

Piotr NIKLAS

POLITECHNIKA GDAŃSKA, KATEDRA ROBOTYKI I SYSTEMÓW MECHATRONIKI
ul. G. Narutowicza 11/12, 0-233 Gdańsk

Realizacja urządzeń automatyki elektroenergetycznej na bazie układów FPGA

Dr inż. Piotr NIKLAS

Absolwent Wydziału Elektroniki i Telekomunikacji Politechniki Gdańskiej, kierunku Automatyka i Robotyka z roku 2000. Doktorat w roku 2004. Obecnie adiunkt w Katedrze Robotyki i Systemów Mechatroniki, Wydziału Elektrotechniki i Automatyki Politechniki Gdańskiej. Tematyka pracy obejmuje zagadnienia związane z automatyką elektroenergetyczną, w szczególności z synchronizacją w systemie elektroenergetycznym. Zainteresowania naukowe również z dziedziny sieci komputerowych.



e-mail: p.niklas@ely.pg.gda.pl

Streszczenie

W artykule opisano wykorzystanie układu FPGA do realizacji układu automatycznej synchronizacji prądnic. Zastosowanie układu FPGA zapewnia w pełni sprzętową realizację procesu synchronizacji. Gwarantuje to deterministyczną i niezawodną realizację procesu synchronizacji. Układ FPGA pozwala również na równoległą realizację poszczególnych zadań procesu synchronizacji.

Słowa kluczowe: układ FPGA, przetwarzanie równoległe, sprzętowa realizacja, układ automatycznej synchronizacji prądnic.

Realization of automatic power system devices based on FPGA chip

Abstract

In the paper there is described implementation of a power object automatic synchronizer with use of the FPGA chip. The FPGA unit is a programmable chip. It is equipped with a specific set of logic elements, among which you can define the network of connections (Fig. 1). In this way, a hardware implementation of the desired functionality of the system is obtained [3]. The task of the automatic synchronizer is to connect a synchronized power object to parallel work, according to the amplitude, frequency and phase conditions. Given the very serious consequences of erroneous execution of the synchronization process [1], automatic synchronizers belong to the group of devices which puts very high demands for reliability. Application of FPGA provides fully hardware realization of the synchronization process. The advantage is high reliability, resulting from elimination of layers of software, which can be a potential source of errors. Another advantage is the true parallel realization of each task of the synchronization process. Each task is carried out in parallel by separate blocks of logic elements, as shown in Fig. 3. This solution also provides fully deterministic execution of the program code. The developed synchronizer enables full registration of parameters of the synchronization process, which is realized by application operating on a PC. Communication between the synchronization process and the application takes place via the Internet and the mechanism of direct memory access DMA. The communication diagram is shown in Fig. 5.

Keywords: FPGA chip, parallel realization, hardware realization, automatic synchronizer of power units.

1. Wstęp

W systemie elektroenergetycznym pracuje wiele różnorodnych urządzeń automatyki elektroenergetycznej. Współcześnie produkowane urządzenia są często urządzeniami mikroprocesorowymi, realizującymi swoje zadania na podstawie realizacji kodu programu. Przykładami takich urządzeń są regulatory prędkości obrotowej oraz regulatory napięcia generatorów synchronicznych, układy automatycznej synchronizacji prądnic, układy automatycznego załączania rezerwy i wiele innych. Urządzenia automatyki elektroenergetycznej z racji odpowiedzialności pełnionych funkcji narzucają duże wymagania co do niezawodności działania. Dodatkowo realizowane funkcje narzucają często wysokie wymagania

co do szybkości przetwarzania informacji przez te urządzenia. Przykładem takiego urządzenia może być układ automatycznej synchronizacji prądnic (UASP). W układzie tym typowo występuje pomiar parametrów dwóch napięć sinusoidalnych oraz ekstrapolacja chwili zgodności fazowej. Nowoczesne rozwiązania UASP realizują dodatkowe funkcje np. posiadają zaimplementowany model układu regulacji prędkości obrotowej oraz kontrolują szybkość zmiany prędkości obrotowej wału prądnicy synchronicznej, w celu niedopuszczenia do zbyt dużych uderów momentu mechanicznego na wale generatora [1]. Techniczna realizacja takiego urządzenia wymusza zastosowanie przetwarzania równoległego. Ze względu na zależności czasowe spowodowane dużą dynamiką procesu synchronizacji zastosowanie przetwarzania sekwencyjnego nie jest możliwe.

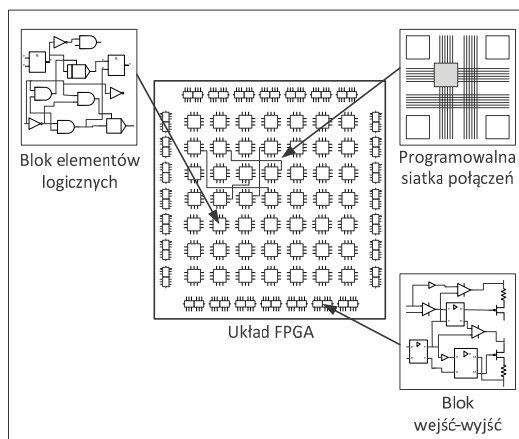
Istnieje kilka możliwości technicznej realizacji takich urządzeń. Jednym z rozwiązań jest zastosowanie systemu wyposażonego w bezpośrednio programowalną macierz bramek określaną skrótowo FPGA (ang. Field Programmable Gate Array). Układy FPGA należą do grupy układów o programowalnej strukturze PLD (ang. Programmable Logic Device). W artykule opisano sposób realizacji zaawansowanego urządzenia automatyki elektroenergetycznej z wykorzystaniem układu FPGA. Jako przykład realizacji wybrano układ automatycznej synchronizacji prądnic, jako szczególnie krytyczny ze względu na zależności czasowe.

2. Charakterystyka układów FPGA

Światowe tendencje wzrostu sprzedaży układów FPGA dowodzą, iż znajdują one coraz szersze spektrum aplikacji. Układ z bezpośrednio programowalną macierzą bramek został opracowany przez firmę Xilinx w roku 1984, raczej jako proste połączenie indywidualnych cyfrowych układów logicznych. Współcześnie obserwuje się, iż układy FPGA zaczynają zastępować użycie układów ze zdefiniowaną na stałe logiką ASIC (ang. Application Specific Integrated Circuits) oraz układów mikroprocesorowych do przetwarzania sygnałów oraz sterowania [2].

Układ FPGA wyposażony jest w określony zbiór elementów logicznych, pomiędzy którymi można definiować siatkę połączeń. Programowanie układu FPGA polega na utworzeniu połączeń pomiędzy elementami logicznymi w określonych punktach siatki. Układy FPGA mogą być reprogramowalne w oparciu o pamięć SRAM lub połączenia są wypalane na stałe (FPGA OTP ang. one-time programmable). Dominującą grupą są układy reprogramowalne. Układy FPGA OTP są wykorzystywane w szczególności do celów wojskowych, gdyż charakteryzują się dużą odpornością na pole elektromagnetyczne np. z broni nuklearnej. Poprzez wykonanie połączeń uzyskuje się sprzętową realizację żądanej funkcjonalności układu. Warto podkreślić, że jest to realizacja wyłącznie sprzętowa, a nie programowo-sprzętowa jak w przypadku rozwiązań mikroprocesorowych. Układy FPGA są zbudowane [3, 4] z matrycowo rozmieszczonych bloków elementów logicznych. Poszczególne bloki elementów logicznych łączone są ze sobą za pomocą magistral linii oraz matryc kluczy, które znajdują się na przecięciu magistral poziomych i pionowych [5]. Na skraju matrycy bloków elementów logicznych umieszczone są bloki wejść-wyjść. Schematyczny sposób budowy układu FPGA przedstawiono na rysunku 1.

Liczba elementów logicznych dla podstawowych układów FPGA wynosi kilka tysięcy, a dla zaawansowanych układów do kilkudziesięciu tysięcy. Wybór odpowiedniej rodziny układów FPGA dostępnych na rynku powinien być podyktowany liczbą wymaganych zasobów logiki i szybkości działania stosownie do wymagań aplikacyjnych.



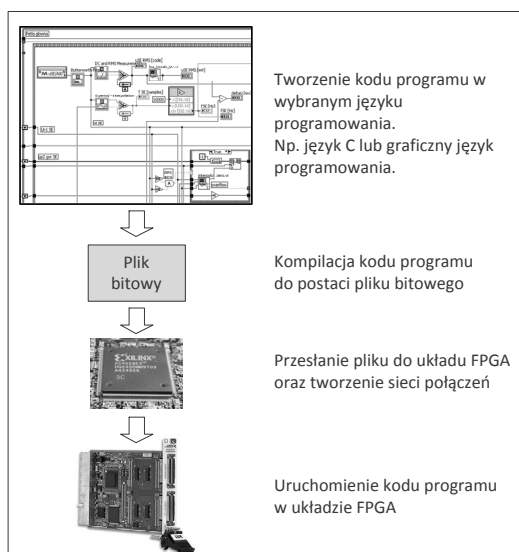
Rys. 1. Budowa układu FPGA
Fig. 1. The FPGA structure

Liczbę dostępnych elementów logiki dla wybranych rodzin układów FPGA podano w tabeli 1. Bloki pamięci LUT (ang. Look Up Table) służą do przechowywania tablicy prawdy funkcji logicznej. Zastosowanie bloków LUT jest szczególnie istotne dla realizacji operacji wykonywanych w pojedynczym cyklu zegarowym. Mnożenie dwóch liczb binarnych wymaga zastosowania dużej liczby indywidualnych bramek logicznych. W celu lepszego wykorzystania zasobów logicznych, układy FPGA są wyposażone w dedykowane bloki realizujące operacje mnożenia.

Tab. 1. Zestawienie liczby elementów logicznych dla wybranych rodzin układów FPGA

Tab. 1. Number of logic elements for selected FPGA families

	Rejestry przesuwające	Bloki LUT	Bloki mnożące	Bloki pamięci RAM (kb)
Virtex-II 1000	10 240	10 240	40	720
Virtex-II 3000	28 672	28 672	96	1 728
Spartan-3 1000	15 360	15 360	24	432
Spartan-3 2000	40 960	40 960	40	720
Virtex-5 LX30	19 200	19 200	32	1 152
Virtex-5 LX50	28 000	28 800	48	1 728
Virtex-5 LX85	51 840	51 840	48	3 456
Virtex-5 LX110	69 120	69 120	64	4 608



Rys. 2. Procedura tworzenia aplikacji na układzie FPGA
Fig. 2. The procedure for creating applications on FPGA chip

Definiowanie siatki połączeń pomiędzy elementami logicznymi odbywa się z poziomu języka opisu sprzętu np. VHDL lub Verilog. Można również użyć środowiska programistycznego wyższego poziomu (np. język C), lub graficznego języka programowania (np. pakiet LabView firmy National Instrument). Procedurę tworzenia aplikacji przedstawiono na rysunku 2.

Układy FPGA łączą zalety stosowania systemów mikroprocesorowych oraz układów ASIC [5]. Stosując układy FPGA uzyskuje się sprzętową, deterministyczną i niezawodną realizację zadanych funkcji. Jednakże bez konieczności długotrwałego i kosztownego procesu projektowania, jakiego wymagają układy ASIC. Powoduje to, że zastosowanie układów FPGA dla urządzeń automatyki elektroenergetycznej, ze względu na stosunkowo niewielki wolumen produkcji jest szczególnie korzystne.

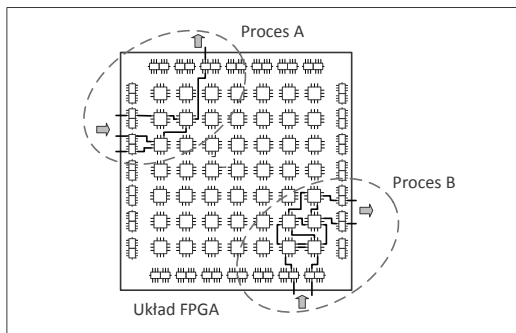
3. Korzyści wynikające z zastosowania układu FPGA do realizacji układu automatycznej synchronizacji prądnic

Realizacja zaawansowanego układu synchronizacji prądnic wymaga spełnienia rygorystycznych zależności czasowych. Do głównych zadań takiego układu realizowanych w całości na układzie FPGA umieszczonym na karcie pomiarowej należy:

- Pomiar częstotliwości napięć sinusoidalnych synchronizowanych obiektów elektroenergetycznych. Sposób pomiaru zastosowany w metodzie bazuje na wykrywaniu chwil przejścia napięcia przez wartość zerową, w odpowiednim kierunku i odpowiedniej sekwencji [6], tak żeby wyeliminować błędy pomiaru częstotliwości w obecności składowych harmonicznych wyższych rzędów.
- Pomiar wartości skutecznej napięć synchronizowanych obiektów elektroenergetycznych.
- Ekstrapolacja chwili zgodności czasowej, z czasem wyprzedzenia równym czasowi własnemu wyłącznika głównego.
- Kontrola wartości przyspieszenia kątownego wału generatora synchronicznego.
- Przeliczanie modelu układu regulacji prędkości kątownej zaimplementowanego w UASP, celem predykcji kierunku i wartości zmian prędkości kątownej.
- Wysyłanie binarnych sygnałów sterujących do układu regulacji prędkości kątownej i regulatora napięcia generatora synchronicznego. Czasy trwania wysłanych sygnałów sterujących dla regulacji prędkości kątownej obliczane są na podstawie parametrów modelu układu regulacji prędkości kątownej.
- Transmisja parametrów synchronizacji do zdalnej aplikacji, odpowiedzialnej za rejestrację parametrów procesu synchronizacji.

Niektóre z wymienionych zadań nie mogą być realizowane szeregowo, gdyż doprowadziłyby to zbyt dużych opóźnień czasowych. Dotyczy to w szczególności przeliczania modelu układu regulacji prędkości kątownej, pomiaru parametrów napięć oraz kontroli wartości przyspieszenia kątownego wału generatora. Wysokie wymagania narzucone są również ze względu na badawczy charakter pracy. Wobec czego zastosowanie systemu z jednym mikroprocesorem nie jest rozwiązaniem w pełni spełniającym postawione wymagania. Zastosowanie układu FPGA umożliwiło umieszczenie kodu programu realizującego powyższe wymienione zadania w osobnych sieciach logicznych. Każda z sieci logicznych wykonywana jest równolegle w pełni sprzętowo z wykorzystaniem niezależnych elementów logicznych. W ten sposób wszystkie wymienione wyżej zadania układu synchronizacji prądnic zrealizowano na jednym układzie FPGA, w osobnych sieciach logicznych wykonywanych równolegle w sposób sprzętowy. Wobec czego cały proces synchronizacji jest wykonywany sprzętowo w układzie FPGA umieszczonym na karcie pomiarowej. Jedynym zewnętrznym elementem jest zdalna aplikacja pracująca na komputerze klasy PC, odpowiedzialna za rejestrację parametrów synchronizacji. Rejestracja parametrów nie ma jednak wpływu

wu na sam przebieg procesu synchronizacji. Koncepcję przetwarzania równoległego w układzie FPGA zilustrowano na rysunku 3.



Rys. 3. Wykorzystanie niezależnych sieci elementów logicznych do równoległej realizacji procesów A i B

Fig. 3. Use of independent logic elements for parallel execution of processes A and B

Równoległa realizacja sieci logicznych jest kluczowa dla prawidłowej i niezawodnej realizacji procesu synchronizacji według zastosowanej metody i umożliwiła jej gruntowne przebadanie. Dodatkowo obsługa wejść pomiarowych oraz binarnych wyjść sterujących realizowana w układzie FPGA odbywa się również w sposób sprzętowy, co gwarantuje minimalne opóźnienia.

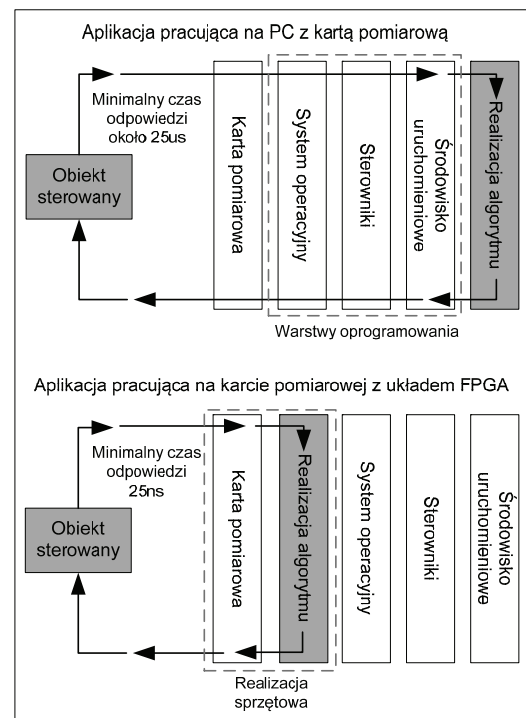
Kolejną zaletą jest stosunkowo krótki czas pisania kodu programu, łatwość jego modyfikacji oraz testowania w środowiskach wysokiego poziomu. Przykładem takiego środowiska jest graficzny język programowania LabView firmy National Instruments. Środowiska programistyczne wysokiego poziomu, są wyposażone w narzędzia do debugowania oraz bogate biblioteki gotowych funkcji. Z uwagi na badawczy charakter pracy był to bardzo ważny czynnik wyboru układu FPGA i programowania w języku wysokiego poziomu. Należy zaznaczyć, iż w celu implementacji programu napisanego w graficznym języku LabView w układzie FPGA, należy go skompilować do poziomu pliku bitowego, definiującego sposób połączeń i przesłać ten plik do układu FPGA. Klasycznie do definiowania funkcji układu FPGA używa się języka opisu sprzętu (np. Verlog, VHDL). Zastosowanie środowiska LabView z punktu widzenia programisty pozwala na ominięcie języka opisu sprzętu. Program jest tworzony w środowisku LabView, następnie jest kompilowany do postaci pliku bitowego i przesyłany do układu FPGA, gdzie następuje wykonywanie odpowiednich połączeń elementów logicznych.

W sytuacji, gdy program wykorzystuje zasoby elementów logicznych układu FPGA w bardzo znacznym stopniu, proces kompilacji kodu napisanego w graficznym języku LabView do pliku bitowego może być stosunkowo długi (np. 6 godzin). Spowodowane jest to próbą optymalizacji liczby wykorzystywanych elementów logicznych.

W przypadku urządzeń automatyki elektroenergetycznej często pełniących odpowiedzialne funkcje, stawia się wysokie wymagania co do niezawodności działania. Wybór układu FPGA do realizacji układu automatycznej synchronizacji prądnic zapewnia wysoką niezawodność. Realizacja procesu w sposób wyłącznie sprzętowy, eliminuje warstwy oprogramowania i sterowników jako potencjalne źródła błędów. Porównanie działania aplikacji sterującej uruchomionej na komputerze klasy PC z kartą pomiarową, z działaniem aplikacji pracującej na układzie FPGA znajdującym się bezpośrednio na karcie pomiarowej przedstawiono na rysunku 4.

Realizacja algorytmu sterowania z układem FPGA dla standardowej częstotliwości taktowania 40Mhz pozwala na uzyskanie czasu odpowiedzi wynoszącego 25ns, przy czym jest to czas gwarantowany. Możliwe jest zastosowanie taktowania z częstotliwością do 200Mhz, co pozwala na uzyskanie czasów odpowiedzi równych 5ns. Wymienione wyżej czasy odpowiedzi dotyczą operacji wykonywanych w jednym taktie zegara. W przypadku

użycia operacji wymagających kilku taktów zegarowych czasy ulegną zwielokrotnieniu. Biblioteki środowiska programistycznego LabView dla układów FPGA, zawierają wiele operacji, które są wykonywane w jednym taktie zegara. Użycie bardziej zaawansowanych funkcji wymaga kilku taktów zegarowych. Obecnie stosowane rodziny układów FPGA pozwalają na realizację w sieci logicznej algorytmu PID z częstotliwością przekraczającą 100kHz. W budowanym układzie automatycznej synchronizacji pętla główna programu, napisanego w graficznym środowisku LabView, jest realizowana z częstotliwością 4kHz. Oznacza to, że napięcie w systemie elektroenergetycznym jest próbkowane 80 razy w ciągu jednego okresu. Z uwagi na zaawansowany algorytm zastosowanej metody synchronizacji, wynik ten należy uznać za bardzo dobry.



Rys. 4. Realizacja algorytmu sterowania na komputerze klasy PC wyposażonym w kartę pomiarową z układem FPGA

Fig. 4. Implementation of the control algorithm on a PC equipped with a data acquisition board with FPGA chip

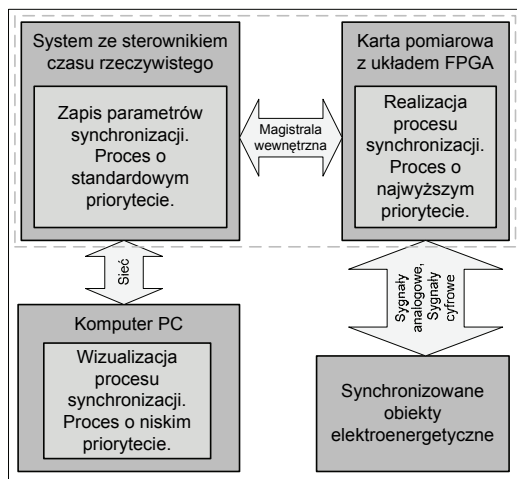
Realizacja sprzętowa jest również w pełni deterministyczna, co jest wymaganiem podstawowym dla systemów czasowo krytycznych, a takim jest proces synchronizacji. W układzie FPGA każdy fragment kodu programu jest wykonywany na oddzielnych elementach logicznych (patrz rysunek 3), co eliminuje problem wspólnego wykorzystywania zasobów przez różne procesy.

4. Realizacja rejestracji parametrów procesu synchronizacji

Z uwagi na badawczy charakter pracy, jednym z wymagań była rejestracja parametrów procesu synchronizacji oraz ich wizualizacja w czasie rzeczywistym. Z uwagi na dużą liczbę rejestrowanych parametrów i dużą dynamikę procesu synchronizacji zadanie to zrealizowano za pomocą zewnętrznej aplikacji uruchomionej na komputerze klasy PC. Połączenie komputera klasy PC z systemem FPGA zrealizowano poprzez połączenie sieciowe. Strukturę połączeń zrealizowanego systemu zaprezentowano na rysunku 5.

Kod programu odpowiedzialny za bezpośrednie wykonywanie procesu synchronizacji, a więc programu o najwyższym priorytecie realizowany jest na karcie pomiarowej wyposażonej w układ FPGA. Karta posiada wejścia/wyjścia analogowe oraz cyfrowe. Kod programu odpowiedzialny za zapis parametrów synchroniza-

cji, który nie jest krytyczny czasowo realizowany jest na sterowniku czasu rzeczywistego z pamięcią stałą. Duża liczba rejestrowanych parametrów synchronizacji wymusiła zastosowanie kanałów DMA (ang. Direct Memory Access) do transferu danych z układu FPGA do sterownika czasu rzeczywistego.



Rys. 5. Schemat blokowy układu automatycznej synchronizacji prądu zrealizowany na układzie FPGA

Fig. 5. Block diagram of the power object automatic synchronizer realized on FPGA chip

Standardowe sposoby przesyłu danych pomiędzy układem FPGA a sterownikiem czasu rzeczywistego w omawianym przykładzie okazały się zbyt mało wydajne. Ponieważ liczba transmitowanych parametrów przekraczała liczbę dostępnych kanałów DMA, zastosowano metodę kompresji danych binarnych. Wizualizacja synchronizacji, jako proces o najniższym priorytecie i nie mający wpływu na przebieg synchronizacji realizowana jest na komputerze klasy PC. W zrealizowanym układzie automatycznej synchronizacji miejsce realizacji poszczególnych procesów uzależnione jest od ich priorytetu. Realizacja zapisu parametrów synchronizacji z uwagi na dużą liczbę generowanych danych nie była możliwa bezpośrednio w układzie FPGA.

5. Badania układu synchronizacji na obiekcie rzeczywistym

Układ automatycznej synchronizacji zrealizowano z użyciem karty pomiarowej NI PXI-7853R wyposażonej w układ FPGA Virtex-5 LX85. Kartę pomiarową umieszczono we wspólnej obudowie z kontrolerem czasu rzeczywistego NI PXI 8106 RT. Raport z implementacji kodu programu synchronizatora na układzie FPGA zawarto na rysunku 6.

Analiza raportu z implementacji wskazuje na zróżnicowane wykorzystanie zasobów logiki dostępnych w układzie Virtex-5 LX85. Należy zwrócić uwagę, że niektóre zasoby zostały wykorzystane w 100%, co może być czynnikiem krytycznym ograniczającym dalszą rozbudowę programu.

Przeprowadzono szereg prób synchronizacji w laboratorium Politechniki Gdańskiej na rzeczywistych obiektach. Jednym obiektem była sieć elektroenergetyczna, o napięciu znamionowym 400 V, drugim obiektem był zespół wytwórczy składający się z generatora synchronicznego (moc znamionowa 27 kVA) napędzanego przez silnik prądu stałego (moc znamionowa 30 kW). Próby synchronizacji prowadzono w zakresie wyrównywania częstotliwości napięć, ekstrapolacji chwili zgodności fazowej oraz sterowania wyłącznikiem głównym. Rezultaty przeprowadzonych prób synchronizacji na obiekcie rzeczywistym w pełni zweryfikowały poprawność działania synchronizatora zbudowanego w oparciu o układ FPGA.

LabVIEW FPGA Bitfile (C) National Instruments, Inc. 2004			
Target: PXI-7853RT			
Compilation Summary -----			
Logic Utilization:			
Number of Slice Registers:	25311	out of 51840	48%
Number of Slice LUTs:	38822	out of 51840	74%
Number used as Logic:	38788	out of 51840	74%
Number used as Memory:	34	out of 13440	0%
Number used as SRL:	34		
Slice Device Utilization Summary:			
Number of BUFs:	3	out of 32	9%
Number of DSP48Es:	48	out of 48	100%
Number of ILOGICs:	47	out of 560	8%
Number of External IOBs:	135	out of 440	30%
Number of LOCed IOBs:	135	out of 135	100%
Number of IOELAYs:	54	out of 560	9%
Number of OLOGICs:	11	out of 560	1%
Number of RAMB18X2SDPs:	1	out of 96	1%
Number of RAMB36_EXPs:	48	out of 96	50%
Number of Slice Registers:	25286	out of 51840	48%
Number used as Flip Flops:	25274		
Number used as Latches:	0		
Number used as LatchThrus:	12		
Number of Slice LUTs:	37778	out of 51840	72%
Number of Slice LUT-Flip Flop pairs:	40977	out of 51840	79%
Clock Rates: (Requested rates are adjusted for jitter and accuracy)			
Base clock: 40 MHz Onboard Clock			
Requested Rate:	40,408938MHz	Theoretical Maximum:	43,312543MHz
Base clock: MiteClk (Used by non-diagram components)			
Requested Rate:	33,037101MHz	Theoretical Maximum:	77,669903MHz
Base clock: ReliableClk40 (Used by non-diagram components)			
Requested Rate:	40,408938MHz	Theoretical Maximum:	231,588698MHz

Rys. 6. Raport z implementacji kodu programu na układzie FPGA

Fig. 6. Report from implementation of the code program on FPGA chip

6. Podsumowanie

W oparciu o układ FPGA zrealizowano zaawansowany układ synchronizacji obiektów elektroenergetycznych. Główne zalety wykorzystania układu FPGA są następujące:

- równoległa realizacja poszczególnych zadań procesu synchronizacji,
- w pełni sprzętowa realizacja, gwarantująca wysoką niezawodność,
- deterministyczna realizacja algorytmu synchronizacji,
- programowanie w języku wysokiego poziomu (włącznie z językami graficznymi),
- łatwość tworzenia i testowania na etapie pisania programu dzięki zaawansowanym narzędziom programistycznym.

Do wad stosowania układów FPGA należy zaliczyć wyższą cenę zakupu w porównaniu do systemów mikroprocesorowych. Dodatkowym ograniczeniem w przypadku bardzo złożonych programów, może być liczba dostępnych zasobów elementów logicznych układu FPGA.

7. Literatura

- [1] Redlarski G.: Nowe rozwiązania automatyzacji procesu łączenia obiektów elektroenergetycznych do pracy równoległej. Wydawnictwo PG, Seria Monografie nr 98, 2009.
- [2] Worchel J.: The Field-Programmable Gate Array (FPGA): Expanding Its Boundaries, InStat Market Research, 2006.
- [3] Clive Maxfield.: The design warrior's guide to FPGAs: devices, tools and flows, Elsevier, 2004.
- [4] Hauck S., DeHon A.: Reconfigurable computing: the theory and practice of FPGA-based computation, Elsevier, 2008.
- [5] Monmasson E., Cirstea M.: FPGA Design Methodology for Industrial Control Systems — A Review, IEEE Transactions on industrial electronics, Vol. 54, No. 4, August 2007.
- [6] Niklas P.: Pomiar częstotliwości napięcia w urządzeniach automatyki elektroenergetycznej, Prace Naukowe Instytutu Maszyn, Napędów i Pomiarów Elektrycznych Politechniki Wrocławskiej, nr.63.