STUDIA INFORMATICA 2000
Volume 21 Number 1 (39)

Franciszek GRABOWSKI
Politechnika Rzeszowska, Zakład Systemów Cyfrowych

MOŻLIWOŚCI I OGRANICZENIA ROUTERÓW FPGA

Streszczenie. Artykuł dotyczy możliwości i ograniczeń routerów potokowych, rekonfigurowalnych FPGA. Osiągalna wydajność jest wynikiem zastosowania struktur FPGA, których immanentną cechą jest równoległość w przeciwieństwie do klasycznej, sekwencyjnej architektury von Neumanna. FPGA zapewniają również elastyczność porównywalną z osiąganą w rozwiązaniach programowych. Parametry te są szczególnie istotne, gdy zwiększają się wymagania i zmieniają standardy, np. IPv6. Poddajemy ocenie przepustowość i opóźnienie procesorów potokowych w routerze. Analiza wskazuje, że rekonfiguracja cząstkowa struktur FPGA jest bardziej optymalna niż rekonfiguracja całkowita.

POSSIBILITIES AND LIMITATIONS OF FPGA ROUTERS

Summary. This work presents possibilities and limitations of the pipeline, reconfigurable, FPGA routers. The performance achievable is a direct consequence of the router architecture, which gives direct exposure to the inherent parallelism, presented in the underlying FPGA circuits, in contrast to classical, sequential, von Neumann architectures. FPGAs are reprogrammable so giving reconfigurable routers the unique combination of software-like reprogrammability with custom hardware performance. These parameters are especially important when increase requirements and change standards, for example IPv6. We examine latency and throughput of pipeline processors in router. Analysis shows that component-level reconfiguration is more optimal than partial reconfiguration.

1. Wprowadzenie

Podejmując zagadnienie dotyczące ograniczeń i możliwości routerów musimy wziąć pod uwagę stan dotychczasowych rozwiązań i wynikające stąd ograniczenia, najbardziej ostre

wymagania i oczekiwania użytkowników i idące w ślad za tym propozycje nowych, doskonalszych konstrukcji.

Co się tyczy możliwości aktualnie produkowanych i eksploatowanych routerów, to wynikają one wprost z ograniczeń strukturalnych całej współczesnej informatyki, tj. stosowania sprzętu o sztywnej architekturze sekwencyjnej von Neumanna. Elastyczność systemów komputerowych, a tym samym i routerów, uzyskujemy na drodze programowej, a więc z natury wolniejszej niż realizacje sprzętowe, jakimi są sieci. Tak więc router w swej klasycznej postaci stanowi wąskie gardło sieci.

Odnosząc się do wymagań stawianych routerom, to najbardziej ostre warunki stawia Internet, który jest jądrem e–gospodarki. Jej zasadniczą cechą jest ustawiczny proces deintermediacji, tj. eliminacji ogniw pośrednich w łańcuchu producent–klient. Skutkiem tego jest eliminacja mediatorów typu hurtownie, banki, biura podróży itp. na rzecz relacji bezpośrednich [1]. Ta presja, czy też kompresja czasu, ma charakter uniwersalny. W związku z tym musi dotyczyć również szeroko rozumianej informatyki. W tym obszarze proces deintermediacji ma charakter ustawiczny, stąd np. ewolucja interfejsów poprzez karty dziurkowane, klawiatury, interfejsy graficzne do interfejsów multimedianych, czy też eliminowanie kart elektronicznych na rzecz identyfikacji bezpośredniej osobnika na podstawie cech biometrycznych lub antropometrycznych. W informatyce najbardziej spektakularną deintermediacją jest odchodzenie procesorów od sekwencyjnej struktury von Neumanna na rzecz struktur rekonfigurowalnych, których immanentną cechą jest równoległość. W ten sposób eliminujemy zbędne fazy, posuwisto–zwrotne, cyklu rozkazowego w relacji procesor–pamięć. Formuła ta poprawiająca relację czas obliczeń/czas komunikacji musi dotyczyć, co jest naturalne, również sieci komputerowych. Dotyczy to szczególnie routerów, gdyż obecnie stosowane wywodzące się, w prostej linii, ze stacji roboczych i zawierające CPU, karty linii i magistralę dzieloną muszą podlegać mechanizmom deintermediacji podobnym jak w przypadku klasycznych procesorów.

Gwałtowny rozwój sieci wymaga coraz większej ich przepustowości. Z drugiej strony standardy sieci muszą ewoluować w celu zwiększenia bezpieczeństwa, integracji usług i innych wymagań, takich jak np. standard IPv6. W związku z tym projektanci sieci muszą wybierać technologie, które posiadają cechy łatwej adaptacji do zmieniających się standardów i oczekiwań, nie tracąc z oczu wydajności i przepustowości.

Sieci aktywne i open signaling [2,3] są próbami pogodzenia, na pozór sprzecznych celów, tj. elastyczności i szybkiego routowania. Chociaż techniki te rokują nadzieje na zwiększanie elastyczności sieci, to budzą zastrzeżenia co do wydajności. Wymagania elastyczności sieci aktywnych podsuwają myśl wykorzystania procesorów ogólnego zastosowania do

zaimplementowania algorytmów rekonfiguracji. Jednocześnie ASIC, które posiadają satysfakcjonujące parametry wydajności, są nieelastyczne.

Jedynym z wyjść z tego impasu jest zastosowanie rekonfigurowalnych architektur obliczeniowych pracujących w trybie przetwarzania strumienia. Połączenie szybkości i elastyczności staje się kluczem do przyszłych rozwiązań. Wraz z odpowiednimi modułami obliczeniowymi i interfejsami przetwarzanie strumieniowe daje możliwość konfiguracji na poziomie sprzętu i przetwarzania pakietów z satysfakcjonującą szybkością.

Struktura tego artykułu jest następująca. W sekcji 2 naszkicowano ewolucję routerów wskazując na routery rekonfigurowalne programowo i sprzętowo. Sekcja 3 dotyczy mechanizmów rekonfiguracji Struktur FPGA. Zdefiniowano mechanizmy rekonfiguracji standardowej, bezpośredniej, wielokrotnego kontekstu i paskową. Istota pracy jest zawarta w sekcji 4 dotyczącej możliwości i ograniczeń potokowych rekonfigurowalnych routerów FPGA.

2. Ewolucja routerów

2.1. Routery konfigurowalne programowo

Sprzęt routera wywodzi się w prostej linii ze stacji roboczej. Platforma ta składa się z CPU, kart linii oraz magistrali dzielonej, która obsługuje komunikację relacji CPU – karty linii. CPU wykonuje operacje wymagane przez protokół trasowania i związane z utrzymaniem i obsługą tablicy marszrut. Tak więc kiedy pakiet dotrze do interfejsu wejściowego, wtedy jest kierowany, poprzez magistralę dzieloną, do CPU. CPU wykonuje określone operacje i zwrotnie przesyła pakiet, przez magistralę, do interfejsu wyjściowego. W ten sposób każdy pakiet przemierza dwukrotnie magistralę. Z drugiej strony wydajność przetwarzania jest ograniczona przez możliwości samej CPU. Zatem poprawy wydajności poszukuje się w architekturze routera. Jedną z możliwości kryją w sobie karty linii, które wyposażone w pamięć cache z tablicą trasowania częściowo odciążają CPU [7].

Innym elementem typowego routera jest system operacyjny. Istnieje wiele interesujących rozwiązań w tym względzie. Są to między innymi rozwiązania klasy sieci aktywne i open signaling [2,3].

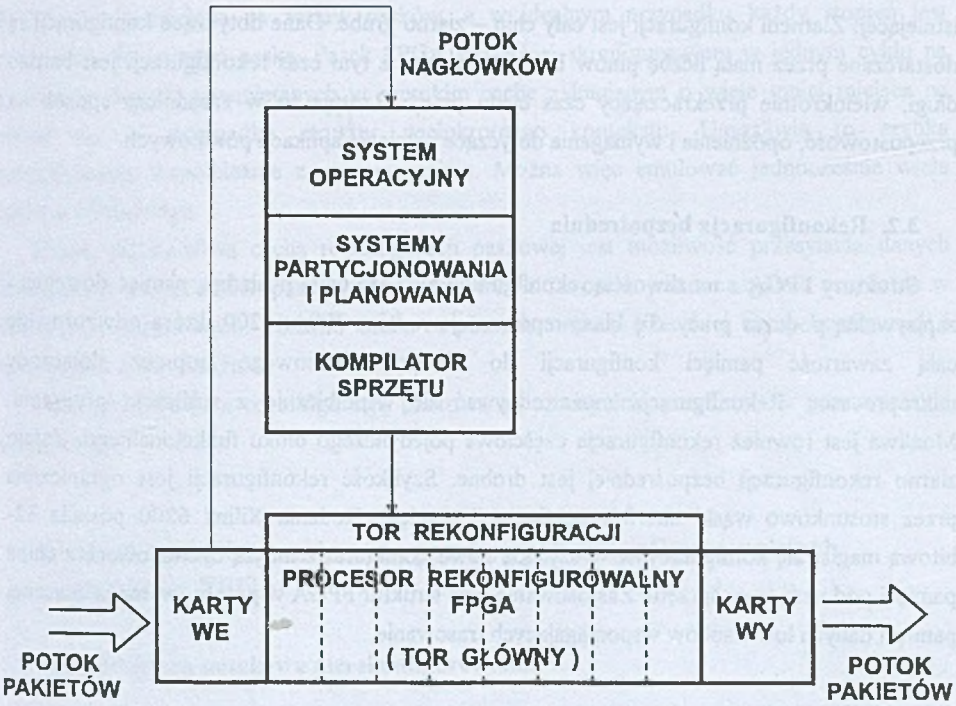
Inny problem dotyczy zastosowania magistrali dzielonej. W danej chwili dostęp do magistrali jest ograniczony do pojedynczej karty linii. W dodatku sama magistrala ma ograniczoną przepustowość. Tak więc kiedy natężenie ruchu w sieci przewyższa możliwości magistrali dzielonej routera, wtedy bufory szybko przepełniają się. Następuje odrzucanie pakietów oraz zwiększenie opóźnienia i zmniejszenie przepustowości routera.

Oprócz ograniczeń natury sprzętowej dotychczasowe rozwiązania są obciążone wadą strukturalną. Są to rozwiązania programowe. Nigdy one nie osiągną pełnej kompatybilności z siecią, która jest sprzętem.

Chcąc uniknąć tych wąskich gardeł, zastosowano struktury specjalizowane ASIC, które gwarantują dużą przepustowość, niską zaś elastyczność. Magistralę dzieloną zastąpiono matrycą przełączników, która umożliwia równoległe przesyłanie pakietów. Jednakże nieuniknione jest zaszywanie algorytmów wyszukiwania adresów w sprzęt rekonfigurowalny dynamicznie podczas wykonywania zadania. Jest to nieuniknione w perspektywie zastosowania w Internecie standardu OC-192, kiedy router będzie musiał wykonywać 10^7 trasowań na sekundę.

2.2. Routery rekonfigurowalne sprzętowo

Router rekonfigurowalny (rys. 1) składa się z bloku sprzętu, planowania i systemu operacyjnego. System operacyjny (SO) zarządza i steruje routerem i może bazować na rozwiązaniach typu open signaling, sieci aktywnych lub innych. SO routera zarządza przetwarzaniem protokołu trasowania oraz ma możliwość odbierania, poświadczania i instalowania nowych protokołów. Proces instalacji polega na tym, że kod protokołu jest ładowany do systemu operacyjnego, partycjonowany i planowany do wykonania na różnych jednostkach sprzętu. System może być partycjonowany funkcjonalnie w ten sposób, że niektóre partie są uruchamiane na procesorze ogólnego zastosowania, niektóre zaś na sprzęcie rekonfigurowalnym. Ciąg informacji zawartych w programie konfiguruje układ pod kątem realizowanego zadania. W rekonfigurowalnym routerze nagłówek ciągu programu pełni rolę protokołu, dane zaś rolę pakietów.



Rys. 1. Schemat blokowy potokowego routera rekonfigurowalnego FPGA
Fig. 1. Schematic diagram of pipeline reconfigurable FPGA router

Jeżeli szybkość zegara linii potokowej routera jest równa szybkości linii, wtedy przetwarzany ciąg nie opóźnia się na routerze. W pełni rekonfigurowalnej strukturze konfigurację można zmieniać podczas przetwarzania. Występują tam np. stopnie: identyfikacja nagłówka, trasowanie pakietu, filtrowanie pakietu, planowanie transmisji pakietu itd.

3. Mechanizmy rekonfiguracji FPGA

3.1. Rekonfiguracja standardowa

Na obecnym etapie rozwoju można wskazać cztery mechanizmy stosowane w rekonfiguracji struktur FPGA [5]: Jest to rekonfiguracja standardowa, bezpośrednia, wielokrotnego kontekstu i paskowa.

Rekonfiguracja standardowa dotyczy popularnych struktur FPGA, takich jak Xilinx 4000 i Altera FLEX, które posiadają jedynie tryb operacyjny i rekonfiguracyjny. Nie ma tam możliwości jednoczesnej pracy i rekonfiguracji, a nawet częściowej modyfikacji konfiguracji

istniejącej. Ziarnem konfiguracji jest cały chip – ziarno grube. Dane dotyczące konfiguracji są dostarczane przez małą liczbę pinów I/O. W związku z tym czas rekonfiguracji jest bardzo długi, wielokrotnie przekraczający czas cyklu pracy. Rzutuje to w zasadniczy sposób na przepustowość, opóźnienia i wymagania dotyczące pamięci i aplikacji potokowych.

3.2. Rekonfiguracja bezpośrednia

Struktury FPGA z możliwością rekonfiguracji bezpośredniej posiadają pamięć dostępną i zapisywalną podczas pracy. Tę klasę reprezentuje rodzina Xilinx 6200, która odwzorowuje całą zawartość pamięci konfiguracji do obszaru adresowego poprzez dołączony mikroprocesor. Rekonfiguracja może odbywać się współbieżnie z realizacją programu. Możliwa jest również rekonfiguracja częściowa pojedynczego bloku funkcjonalnego. Zatem ziarno rekonfiguracji bezpośredniej jest drobne. Szybkość rekonfiguracji jest ograniczona przez stosunkowo wąski interfejs konfiguracji pamięci. Rodzina Xilinx 6200 posiada 32-bitową magistralę konfiguracyjną. Wszystkie nowe konfiguracje muszą być ładowane z chipa pamięci pod nadzorem sprzętu. Zastosowanie tych struktur FPGA w potoku wymaga znacznej pamięci danych lub zasobów wspomagających trasowanie.

3.3. Rekonfiguracja wielokrotnego kontekstu

Mechanizm ten jest podobny do standardowego z wyjątkiem tego, że zamiast jednej konfiguracji może być zrealizowanych wiele kompletnych rekonfiguracji. Logiczna rekonfiguracja całej struktury może być wykonana w czasie porównywalnym z czasem cyklu pracy. Ziarnem konfiguracji jest cały chip. Mechanizm ten rozwiązując problem szybkości rekonfiguracji posiada również ograniczenia. Po pierwsze, nieużywane informacje dotyczące rekonfiguracji muszą być pamiętane w wielu małych, powierzchniowo nieefektywnych pamięciach rozproszonych na całym chipie. Rozmiar komórki tej struktury jest znacznie większy niż komórki standardowej FPGA. Ponadto liczba „wirtualnego" sprzętu emulowana przez FPGA ogranicza zasoby sprzętowe w komórce. Wielokrotna konfiguracja ograniczona jest możliwościami szyny. Implementacja w potoku wymaga znacznej ilości pamięci danych.

3.4. Rekonfiguracja paskowa

Łączy wiele zalet konfiguracji bezpośredniej i wielokrotnego kontekstu oraz posiada wiele cech, które są bardzo przydatne w aplikacjach potokowych. Najbardziej istotną cechą tej rekonfiguracji jest to, że ziarnistość rekonfiguracji może być dobierana w szerokim zakresie i dopasowywana do aplikacji. Potok może być łatwo podzielony na stopnie. Struktura paskowa

FPGA jest podzielona na zestaw pasków, a w idealnym przypadku każdy stopień jest przypisany do jednego paska. Pasek FPGA może być skonfigurowany w jednym cyklu na podstawie danych zapamiętanych w szerokim cache zajmującym o wiele mniej miejsca na chipie niż w przypadku struktur wielokrotnego kontekstu. Umożliwia to szybką rekonfigurację współbieżnie z wykonywaniem. Można więc emulować jednocześnie wiele sprzętu wirtualnego.

Drugą podstawową cechą rekonfiguracji paskowej jest możliwość przesyłania danych konfiguracji do sąsiednich pasków wewnątrz FPGA. Ta cecha jest szczególnie przydatna w aplikacjach potokowych. Ponadto cecha ta pozwala uniknąć szerokiej szyny konfiguracji i związanych z tym opóźnień.

Struktury FPGA z rekonfiguracją paskową są szczególnie przydatne w przetwarzaniu potokowym.

4. Możliwości i ograniczenia potokowych rekonfigurowalnych routerów FPGA

4.1. Struktura potokowa nierekonfigurowalna

Za punkt wyjścia analizy przyjmijmy potok nierekonfigurowalny, np. typu ASIC, zawierający k stopni. Każdy stopień potoku realizuje określoną operację. Załóżmy, że potok wykonuje n zadań (instrukcji). Można zauważyć, że pierwsze zadanie potok wykonuje w czasie $k\tau$, gdzie τ jest okresem sygnału generatora taktującego (czasem trwania taktu zegara). Pozostałe zadania są realizowane w czasie $(n\text{-}1)\tau$. Zatem całkowity czas wykonania n zadań w potoku nierekonfigurowalnym, T_{pn}, jest określony jako:

$$T_{pn} = [k + (n-1)]\tau \tag{1}$$

Biorąc, z drugiej strony, pod uwagę architekturę sekwencyjną (niepotokową), można zauważyć, że w tym przypadku czas wykonania n zadań wynosi:

$$T_s = nk\tau \tag{2}$$

Z kolei przepustowość potoku nierekonfigurowalnego, H_{pn}, zawartego w jednej strukturze, jest definiowana jako stosunek ilości wykonanych zadań n do czasu wykonania określonego równaniem (1). Tak więc:

$$H_{pn} = \frac{n}{[k+(n-1)]\tau} \tag{3}$$

4.2. Rekonfiguracja całkowita

Biorąc pod uwagę architekturę rekonfigurowalną FPGA, można wskazać dwie odmiany rekonfiguracji. Jest to rekonfiguracja całkowita lub rekonfiguracja cząstkowa. Rekonfiguracja całkowita jest możliwa w przypadku struktur FPGA standardowych i wielokrotnego kontekstu, natomiast rekonfiguracja cząstkowa jest realizowana w strukturach o mechanizmach rekonfiguracji bezpośredniej i paskowej. Niezależnie od tego, czy potok jest nierekonfigurowalny czy też rekonfigurowalny, zawiera on k stopni. Jednakże odmienność polega na tym, że potok nierekonfigurowalny składa się z jednej struktury zawierającej wszystkie stopnie, np. ASIC, natomiast potok rekonfigurowalny składa się z N_1 struktur FPGA. Stąd ilość stopni k w strukturze wynosi k/N_1.

W przypadku rekonfiguracji całkowitej cykl pracy polega na tym, że po wykonaniu zadania system rekonfiguruje poszczególne struktury FPGA zmieniając układ całego potoku. Zakładamy, że do rekonfiguracji całkowitej pojedynczej struktury jest niezbędny czas $C_1\tau$, gdzie C_1 jest ilością taktów zegara niezbędnych do rekonfiguracji. W tym przypadku czas wykonania n zadań przez pojedynczą strukturę potoku, tj. $N_1=1$, T_{1r}, wynosi:

$$T_{1r} = \left[\frac{k}{N_1} + (n-1) + C_1\right]\tau \tag{4}$$

Na podstawie równania (4) można określić czas wykonania n zadań przez cały potok, T_{pr}:

$$T_{pr} = N_1\left[\frac{k}{N_1} + \frac{(n-1)}{N_1} + C_1\right]\tau \tag{5}$$

Biorąc pod uwagę równania (1) i (5), możemy określić względne opóźnienie czasowe potoku, tj. stosunek czasu opóźnienia wnoszonego przez potok rekonfigurowalny całkowicie do czasu opóźnienia potoku nierekonfigurowalnego:

$$T_{w1} = \frac{T_{pr}}{T_{pn}} = \frac{N_1\left[\frac{k}{N_1} + \frac{(n-1)}{N_1} + C_1\right]\tau}{[k+(n-1)]\tau} = \frac{k+(n-1)+N_1C_1}{k+(n-1)} = 1 + \frac{N_1C_1}{k+(n-1)} \tag{6}$$

Przepustowość potoku rekonfigurowalnego w całości, H_{pr}, jest dana jako:

$$H_{pr} = \frac{n}{N_1\left[\frac{k}{N_1} + \frac{(n-1)}{N_1} + C_1\right]\tau} \tag{7}$$

Biorąc pod uwagę równania (3) i (7), możemy określić przepustowość względną potoku o rekonfiguracji całkowitej odniesioną do przepustowości potoku nierekonfigurowalnego:

$$H_{w1} = \frac{H_{pr}}{H_{pn}} = \frac{\frac{n}{N_1\left(\frac{k}{N_1} + \frac{(n-1)}{N_1} + C_1\right)\tau}}{\frac{n}{[k+(n-1)]\tau}} = \frac{1}{1+\frac{N_1 C_1}{k+(n-1)}} = \frac{1}{T_{w1}} \tag{8}$$

4.3. Rekonfiguracja cząstkowa

Rekonfiguracja cząstkowa jest realizowana na bazie struktur z mechanizmem bezpośrednim lub paskowym. Biorąc pod uwagę przypadek idealny, rekonfiguracja jest dokonywana współbieżnie z realizowanymi zadaniami. Tak więc następuje ustawiczne, cykliczne zastępowanie pojedynczych, zbytecznych już konfiguracji stopni potoku na nowe konfiguracje wynikające z algorytmu przetwarzania. Ze względu na to, że na obecnym etapie rozwoju technologii struktur FPGA trudno jest realizować rekonfigurację współbieżną, w rzeczywistości stosowana jest rekonfiguracja z przeplotem. W takim przypadku wymagany czas rekonfiguracji stopnia wynosi $\tau N_2 C_2/k$, gdzie C_2 jest ilością taktów zegara niezbędnych do rekonfiguracji cząstkowej. Zatem czas konfiguracji całego potoku wynosi $\tau N_2 C_2$. Dlatego czas opóźnienia transmisji całego potoku jest dany zależnością:

$$T_{pc} = [k+(n-1)+N_2 C_2]\tau \tag{9}$$

Natomiast przepustowość całego potoku można określić jako:

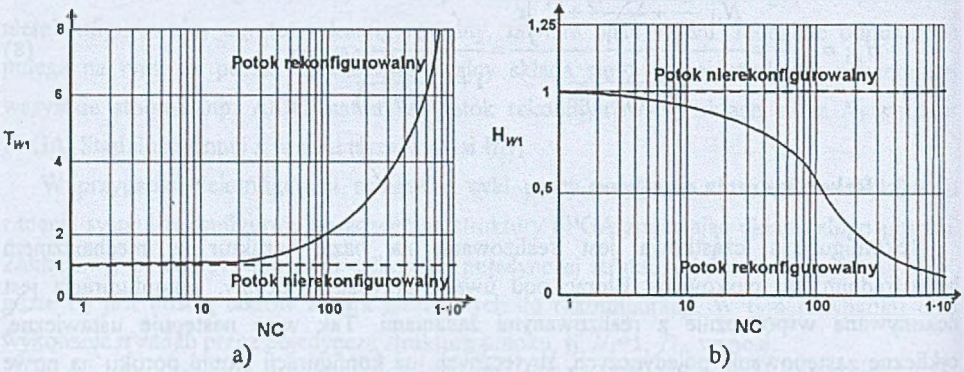
$$H_{pc} = \frac{n}{[k+(n-1)+N_2 C_2]} \tag{10}$$

W przypadku rekonfiguracji cząstkowej można, biorąc pod uwagę równania (1) i (9), określić odpowiednio względny czas transmisji przez potok:

$$T_{w2} = \frac{k+(n-1)+N_2C_2}{k+(n-1)} = 1 + \frac{N_2C_2}{k+(n-1)} \tag{11}$$

oraz względną przepustowość routera:

$$H_{w2} = \frac{\overline{[k+(n-1)+N_2C_2]\tau}^{\,n}}{\overline{[k+(n-1)]\tau}^{\,n}} = \frac{k+(n-1)}{k+(n-1)+N_2C_2} = \frac{1}{1+\dfrac{N_2C_2}{k+(n-1)}} = \frac{1}{T_{w2}} \tag{12}$$



Rys. 2. Względny czas opóźnienia potoku (a), względna przepustowość potoku (b)
Fig. 2. Relative latency of pipeline (a), and relative throughput of pipeline (b)

Biorąc pod uwagę wszystkie przytoczone zależności, można zauważyć (rys. 2), że najistotniejszymi parametrami wpływającymi zarówno na degradację przepustowości, jak i czas transmisji routerów rekonfigurowalnych są: suma $k+(n-1)$ oraz iloczyn NC. O ile $k+(n-1)$ jest w zasadzie zdeterminowane warunkami zewnętrznymi, o tyle iloczyn NC jest w gestii konstruktora. W praktyce $N_1C_1>N_2C_2$. Wynika stąd, że rekonfiguracja cząstkowa jest bardziej optymalna niż rekonfiguracja całkowita. Układy te można optymalizować stosując w miejsce potoku synchronicznego potoki asynchroniczne lub falowe [6].

5. Wnioski

Biorąc pod uwagę aktualny stan zaawansowania technologicznego sieci komputerowych, wymagania i oczekiwania użytkowników oraz dokonującą się reorientację sprzętu komputerowego nierekonfigurowalnego na rekonfigurowalny można zauważyć tendencję

zmian w spojrzeniu na routery, szczególnie IP. Wydaje się, że jest to początek drogi. W artykule przedstawiono podstawowe kwestie z tym związane.

Przewaga routerów rekonfigurowalnych sprzętowo nad rozwiązaniami stosowanymi dotychczas przejawia się poprzez wzrost elastyczności, skrócenie czasu projektowania, wydłużenie okresu eksploatacji, zmniejszenie kosztów modernizacji i konserwacji oraz zmniejszenie kosztów produkcji.

LITERATURA

1. Tapscott D.: Gospodarka cyfrowa, Business Press, Warszawa 1998.
2. Peterson L. L., Karlin S. C., Li K.: OS support for general-purpose routers, Proc. of the Seventh Workshop on Hot Topics in Operating Systems, IEEE, 1998.
3. Pradhan P., Chiueh T. Z.: Operating systems support for programmable cluster-based Internet routers, Proc. of the Seventh Workshop on Hot Topics in Operating Systems, IEEE, 1998.
4. Metz C.: IP routers: New tool for Gigabit networking, IEEE Internet, 1998, November-December, p. 14-18.
5. Schmitt H.: Incremental reconfiguration for pipelined applications, Proc. of the 5th IEEE Symposium on FPGA-Based Custom Computing Machines (FCCM'97).
6. Pahri K. K.: VLSI Digital Signal Processing Systems, John Wiley & Sons, Inc, New York 1999.
7. Duato J., Yalamanchili S., Ni L.: Interconnection Networks. An Engineering Approach, IEEE Computer Society, Washington 1998.

Recenzent: Dr inż. Lech Znamirowski

Wpłynęło do Redakcji 27 marca 2000 r.

Abstract

This work presents possibilities and limitations of the pipeline, reconfigurable, FPGA routers. The performance achievable is a direct consequence of the router architecture, which gives direct exposure to the inherent parallelism, presented in the underlying FPGA circuits, in contrast to classical, sequential, von Neumann architectures. FPGAs are reprogrammable so

giving reconfigurable routers the unique combination of software-like reprogrammability with custom hardware performance. These parameters are especially important when increase requirements and change standards, for example IPv6.

The paper consists from four sections. Section 2 refers of router evolution. Section 3 describes mechanisms of FPGA configuration. Essence of work is contained in section 4.

We examine latency and throughput of pipeline processors using component-level reconfiguration and partial reconfiguration. The point of reference is inflexible (ASIC) pipeline processor. Pipeline inflexible constructed is from one structure FPGA. All applications have k identically - sized pipeline stages.

In the case of component-level reconfiguration pipeline contains N_1 FPGA structure. Thus number of pipeline stage in structure is k/N_1. In this case the time required to complete computation in pipeline in given by equation (5), where C_1 is number of cycle to reconfigure FPGA. Relative time of calculation is expressed by equation (6). The relative throughput of this implementation can be described by the formula (8).

Partial pipeline reconfiguration is a restricted form of local run-time reconfiguration, in which the pipeline is separated into k components, each corresponding to one pipeline stage. During each stage of the computation, we add one additional stage to the reconfiguration, and remove a stage if necessary to keep the amount of configuration within the capacity of the FPGA. Reconfiguration in this manner can be visualized as the scrolling of a window through the computation. Assume that the time it takes to execute a pipeline stage is τ, and the number of execution cycles required to reconfigure this entire FPGA is C_2. Therefore, the time required to substitute a new pipeline stage into the reconfiguration is $\tau N_2 C_2/k$. Thus relative latency and throughput can be described equation (11) and (12), respectively.

Taking into account all quoted dependencies can notice that with most essential parameters influencing both on degradation of throughput and latency reconfigurable router are: sum $k+(n-1)$ and product NC. Sum $[k+(n-1)]$ is determined conditions external. On product NC influences constructor. In practice $N_1C_1>N_2C_2$. Thus can notice that reconfiguration partial is more optimum than reconfiguration entire.

Advantages of the reconfigurable routers, in comparison with classical solutions, are following: increase flexibility, decrease design time, extend system life, reduce part count, reduce development and maintenance costs, and reduce chip fabrication costs through defect tolerance.