

Dr hab. Jacek Sekutowicz

SLAC, Stanford University
2575 Sand Hill Road
Menlo Park, California, 94025, USA
jksekuto@stanford.edu

DESY, Deutsches Elektronen-Synchrotron
Notkestraße 85
22607 Hamburg, Germany
jacek.sekutowicz@desy.de

Recenzja Rozprawy Doktorskiej dla Wydziału Elektrotechniki, Elektroniki, Informatyki i Automatyki Politechniki Łódzkiej

Tytuł Rozprawy:

Metodologia prototypowania sterowników cyfrowych budowanych w oparciu o architekturę MTCA do zastosowań w systemach synchronizacji optycznej laserów na swobodnych elektronach

Prototyping methodology for digital controller design based on the MTCA infrastructure for optical synchronization systems in free electron lasers

Autor Rozprawy:

Mgr inż. Paweł Prędko

1. Uwagi wstępne i tematyka Rozprawy w kontekście badań naukowych

Przedstawiona przez mgr. inż. Pawła Prędkiego Rozprawa Doktorska poświęcona jest projektowaniu i prototypowaniu sterowników cyfrowych stosowanych w układach elektronicznych synchronizujących pulsy koherentnego promieniowania laserów na swobodnych elektronach (*ang. Free Electron Laser, FEL*) z pulsami laserów optycznych, stosowanych zarówno w eksperymentach przeprowadzanych z wykorzystaniem FEL jak i laserów optycznych oddziałujących bezpośrednio z wiązką elektronową liniowego akceleratora (liniaka) przyspieszającego paczki elektronów, przed przejściem ich przez undulator, w celu podniesienia wydajności procesu wytwarzania fotonowych błysków w FEL.

Przykładem pierwszego z wymienionych zastosowania są układy cyfrowe do synchronizacji laserów optycznych używanych w badaniach przebiegu reakcji chemicznych w eksperymentach typu wzbudzenie i próbkowanie (*ang. pump and probe*), w których optyczny laser inicjuje reakcję chemiczną a pulsy promieniowania FEL, z opóźnieniem dobranym przez eksperymentatora, próbują obszar zachodzenia reakcji, umożliwiając obserwację przemieszczania się atomów oraz określenia w funkcji czasu kolejnych faz powstawania nowej struktury przestrzennej związku chemicznego.

Przykładem drugiego zastosowania układów elektronicznych z cyfrowymi sterownikami jest synchronizacja laserów optycznych oddziałujących z wiązką elektronową FEL w celu przyspieszenia procesu mikro-grupowania elektronów w paczkach (*ang. seeding*), co umożliwia zastosowanie krótszych i mniej kosztownych undulatorów w procesie samowzmacniającej emisji spontanicznej (*ang. SASE*) oraz poprawia mono-energetyczności wiązek fotonowych w FEL.

Kolejnym przykładem wykorzystania układów elektronicznych ze sterownikami cyfrowymi jest optymalizacja procesu przyspieszania w liniaku FEL poprzez pomiar czasu przelotu elektronów w wybranych punktach ich trajektorii, z wykorzystaniem tzw. monitorów przylotu paczek elektronowych (*ang. Beam Arrival Monitor, BAM*) i odpowiedni dobór faz kolejnych wzmacniaczy wysokiej częstotliwości (w. cz.), zasilających struktury przyspieszające w celu uzyskania maksymalnej energii końcowej elektronów.

Synchronizację pulsów laserów optycznych oddziałujących z wiązką elektronową i tych używanych w eksperymentach, uzyskuje się poprzez dopasowanie czasu wysłania pulsu lasera optycznego do czasu generacji paczek w źródle elektronów, czasu potrzebnego na ich akcelerację i kompresję, czasu przejścia przez undulatory, oraz czasu dotarcia błysku fotonowego FEL do miejsca eksperymentu. Główną trudnością techniczną w synchronizacji jest mała długość skompresowanych paczek elektronów poruszających się z prędkością bliską prędkości światła, a co za tym idzie niezwykle krótki czas trwania fotonowych pulsów FEL, który jest zwykle poniżej 100 fs.

We współczesnych akceleratorach badawczych, wykorzystywanych zarówno w fizyce wysokich energii jak i tych stosowanych w laserach na swobodnych elektronach, do synchronizacji oddalonych od siebie urządzeń stosowane są techniki optyczne. Podstawowym elementem synchronizacji optycznej jest główny oscylator laserowy (*ang. Master Laser Oscillator, MLO*), którego optyczne mody zsynchronizowane są fazowo (*ang. mode-locked*) w celu uzyskania krótkich czasowo pulsów optycznych. Dodatkowo, MLO jest zsynchronizowany z głównym oscylatorem wysokiej częstotliwości (*ang. Master Oscillator, MO*), który pełni rolę referencyjnego zegara dla całego akceleratora i urządzeń eksperymentalnych.

W przypadku dwóch laserów na swobodnych elektronach w hamburskim laboratorium DESY (*niem. Deutsches Elektronen-Synchrotron*): FLASH (*Free Electron Laser in Hamburg*) i E-XFEL (*ang. European X-ray Free Electron Laser*), z których pierwszy jest eksploatowany od ponad 10-ciu lat a drugi znajduje się w końcowej fazie budowy, i w których cyfrowe układy elektroniczne projektowane z użyciem metod i narzędzi zaproponowanych przez mgr. inż. Pawła Prędkiego zostaną zastosowane, MLO jest laserem erbowym Er:YAG, generującym ciąg pulsów światła o długości fali 1150 nm z częstotliwością 216.6 MHz i czasie trwania 200 fs. Pulsy MLO wysyłane są jednocześnie do kilku kanałów, z których każdy obsługuje jedno stabilizowane połączenie optyczne (*ang. link-stabilization-unit, LSU*). W każdym LSU pulsy są rozdzielane na puls referencyjny i puls wysyłany światłowodem o skompensowanej długości do urządzeń synchronizowanych. Korekta długości światłowodu, której dokonuje się przy pomocy piezo-elementów dla szybkich zmian i silnikami krokowymi dla wolnych zmian, jest konieczna ze względu na niestabilność temperatury i wilgotności w tunelu akceleratora oraz oscylacji mechanicznych światłowodu. Pulsy wysyłane światłowodem są na jego końcu częściowo odbijane przez półprzezroczyste lustro. Całkowity czas propagacji odbitej części pulsu świetlnego jest porównywany w optycznym korelatorze w LSU z czasem pulsów referencyjnych. Sygnał elektryczny odpowiadający różnicy czasów jest dostarczany do sterowników cyfrowych piezo-elementów i silników krokowych regulujących długość światłowodu.

Warto już w tym miejscu podkreślić, że synchronizacja w urządzeniach badawczych, wykorzystujących akceleratory liniowe do przyspieszania elektronów oraz lasery optyczne w eksperymentach lub w optymalizacji procesu generacji fotonów, odgrywa niezwykle istotną rolę. Udział mgr. inż. Pawła Prędkiego w budowaniu elementów synchronizacji, a zwłaszcza uproszczenie procesu ich projektowania i modelowania, jest istotnym wkładem w rozwój fotonowych metod badawczych.

2. Omówienie Rozprawy

Rozprawa doktorska mgr. inż. P. Prędkiego składa się z sześciu rozdziałów, jednego rozdziału zawierającego dodatki oraz spisu literatury zawierającego 61 pozycji. W sześciu

wymienionych artykułach mgr inż. Prędko jest współautorem. Praca napisana jest w języku angielskim. Ponieważ wiele oznaczeń i skrótów angielskich nie zostało jeszcze oficjalnie przetłumaczone na język polski i nie pojawia się w zaleceniach Komisji Nazewnictwa PTF, w dalszej części recenzji będą używane również oryginalne nazwy i skróty angielskie.

Rozdział I: Wstęp (Introduction)

W pierwszym podrozdziale Wstępu, Autor opisuje laser na swobodnych elektronach FLASH oraz podaje podstawowe informacje odnośnie koncepcji synchronizacji w tym laserze. W kolejnym podrozdziale opisana jest krótko architektura układów elektronicznych budowanych w standardzie MTCA (*Micro Telecommunications Computing Architecture*), która coraz częściej znajduje zastosowanie w elektronice dużych urządzeń badawczych, zwłaszcza od 2011 roku, w którym powstała jej nowa wersja MTCA.4 zawierająca w standardzie RTM (*Rear Transitions Module*) i mająca powiększone możliwości o synchronizację jak również o koordynację czasową (*ang. timing*). Następnie Autor wymienia podstawowe elementy kaset MTCA.4, które zawierają zwykle kilka modułów rozszerzających AMC (*Advance Mezzanine Card*), w tym moduły rozszerzające typu FMC (*Field-Programmable Gate Array (FPGA) Mezzanine Card*), moduły MCH (*MTCA Carrier Hub*), CU (*Cooling Unit*) oraz PM (*Power Module*).

Kolejny podrozdział Wstępu zawiera opis komercyjnego graficznego interfejsu użytkownika MathWorks Simulink i pakietu programów Xilinx System Generator, z których pierwszy jest już używany od ok. 20 lat przez projektantów układów elektronicznych, stosowanych do cyfrowego przetwarzania sygnałów i do regulacji. Środowisko MathWorks Simulink wyposażono w dużą bibliotekę, co ułatwia projektowanie i modelowanie układów cyfrowych, np. takich jak regulatory (sterowniki) PID, czyniąc MathWorks Simulink doskonałym narzędziem do weryfikacji koncepcji projektowanego układu elektronicznego. Według Autora, niedostatkami tego graficznego interfejsu jest brak możliwości uwzględnienia podzespołów elektronicznych, z użyciem których projekt będzie realizowany. Jest to bardzo istotne w przypadku powszechnie obecnie stosowanych programowalnych układów scalonych FPGA, których optymalne wykorzystanie musi uwzględniać szybkość wykonywanych operacji, szybkość odczytywania i format danych, opóźnienia czasowe jak również dostępność zasobów.

W dalszej kolejności opisany jest krótko pakiet programów System Generator firmy Xilinx, produkującej układy scalone FPGA używane w systemach elektronicznych do synchronizacji. Pakiet Xilinx System Generator jest graficznym interfejsem uzupełniającym MathWorks Simulink. Zawiera on standardowe elementy niezbędne do cyfrowego przetwarzania sygnału, takie jak: sumatory, mnożniki czy akumulatory. Bloki pakietu Xilinx System Generator mogą być poddane syntezie logicznej. Następnie dodatkowe programy Xilinx mogą być użyte do mapowania, place&route i do wygenerowania końcowego pliku programującego FPGA.

Rozdział II: Motywacja (Motivation)

W pierwszym podrozdziale Rozdziału II mgr inż. Prędko podaje opis metodologii realizacji oprogramowania systemowego (firmware'u) współczesnych układów cyfrowych do przetwarzania sygnałów na potrzeby synchronizacji optycznej. W realizacji tych układów wykorzystuje się obecnie programowalne układy scalone FPGA, mające duże zasoby bramek logicznych i bloków pamięci, posiadające również bloki wejściowo-wyjściowe (*ang. Input-Output Blocks*) i elementy takie jak transceivery (*ang. transceiver*). Układy FPGA są zazwyczaj nieco wolniejsze od układów ASIC (*ang. Application Specific Integrated Circuit*), ale są od nich tańsze, a przede wszystkim można je reprogramować, co czyni je atrakcyjniejsze w przypadku zastosowań w urządzeniach badawczych. Do programowania FPGA używane są niskopoziomowe języki opisu sprzętu HDL (Verilog lub VHDL), co jest znacznym utrudnieniem, gdyż każda rekonfiguracja układu wymaga współdziałania programistów znających te języki i nie może być wykonana wyłącznie przez ekspertów od cyfrowego

przetwarzania sygnałów i synchronizacji, którzy zazwyczaj posługują się programem MatLab do opisu procesu. Programowanie w HDL różni się znacznie od programowania w MatLab, gdyż wymaga ono używania arytmetyki stałoprzecinkowej i często zmiany formatu danych, jak również uwzględnienia dostępnych zasobów oraz komunikacji ze sprzętem zewnętrznym.

Następnie Autor omawia metodologię opracowywania firmware'u dla infrastruktury badawczej w DESY. Powstający zbiór programów, w ramach ogólnego projektu LLRF VHDL (*ang. LLRF, Low Level Radio-Frequency*), jest również wykorzystywany w cyfrowych układach do synchronizacji optycznej w laserze FLASH i będzie stosowany w największym na świecie laserze na swobodnych elektronach E-XFEL.

W następnym podrozdziale mgr inż. Prędko omawia trudności stosowania pakietu narzędzi System Generator, powracając do dyskusji tego pakietu komercyjnego firmy Xilinx. Podstawową wadą, według Autora, jest czasochłonne manualne wykonywanie połączeń (*ang. interfacing*) do zewnętrznego sprzętu (urządzeń peryferyjnych), co musi być zrobione każdorazowo, gdy np. zmienia się ilość portów wejścia-wyjścia. Podobna sytuacja ma miejsce w przypadku re-konfigurowalnych portów wejścia-wyjścia platformy National Instruments (NI), w której stosuje się pakiet oprogramowania LabVIEW FPGA Module, będącym graficznym interfejsem ułatwiającym programowanie hardware'u. W ostatnim podrozdziale Autor formułuje dwie tezy Rozprawy.

Teza 1: Using GUI-based tools to design digital signal processing blocks, which can be used in control applications, produces results comparable to those that can be obtained using hardware description language implementation as far as mathematical accuracy, speed of operation and resource utilization are concerned.

Teza 2: It is possible to design a toolset for development of FPGA-based digital signal processing and control projects compatible with the MTCA.4 firmware structure to be used by application engineers with no hardware description language background, allowing them to model, simulate, synthesize and implement those projects.

Tezy Rozprawy sformułowane są w sposób ogólniejszy niż wynikałoby to z jej tytułu, który wskazuje na sterowniki (regulatory) do synchronizacji optycznej laserów na swobodnych elektronach. Dotyczą one, po pierwsze, ogólnie projektowania i modelowania układów cyfrowych do przetwarzania sygnałów opartych na FPGA, kompatybilnych z MTCA.4, po drugie, możliwości opracowania graficznego interfejsu dla niewyspecjalizowanych w HDL projektantów, zachowującego numeryczną dokładność, prędkość obliczeń i efektywność wykorzystania zasobów, w stopniu porównywalnym z projektowaniem tych układów w językach HDL.

Rozdział III: Zastosowanie Algorytmów Cyfrowego Przetwarzania Sygnału w FPGA (Implementation of DSP algorithms in FPGA)

W Rozdziale III mgr Prędko dokonuje porównania programowania FPGA z wykorzystaniem bloków biblioteki Xilinx System Generator i bloków opracowanych w języku VHDL, pochodzących z biblioteki LLRF w DESY oraz z biblioteki OpenCores. Porównanie dokonane jest dla dwóch algorytmów: Coordinate Rotation Digital Computer (CORDIC) i Infinity Impulse Response (IIR) Filter, zarówno pod względem dokładności wyniku końcowego, latencji jak i ilości niezbędnych zasobów.

CORDIC jest algorytmem obliczającym funkcje trygonometryczne i hiperboliczne. Autor wykorzystuje ten algorytm do obliczenia współrzędnych kartezjańskich dla ośmiu wektorów zdefiniowanych we współrzędnych biegunowych. Z porównania przeprowadzonego przez Autora wynika, że błąd średniokwadratowy nie przekracza 4 najmniej znaczących bitów ($6E-5$) dla bloku CORDIC4.0 oraz dla algorytmu zbudowanego z bloków uniwersalnych pakietu System Generator, zarówno z ograniczeniem zakresu jak i bez tego ograniczenia, dla biblioteki LLRF, jak również dla

pięciu z pośród ośmiu wektorów przy zastosowaniu oprogramowania OpenCores. Zgodnie z wyjaśnieniem Autora, znacznie większy błąd dla pozostałych trzech wektorów występuje przy wykorzystywaniu oprogramowania OpenCores, gdyż kąt we współrzędnych biegunowych jest poza przedziałem $\langle -\pi/2, \pi/2 \rangle$. Należy w tym miejscu zaznaczyć, iż z porównania Tabeli 3.2 z Tabelą 3.7 wynika, że jeden z tych trzech wektorów (patrz wiersz 4), którego współrzędne kartezjańskie obarczone są dużym błędem, ma kąt 300° (-60°), będący w przedziale $\langle -\pi/2, \pi/2 \rangle$, co wskazuje na inną przyczynę błędu niż ta wskazana przez Autora.

Następnie mgr inż. Prędko porównuje latencję i wykorzystywanie zasobów. Podsumowanie porównania podane jest w Tabeli 3.8. W końcowym wniosku Autor podkreśla, że użycie bloków uniwersalnych oprogramowania System Generator firmy Xilinx daje wyniki z dokładnością zbliżoną do oprogramowania w VHDL oraz, że wyniki otrzymywane są z mniejszą latencją i z oszczędniejszym użyciem zasobów niż przy użyciu gotowego bloku CORDIC4.0.

Drugi przykład, algorytm Infinity Impulse Response (IIR) Filter, różni się od omawianego algorytmu CORDIC, gdyż wymaga operacji mnożenia, która we współczesnych FPGA wykonywana jest w specjalistycznych jednostkach obliczeniowych (*ang. slice*) dedykowanych do cyfrowego przetwarzania sygnałów. Autor przedstawia realizację filtra Butterwortha 2-rzędu, z użyciem FPGA Virtex-6, wyposażonego w jednostki obliczeniowe typu DSP48E1 oraz analizuje realizację tego filtru z użyciem oprogramowania System Generator i programów w języku VHDL z biblioteki LLRF. Analiza tego przykładu podsumowana jest przez Autora w Tabeli 3.11. Użycie uniwersalnych bloków System Generator daje najmniejszą latencję, ale używa dwukrotnie większą ilość zasobów, w porównaniu z implementacją wykorzystującą oprogramowanie VHDL z biblioteki LLRF. Zastosowanie oprogramowania System Generator z wykorzystaniem DSP48E1 potrzebuje małej ilości zasobów, ale charakteryzuje się znacznie niższą częstotliwością pracy w porównaniu z pozostałymi oprogramowaniami.

Również w przypadku tego przykładu Autor wnioskuję, że zastosowanie graficznego interfejsu użytkownika System Generator firmy Xilinx, ułatwiającego znacznie realizację tego algorytmu programującym bez lub z małym doświadczeniem w programowaniu w języku VHDL, nie powoduje obniżenia efektywności realizacji zarówno pod względem latencji jak i potrzebnych do realizacji zasobów.

Rozdział IV: RapidX

W Rodziale IV opisana jest biblioteka RapidX autorstwa mgr. Prędkiego, będąca rozszerzeniem graficznych interfejsów MathWorks Simulink i Xilinx System Generator, umożliwiającą graficzne projektowanie układów cyfrowych do przetwarzania sygnałów, wykonywanych w oparciu o architekturę MTCA.4 i wykorzystujących FPGA.

W pierwszym podrozdziale Autor omawia szczegółowo opracowane przez niego elementy graficznego interfejsu RapidX, między innymi: bloki przetworników i pamięci, bloki monitorów i wyzwajające, bloki realizacji funkcji matematycznych i opóźniające. Ilość przedstawionych elementów biblioteki Rapidx świadczy o złożoności podjętego przez Autora zadania realizacji tego graficznego interfejsu. W drugim podrozdziale Autor opisuje krótko, jakie dodatkowe oprogramowanie jest konieczne do korzystania z biblioteki RapidX, która dla kompleksowego programowania wymaga jedynie programów MathWorks Matlab z Simulink i Xilinx ISE (Integrated Synthesis Environment). Następnie, w kolejnym podrozdziale, omówione są podstawowe kroki realizacji projektu z użyciem biblioteki RapidX. Rozdział zakończony jest podsumowaniem, w którym podkreślona jest przez Autora przydatność graficznego interfejsu użytkownika RapidX w realizacji projektów w architekturze MTCA.4.

Rozdział V: Zastosowania (Applications)

W tym rozdziale przedstawione są dwa przykłady projektów, które są zrealizowane z użyciem RapidX, i które znalazły praktyczne zastosowanie w DESY. Pierwszym jest układ testujący przetworniki analogowo-cyfrowe i cyfrowo-analogowe płyt DFMC-AD16 i DRTM-AD84. Po omówieniu użycia RapidX w projekcie, Autor podaje wyniki testów, polegających na przetwarzaniu sinusoidalnego sygnału o niskiej częstotliwości z generatora przez przetworniki analogowo-cyfrowe i następnie przetworzenie tego sygnału na analogowy przez przetworniki cyfrowo-analogowe. Amplitudy w ten sposób otrzymanych sygnałów sinusoidalnych, nieco odbiegają od oczekiwanych wartości, zwłaszcza dla płyty DRTM-AD84. Autor, jako przyczynę, podaje błędy w firmowym projekcie elektronicznym wejścia przetwornika analogowo-cyfrowego wczesnej wersji tej płyty, która była przez Autora testowana.

Drugim projektem przedstawionym w następnym podrozdziale, jest kompleksowy projekt zawierający dualny regulator (sterownik) PI. Mgr Prędko opisuje kolejno bloki tego projektu realizowalne z wykorzystaniem RapidX, w tym między innymi: przetworniki analogowo-cyfrowe, generator, PI regulator, bloki akwizycji danych czy bloki transmitera LLL (*ang. low latency link transmitter*).

Bardzo istotne jest, opisane przez Autora w dalszej kolejności, zastosowanie tego kompleksowego projektu regulatora do synchronizacji pulsów lasera optycznego, generowanych na częstotliwości 216.666 MHz, z referencyjnym sygnałem wysokiej częstotliwości 1.3 GHz, która jest częstotliwością pracy akceleratorów FLASH i E-XFEL. Zmiana długości wnęki optycznej lasera przy pomocy piezo-elementu, regulowanego sterownikiem PI, pozwala na zmianę częstości 216.666 MHz w zakresie \pm kilku kHz oraz na synchronizację fazy tego sygnału w stosunku do sygnału referencyjnego. Prezentowane wyniki pokazują, że sterownik jest w stanie dostroić częstotliwość i fazę w zakresie odstrojenia do 400 Hz. Dla większego odstrojenia, do 1 MHz, wartość absolutna scałkowanego odstrojenia fazowego wynosi ok. 70 fs, co wskazuje na konieczność dalszej optymalizacji regulacji.

Podsumowując ten Rozdział mgr Prędko zaznacza, że w podanych dwóch przykładach wykorzystano i sprawdzono większość bloków biblioteki RapidX, pozwalających na programowanie układów cyfrowych w architekturze MTCA zawierających scalone układy FPGA bez konieczności używania języków HDL.

Rozdział VI: Podsumowanie (Summary)

Ostatni rozdział jest podsumowaniem Rozprawy. Autor podkreśla w nim, że architektury używane obecnie w dużych infrastrukturach badawczych, takie jak VME, National Instruments RIO, ATCA czy MTCA4, pozwalają na dopasowanie projektowanych układów cyfrowych przetwarzających sygnały i regulujących do indywidualnych wymagań urządzenia badawczego oraz na modyfikację i optymalizację tych układów. Łatwość zmiany samych algorytmów pracy układów, przez inżynierów i naukowców posługujących się graficznymi interfejsami, np. MathWork Simulink, napotyka zwykle w procesach modyfikacji i optymalizacji na utrudnienia spowodowane procesem programowania hardware'u w wysoko specjalistycznych językach HDL. Z tym utrudnieniem poradzono sobie w architekturze National Instruments PXI, dzięki graficznemu interfejsowi użytkownika LabView FPGA Module. Inaczej sytuacja przedstawia się w przypadku architektury MTCA, dla której Xilinx, chcąc choćby częściowo rozwiązać ten problem, opracował graficzny interfejs System Generator do programowania FPGA. Biblioteka System Generator, zawierająca wiele gotowych bloków operacji logicznych i arytmetycznych, takich jak testowane CORDIC i FIR, nie oferuje gotowych rozwiązań, między innymi dla interfejsu transmisji danych między poszczególnymi płytami czy komunikacji z centralną jednostką obliczeniową. Było to motywacją opracowania przez Autora oprogramowania RapidX.

Następnie Autor podsumowuje dowód tez Rozprawy, pokazując, że zarówno dokładność obliczeń, wykorzystanie zasobów i latencja mogą być porównywalne w przypadku programowania układów cyfrowych w HDL z programowaniem przy użyciu graficznego interfejsu Xilinx System Generator oraz że można opracować rozszerzenie tego interfejsu (RapidX), pozwalające na programowanie bez znajomości języków HDL.

Rozdział VII: Dodatek (Appendix)

Dodatek zawiera szczegółowe opisy opracowanych przez Autora dwunastu bloków graficznego interfejsu RapidX oraz opisy siedmiu płyt, dla których wykonano oprogramowanie z użyciem RapidX, które są stosowane w DESY: dwóch typu AMC, czterech typu RTM i jednej typu FMC. Na końcu, Autor zamieścił schematy implementacji IIR i CORDIC, bloków interfejsu graficznego System Generator, omawianych w Rozdziale III.

3. Podsumowanie recenzji i ocena Rozprawy

Podjęte w Rozprawie zagadnienia synchronizacji optycznej w laserach na swobodnych elektronach oraz uproszczenia projektowania i programowania układów cyfrowych, wykorzystywanych w synchronizacji, są bardzo istotne dla eksploatacji tych dużych i niezwykle kosztownych urządzeń badawczych. Dzięki graficznemu interfejsowi System Generator firmy Xilinx i opracowanemu przez mgr. inż. Pawła Prędkiego jego rozszerzeniu RapidX, programowanie układów do cyfrowego przetwarzania sygnałów i regulacji, nie wymaga znajomości języków HDL i może być wykonane znacznie szybciej i bezpośrednio przez inżynierów i naukowców biegłych jedynie w programowaniu algorytmów przy pomocy graficznych interfejsów, np. MathWork Simulink.

Układy cyfrowe do przetwarzania sygnałów i regulacji, stosowane w dzisiejszych urządzeniach badawczych, wymagają często modyfikacji i dopasowania do optymalizowanych warunków pracy tych urządzeń i warunków przeprowadzanych eksperymentów. W związku z tym, coraz częściej buduje się je w architekturze MTCA z wykorzystaniem re-programowalnych układów scalonych FPGA, umożliwiających szybką re-konfigurację tych układów. Zarówno efektywność przyspieszania femto-sekundowych paczek elektronów jak i synchronizacja laserów optycznych z błyskami fotonowymi generowanymi w undulatorach są bardzo istotne ze względu na wysokie koszty eksploatacji laserów na swobodnych elektronach, które szacuje się na kilkadziesiąt tysięcy euro na godzinę pracy dużych laserów, takich jak LCLS w Uniwersytecie Stanforda w Palo Alto czy E-XFEL w DESY w Hamburgu.

Autor udowodnił pierwszą tezę Rozprawy, pokazując na dwóch algorytmach, że programowanie przy pomocy graficznego interfejsu System Generator daje zbliżone rezultaty w dokładności obliczeń, ilości użytych zasobów i latencji, do programowania w językach HDL. Dowód pierwszej tezy pokazuje również celowość opracowania rozszerzenia graficznego interfejsu firmy Xilinx przez graficzny interfejs RapidX, który jest postulowany w drugiej tezie Rozprawy.

RapidX, ułatwiający znacznie programowanie cyfrowych układów z FPGA, jest niewątpliwie oryginalnym i nowatorskim wkładem Autora w rozwoju tematyki programowania przy pomocy graficznych interfejsów użytkownika, wykraczającym poza samą problematykę synchronizacji w laserach, gdyż może on być wykorzystany do programowania dla innych zastosowań. Laboratoryjne potwierdzenie zastosowania sterowników projektowanych z użyciem RapidX do synchronizacji MLO z głównym sygnałem referencyjnym 1.3 GHz w FLASH i E-XFEL jest, z jednej strony, krokiem w kierunku optymalizacji synchronizacji, z drugiej strony, istotnym eksperymentalnym dowodem, potwierdzającym obie tezy Rozprawy.

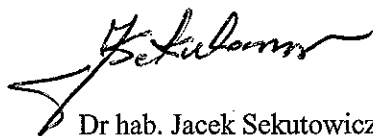
Struktura Rozprawy doktorskiej mgr. inż. Pawła Prędkiego jest przejrzysta. Rozprawa jest bogato ilustrowana, zawiera 112 rysunków, schematów, diagramów i zrzutów ekranowych, z których niestety nie wszystkie są czytelne, zwłaszcza te z czarnym tłem (patrz 3.23-25, 4.12 i 4.15). Należy

również w tym podsumowaniu wspomnieć, że w Rozdziale III wyjaśnienie Autora niedokładności obliczeń algorytmu CORDIC dla oprogramowania OpenCores jest niewystarczające w przypadku jednego z wektorów.

Rozprawa jest napisana w poprawnym języku angielskim, jakkolwiek niektóre sformułowania i czasami zbyt długie zdanie utrudniają zrozumienie tekstu.

Autor nie uniknął kilku nieistotnych pomyłek literowych (patrz strona 18: jest *the* zamiast *then*, strona 37: jest *the* zamiast *The*, strona 48: jest *LBS* zamiast *LSB*, strona 88: jest *is* zamiast *are*). Na stronie 56 jest definicja *LUT*, choć skrót był pięciokrotnie użyty wcześniej, bez definicji. Brakuje odniesienia w tekście do rysunku 4.8 na stronie 57.

Zaliczając tę Rozprawę do rozpraw w pełni spełniających wymagania stawiane przez obowiązujące przepisy o stopniach naukowych, wnioskuję o dopuszczenie mgr. inż. Pawła Prędkiego do obrony pracy doktorskiej. Dodatkowo, doceniając tematykę Rozprawy i cenny wkład Autora w optymalizację pracy dużych i kosztownych urządzeń badawczych, jakimi są lasery na swobodnych elektronach, wnioskuję o wyróżnienie przedstawionej Dysertacji.



Dr hab. Jacek Sekutowicz