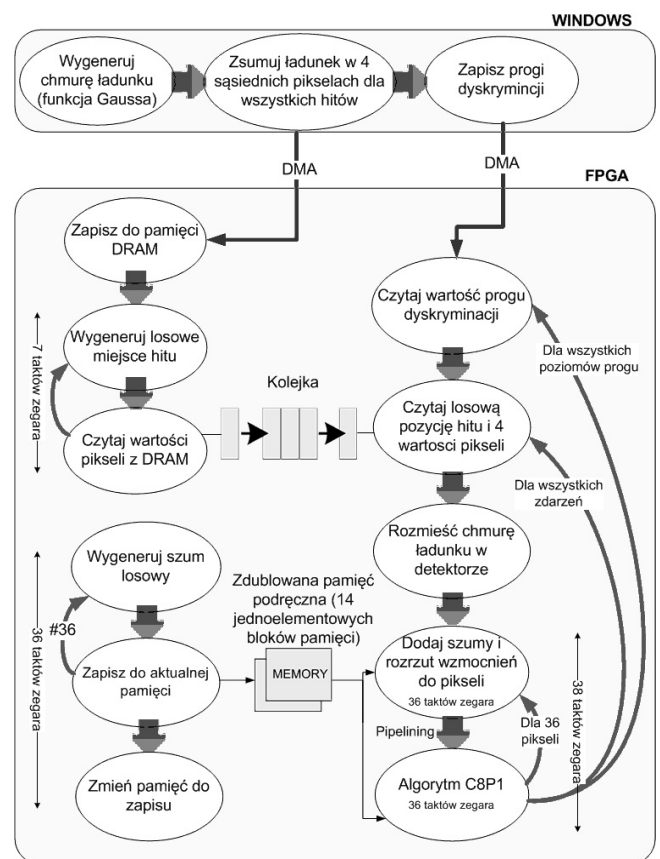
Wykonywane symulacje pokazują, że C8P1 pozwala na poprawny pomiar amplitudy impulsu mimo jego fizycznego podziału oraz na poprawne wskazanie piksela, w okolicy którego uderzyła cząstka. Niestety, modelowanie zjawisk w systemach wysokiego poziomu, tj. na standardowym komputerze typu desktop PC, mimo wygody zapisu w dowolnym języku programowania i zapewnienia odpowiedniej warstwy abstrakcji oraz, co ważne, dużej rozdzielczości numerycznej, rzadko są najszybszymi z możliwych metod. Szczególnie, jeżeli mamy do czynienia z metodami Monte Carlo, które są powtarzane dla danego procesu dziesiątki lub setki tysięcy razy.

W przypadku symulacji wykonywanych dla potrzeb projektowania dedykowanych układów odczytowych dla detektorów pikselowych mamy do czynienia z modelowaniem wielowymiarowym, gdyż zadaniem jest nie tylko weryfikacja algorytmu, ale też znalezienie optymalnych parametrów układu, przy których detekcja miejsca zderzenia będzie poprawna. Istnieje zatem potrzeba przeprowadzania symulacji dla parametrów takich jak: próg dyskryminacji, poziom szumów, energia cząstki zmiennych w zadanych zakresach.

## Wykorzystanie układu FPGA do symulacji

Zaimplementowanie algorytmów w systemie Windows pozwoliło na uzyskanie poprawnych wyników, natomiast ze względu na dużą ilość operacji, czas wykonania był znaczący. Zauważono, że pewne operacje mogą być wykonywane równoległe, dlatego alternatywą może być zaimplementowanie symulacji sprzętowo w układach FPGA, czyli układach programowalnych bramek

logicznych. Łączą one zalety dedykowanych układów scalonych z systemami opartymi na procesorach, pozwalając na zwiększenie szybkości obliczeń przy zachowaniu możliwości ponownego programowania zadań. Każde zdefiniowane zadanie jest przypisane do odpowiedniej sekcji układu FPGA, co umożliwia niezależne ich wykonywanie bez konieczności współdzielenia zasobów. Środowisko LabVIEW firmy National Instruments pozwala na projektowanie aplikacji działających w układach FPGA z wykorzystaniem wysokopoziomowych narzędzi, co pozwoliło na szybką implementację opisanych symulacji i weryfikację wyników. Ogólny schemat algorytmu z uwzględnieniem rozłącznych wątków w systemie Windows oraz w układzie FPGA został pokazany na rys. 5.



Rys. 5. Schemat blokowy symulacji zaimplementowanej w układzie FPGA  
 Fig. 5. Idea of simulation implemented in FPGA

W algorytmie C8P1 wyróżnić można szereg niezależnych operacji dla pojedynczego zdarzenia, takich jak:

- wylosowanie miejsca uderzenia cząstki w detektor,
- wylosowanie szumu pojedynczego piksela,
- wylosowanie rozrzutu wzmacnienia,
- wylosowanie szumu komparatorów i dyskryminatorów.

Wszystkie te operacje mogą być wykonywane równoległe, zostały więc zaimplementowane jako niezależne funkcje wykonywane ze zdefiniowaną częstotliwością (40 MHz). Po wylosowaniu omówionych wartości, zapisano je do pamięci podręcznej, w której przechowywane są do czasu wykonywania obliczeń dla jednego wygenerowanego zdarzenia. Zaproponowano także dodatkowe usprawnienie polegające na obsłudze dwóch kopii tej pamięci, w której przechowywane są losowane wartości. W ten sposób umożliwiono równoległy odczyt z pamięci i generację kolejnych wartości, które zostaną wykorzystane w następnym wykonaniu algorytmu C8P1.

Wykorzystano także zewnętrzną dla układu FPGA pamięć DRAM, w której można zapisać duże zbiory danych. Ponieważ sumowanie ładunku w sąsiednich pikselach każdorazowo po

wylosowaniu nowego miejsca zderzenia cząstki z detektorem wiązało się z dużym kosztem obliczeniowym, wygenerowano wszystkie możliwe podziały chmury ładunku pomiędzy 4 piksele w zależności od miejsca uderzenia (dla rozdzielczości 0,5  $\mu\text{m}$  i rozmiaru piksela 75  $\mu\text{m}$ ) i zapisano te wartości do pamięci DRAM (dane zajęły 360 kB). Wówczas wyznaczenie sumarycznego ładunku w zadanych pikselach konieczne z punktu widzenia algorytmu, sprowadza się do odczytania 4 wartości z odpowiedniej komórki pamięci.

### 3. Wyniki

Na algorytm C8P1 składa się szereg operacji w tym dodawanie wartości szumów, sumowanie 4 sąsiednich pikseli, porównania z progiem dyskryminacji jak i porównania wartości pikseli między sobą. Dzięki zaimplementowaniu algorytmu w układzie w FPGA wszystkie wymienione obliczenia dla jednego piksela są wykonywane w jednym taktie zegara. Algorytm jest powtarzany w symulacjach 36-krotnie dla wszystkich pikseli matrycy. Zrównoleglenie operacji generacji losowych wartości szumów i miejsca zderzenia (z wykorzystaniem techniki przetwarzania potokowego – ang. 'pipelining') powoduje, że wszystkie obliczenia sumarycznie wykonują się w 38 taktach zegara.

W tab. 2 zaprezentowano porównanie czasów wykonania pojedynczych wątków symulacji, jak i całego kroku symulacji dla jednego zdarzenia w systemie Windows i w układzie FPGA.

Tab. 2. Porównanie zmierzonych średnich czasów wykonania poszczególnych operacji w symulacjach dla systemu Windows i układu FPGA

Tab. 2. Comparison of different operation average execution time for Windows system and FPGA chip

Hardware	Czas wykonania operacji		
	C8P1 dla pojedynczego piksela	Generacja szumów dla pojedynczego piksela	Wykonanie kroku symulacji dla całej matrycy
Intel Core i7 Q 820 (1.73 GHz)	200 $\mu\text{s}$	210 $\mu\text{s}$	7.2 ms
Virtex-5 SX50T FPGA	25 ns	25 ns	950 ns

Zaprezentowane wyniki pokazują, że dzięki implementacji metody Monte Carlo w układzie FPGA, symulacje mogą zostać znacząco przyspieszone. Ma to szczególne znaczenie, gdy badanych jest szereg różnych parametrów, gdyż przykładowe symulacje mogą obejmować obliczenia dla 10 000 zdarzeń i 100 różnych progów dyskryminacji. Obliczenia te wykonują się w układzie FPGA około 1000 razy szybciej, natomiast wyniki działania algorytmów dla systemu Windows i układu FPGA są porównywalne.

### 4. Wnioski

Zaprezentowane podejście wykorzystania symulacji komputerowych w początkowych etapach projektowania m.in. detektorów promieniowania X pozwala na weryfikację algorytmów, które mogą być następnie zaimplementowane bezpośrednio w układzie scalonym. Symulacje są niezbędnym narzędziem do oceny poprawności i efektywności przyjętych metod, jednak często w przypadku złożonych algorytmów i wielu parametrów wejściowych ich czas trwania jest znaczący. Dlatego zaproponowanie rozwiązanie polegające na wykorzystaniu do symulacji układu FPGA, może być przydatne zwłaszcza w przypadku stosowania metody Monte Carlo, która wymaga wielokrotnego powtarzania obliczeń dla kolejnych wygenerowanych danych wejściowych.

Wykonane symulacje posłużyły do weryfikacji algorytmu C8P1, który został zaimplementowany w pikselowym układzie scalonym wykonanym w technologii 45 nm.

*Projekt został sfinansowany ze środków Narodowego Centrum Nauki przyznanych na podstawie decyzji numer DEC-2011/01/B/ST/05155.*

### 5. Literatura

- [1] Szczygiel R., Grybos P., Maj P.: A Prototype Pixel Readout IC for High Count Rate X-Ray Imaging Systems in 90 nm CMOS Technology. Transactions on Nuclear Science, vol. 57, No.3, June 2010.
- [2] Szczygiel R., Grybos P., Maj P., Zolodz M.: PXD 18k - Fast Single Photon Counting Chip with Energy Window for Hybrid Pixel Detector, 2011 IEEE Nuclear Science Symposium Conference Record.
- [3] Maj P., Baumbaugh A., Deptuch G., Grybos P., Szczygiel R.: Algorithms for minimization of charge sharing effects in a hybrid pixel detector taking into account hardware limitations in deep submicron technology. IOP Publishing for SISSA MEDIALAB, December, 2012.
- [4] Castoldi A., Rehak P., Holl P.: Signal Charge Sharing in Multilinear Drift Detectors: Design and Experimental Characterization, IEEE Transactions on Nuclear Science, Vol. 44, No. 2, April 1997.
- [5] Komarov S., Yin Y., Wu H., Tai Y.: Simulation Study of Charge Collection in Highly Pixelated CdZnTe Detector for PET Imaging. 2009 IEEE Nuclear Science Symposium Conference Record.
- [6] Maj P., Baumbaugh A., Deptuch G., Grybos P., Szczygiel R.: Minimization of Charge Sharing Effect in Silicon Hybrid Pixel X-Ray Detectors Based on Pattern Recognition Algorithm, IEEE International Conference on Industrial Technology (ICIT), 19-21 March 2012, pp.551 – 556.
- [7] Bergamaschi A., Broennimann Ch., Dinapoli R., Eikenberry E., Gozzo F., Henrich B., Kobas M., Kraft B., Patterson P., Schmitt B.: Performance of a single photon counting microstrip detector for strip pitches down to 10 mm. Nuclear Instruments and Methods in Physics Research A 591 (2008) 163–166.

otrzymano / received: 18.12.2012

przyjęto do druku / accepted: 01.02.2013

artykuł recenzowany / revised paper

## INFORMACJE

# Wersja elektroniczna miesięcznika PAK

## Pomiary Automatyka Kontrola – Measurement Automation and Monitoring

Artykuły opublikowane w PAK po roku 1989 są dostępne w wersji elektronicznej m.in. w bazie artykułów PAK ([www.pak.info.pl](http://www.pak.info.pl)), w folderze „Archiwum numerów miesięcznika PAK”:

- pełne teksty artykułów starszych niż 3 lata oraz streszczenia artykułów najnowszych można pobrać bezpłatnie,
- pełne teksty artykułów najnowszych można otrzymać za opłatą (5 PLN +1,15 PLN VAT).