

ZASTOSOWANIA UKŁADÓW FPGA W ALGORYTMACH WYLICZENIOWYCH

Streszczenie: W artykule przedstawiono budowę oraz zasadę działania układów FPGA oraz porównanie do mikrokontrolerów na przykładzie wykonywania algorytmu filtracji FIR. Zostały opisane podstawowe bloki przetwarzania sygnałów dostępne w oprogramowaniu Xilinx ISE.

Słowa kluczowe: układy programowalne, przetwarzanie sygnałów.

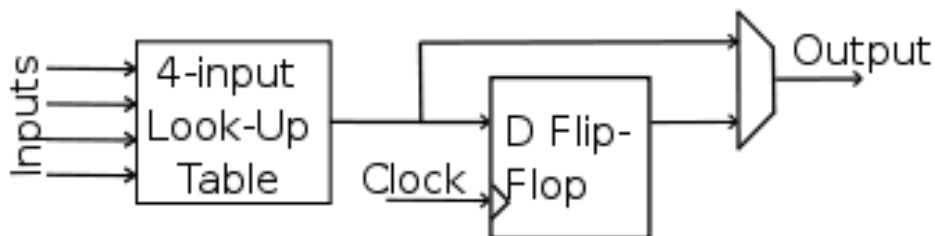
APPLICATIONS OF FPGAS IN ENUMERATION ALGORITHMS

Abstract: The paper presents the structure and the operation principles of FPGAs and microcontrollers to compare the algorithms performance on the FIR filter example. The paper describes the basic signal processing blocks available in the Xilinx ISE software.

Keywords: programmable circuits, signal processing.

1. Wstęp

Field Programmable Gate Array (FPGA, z ang. bezpośrednio programowalna macierz bramek) to rodzaj programowalnego układu logicznego. Dla projektanta układy FPGA mają funkcjonalność taką samą, jak układy typu ASIC [1], jednak mogą być wielokrotnie przeprogramowane po tym, jak już zostały wytworzone, zakupione i zamontowane w urządzeniu docelowym. Największymi dostawcami tego typu układów są firmy Altera Corp. i Xilinx, a także Actel, Atmel, Cypress, Lattice Semiconductor i QuickLogic.



Rys. 1. Typowy element logiczny struktury FPGA

Na ogół układy FPGA zawierają rozmieszczone matrycowo bloki logiczne CLB (ang. Configurable Logic Block) [Rys. 1]. Poszczególne bloki są łączone ze sobą za pośrednictwem linii traktów połączeniowych (routing channels) oraz programowalnych matryc kluczy

połączeniowych umieszczonych w miejscu krzyżowania się traktów poziomych i pionowych. Na obrzeżach matrycy bloków logicznych znajdują się programowalne bloki IOB (wejściowo-wyjściowe). Struktury FPGA zawierają od 64 do dziesiątków tysięcy bloków logicznych o bardzo zróżnicowanej budowie. Bloki logiczne mogą być bardzo złożone, jest ich wówczas mniej w układzie lub względnie proste i jest ich wówczas więcej. Zazwyczaj złożone bloki logiczne zawierają dwie lub więcej pamięci RAM umożliwiających tworzenie tablic wartości funkcji LUT (Look-up Table) i dwa lub więcej przerzutników. W większości układów są to tablice czterowejściowe (pamięć RAM o pojemności 16 bitów). W układach o prostszej budowie, bloki logiczne zawierają zwykle dwuwejściowe układy generacji funkcji kombinacyjnych lub multipleksery czterowejściowe i ewentualnie przerzutniki.

Nowoczesne układy FPGA mają możliwość przeprogramowania „w locie” poprzez zastosowanie mechanizmu częściowej rekonfiguracji, co prowadzi do pomysłu rekonfigurowalnego komputera czy rekonfigurowalnego systemu - czyli układów, które mogą dostosować swoją strukturę tak, by lepiej sprostać zadaniom, przed którymi stoją w danym momencie.

Układy FPGA używane są w cyfrowym przetwarzaniu sygnałów, lotnictwie i wojsku, w fazie prototypowej układów ASIC i w wielu innych dziedzinach. Na przykład układy FPGA firmy Xilinx z powodzeniem sprawdziły się w misji na Marsa zakończonej lądowaniem łazików Spirit i Opportunity.



Rys. 2. Układ z rodziny Virtex firmy Xilinx

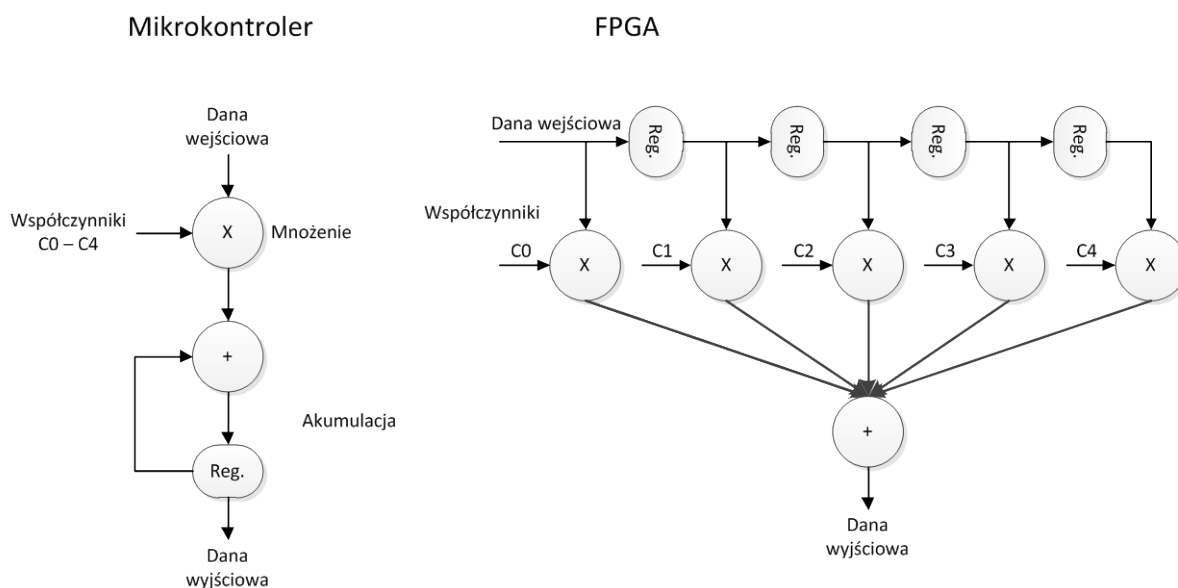
Aby zdefiniować zachowanie układu FPGA używa się języka opisu sprzętu takiego jak Verilog czy VHDL. Następnie przy pomocy narzędzi syntezy generuje się listę połączeń, która następnie w procesie implementacji jest odwzorowywana w konkretnym układzie. Należy zwrócić uwagę, że proces syntezy dopuszcza tworzenie układów logicznych dowolnych rozmiarów, podczas gdy proces implementacji jest próbą wpisania go do konkretnego układu FPGA, gdzie może zabraknąć zasobów do realizacji zadanej logiki. Do zaprogramowania układu FPGA służy plik binarny, który zawiera informacje o konfiguracji układu.

Układy FPGA to jedna z dwóch rodzin programowalnych cyfrowych układów logicznych o dużym stopniu scalenia - drugą grupę stanowią układy CPLD (Complex Programmable Logic Device) o nieco innej architekturze. [1]

2. Mikrokontroler kontra układ programowalny

Na rys. 3 zostało przedstawione porównanie algorytmu filtra FIR wykonywanego przez

mikrokontroler oraz układ programowalny FPGA. Procesor pobiera próbki sygnału i kolejno każdą przemnaża przez odpowiedni współczynnik oraz sumuje wynik z pozostałymi. W przypadku gdy należy wykonać mnożenia dla dwóch lub więcej kanałów, mikrokontroler musi wykonać obliczenia jedno po drugim, co dodatkowo wydłuża czas potrzebny na przetworzenie sygnału.

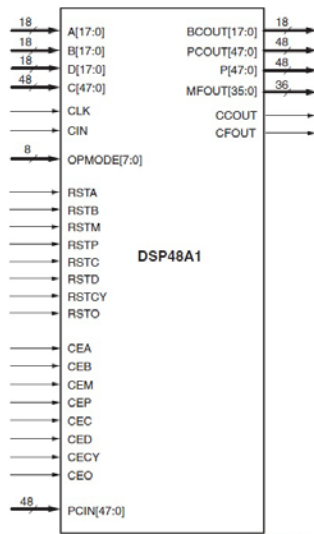


Rys. 3. Porównanie algorytmu FIR pomiędzy procesorem a układem FPGA

Rysunek 3 przedstawia porównanie wykonywania operacji MAC (ang. Multiply ACcumulate – mnoż i akumuluj), na przykładzie filtracji FIR[3], przez procesor oraz układ FPGA. Mikrokontroler wykonuje tyle operacji mnożenia i akumulacji, ile współczynników posiada filtr zaimplementowany w oprogramowaniu. Przy założeniu, że każda operacja zajmuje jeden cykl zegarowy (podczas pracy potokowej), to przy filtrze o długości 128 współczynników oraz zegarze 100MHz czas potrzebny na wykonanie algorytmu będzie wynosił $128 \cdot 10\text{ns} = 1280\text{ns}$. Ograniczeniem dla procesora jest tutaj czas upływający pomiędzy kolejnymi próbkami. Często oznacza to konieczność kompromisu pomiędzy próbkowaniem a długością filtru. W przypadku układu programowalnego czas potrzebny na wykonanie tych samych obliczeń będzie wynosił 10ns, przy tym samym zegarze taktującym, i jest on niezależny od ilości współczynników. Dla dłuższego filtru różnica w czasach będzie odpowiednio większa ponieważ procesor musi wykonać więcej działań, natomiast układ FPGA wykonuje algorytm w sposób równoległy.

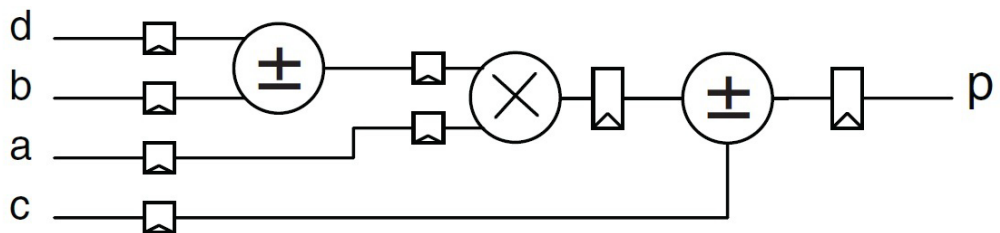
3. Bloki przetwarzania sygnałów DSP

Układy FPGA posiadają również wyspecjalizowane bloki DSP wykorzystujące operacje mnożenia i akumulacji. Na przykład w oprogramowaniu firmy Xilinx są gotowe moduły realizujące algorytmy obliczeniowe takie jak FFT, filtracje. Rysunek 4 przedstawia blok DSP48A1 realizujący różne działania na liczbach wejściowych. Blok ten posiada również wejścia oraz wyjścia umożliwiające łączenie kaskadowe modułów. Schemat blokowy modułu przedstawia rysunek 5.



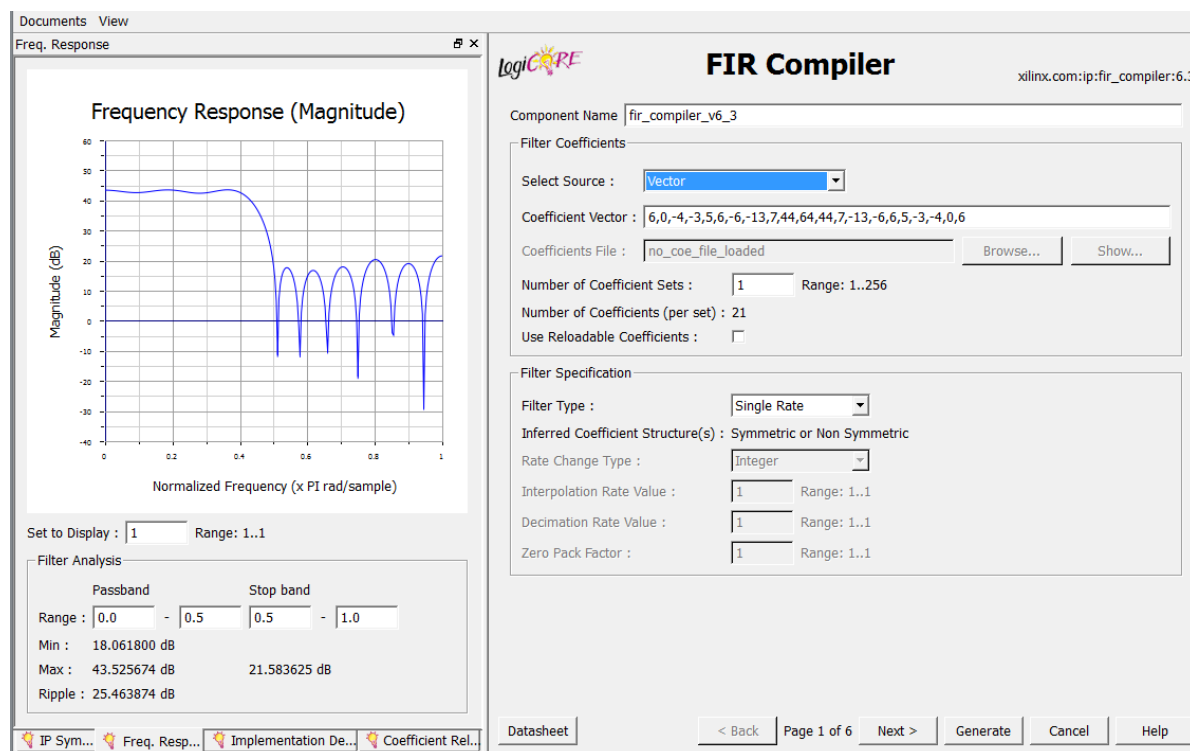
Rys. 4. Blok DSP dostępny w układach firmy Xilinx

Dostępny jest również blok realizujący filtrację FIR [Rys. 6]. Dużym ułatwieniem jest to, że można sobie zaprojektować filtr którego rzeczywista charakterystyka jest przedstawiana w oknie. Po ustaleniu parametrów jest generowany moduł, który można wielokrotnie dołączać do projektu. W oprogramowaniu są również takie bloki jak DDS (Direct Digital Synthesis) czy realizujące szybką transformatę fouriera (FFT).



Rys. 5. Schemat blokowy modułu DSP48A1

Układy programowalne umożliwiają również implementacje procesorów softwarowych, które można zaprojektować według potrzeb projektowych. Część układów posiada również wbudowany rdzeń sprzętowy procesora który obudowuje się peryferiami. Umożliwia to zaprojektowanie mikrokontrolera, który następnie można programować w języku C/C++ oraz podłączać do niego moduły obróbki sygnałów - wszystko w ramach jednego układu scalonego. Takie rozwiązanie wykorzystuje zaletę procesora jaką jest łatwość programowania oraz szybkość wykonywania obliczeń przez układy logiczne.



Rys. 6. Ustawienia kompilatora filtracji FIR

4. Podsumowanie

W artykule zostały przedstawione tylko podstawowe możliwości obliczeniowe układów programowalnych FPGA. Istnieje możliwość projektowania modułów wykonujących kompresję obrazu w czasie rzeczywistym, czy wykorzystywanych w różnych rodzajach komunikacji przewodowej oraz bezprzewodowej, np. GSM. Układy te zwiększają również elastyczność płytek drukowanych, przy zastosowaniu tych samych elementów można uzyskać różne funkcjonalności urządzenia po przez wgranie odpowiedniego oprogramowania. Układy FPGA są również często stosowane w urządzeniach prototypowych, w których istnieje konieczność nanoszenia poprawek. Ostateczną wersję programu przekazuje się do firmy, która produkuje gotowe układy scalone bez możliwości przeprogramowania, ale za to pobierające mniej energii. Takie rozwiązanie stosuje się przy produkcji masowej.

Literatura

- [1] <http://pl.wikipedia.org>, 03.12.2012r
- [2] <http://www.xilinx.com>, 03.12.2012r
- [3] <http://www.agh.edu.pl>, 03.12.2012r