

MỘT PHƯƠNG PHÁP ĐIỀU KHIỂN TÁI KIẾN TRÚC PIPELINE CHỨC NĂNG THEO TIÊU CHUẨN ĐỘ TRỄ TỐI THIỂU ML

Chu Đức Toàn^{1*}, Trịnh Quang Kiên², Phạm Minh Tới²,
Hoàng Thị Phương³, Phạm Xuân Bách³, Vũ Anh Tuấn⁴

¹ Trường Đại học Điện lực, ² Học viện Kỹ thuật Quân sự,

³ Trường Đại học Sư phạm Kỹ thuật Nam Định,

⁴ Trường Cao đẳng Kinh tế - Kỹ thuật công nghệ

TÓM TẮT

Sử dụng lý thuyết mạch khóa (Switching Theory) để thẩm định khả năng giảm trễ thao tác trong Pipeline chức năng đạt mức cực tiểu (Minimal Latency - ML), bài báo đề xuất phương pháp tái cấu hình Pipeline bằng phương pháp phân hoạch có sử dụng công nghệ FPGA để thiết lập cấu hình nhanh áp dụng trong thiết kế các hệ xử lý song song chuyên dụng nhằm nâng cao tốc độ tính toán.

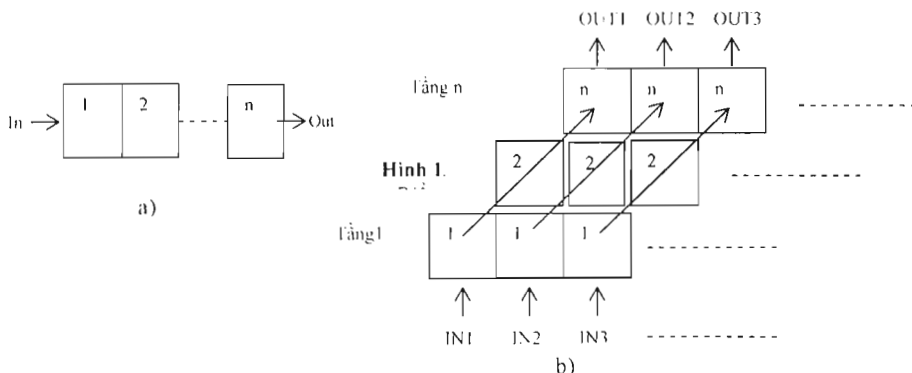
Từ khóa: Điều khiển tái kiến trúc Pipeline, năng cao tốc độ tính toán, công nghệ FPGA, xử lý song song.

ĐẶT VẤN ĐỀ

Nhiều khi tái chiến đấu là những đối tượng rất phức tạp, như những hệ thống vũ khí có điều khiển, tầm xa, khả năng sát thương lớn, giá thành cao [1]. Chúng là sự tích hợp của các hệ cơ, điện, điện-điện tử, điện tử-tin học... với nhiều tham số kỹ thuật có mối quan hệ phức tạp phản ánh tính sẵn sàng chiến đấu. Khi cần giám sát, kiểm tra các tham số của các khí tài này thì yêu cầu phải có đủ số lượng mẫu tại bất cứ thời điểm nào để phân tích tính năng kỹ, chiến thuật theo thuật toán. Điều này dẫn

đến nhiệm vụ tạo hệ có khả năng xử lý tham số song song, cụ thể từ thao tác nối tiếp như hình 1a phải chuyển thành hệ song song trên kiến trúc Pipeline như hình 1b [2,3].

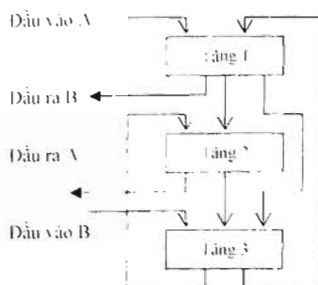
Với phương pháp này, sau n nhịp clock đầu tiên thì cứ mỗi phép xử lý tiếp theo chỉ cần đúng 1 chu kỳ clock. Do vậy tốc độ xử lý về mặt nguyên tắc sẽ tăng lên n lần. Nội dung chính của bài báo là tổng hợp kiến trúc Pipeline tối ưu bằng phương pháp tái kiến trúc theo chuẩn độ trễ tối thiểu.



Hình 1: a) Thao tác nối tiếp; b) Thao tác song song trên kiến trúc Pipeline.

PHƯƠNG PHÁP MÔ TẢ HOẠT ĐỘNG CỦA PIPELINE

	t	t	t	t	t
Tầng 1	A	B		A	B
Tầng 2		A		B	A
Tầng 3	B		AB		



Hình 2: Pipeline và bảng giới hạn Reservation tương ứng

Bảng giới hạn Reservation [4] được sử dụng để mô tả hoạt động của Pipeline. Mỗi tầng của Pipeline được mô tả trong một hàng, mỗi hàng được chia thành nhiều cột, mỗi cột được thực hiện trong một chu kỳ đồng hồ. Hình 2 là cấu trúc Pipeline minh họa và bảng giới hạn Reservation của nó, tại một thời điểm t_i nếu có thao tác diễn ra sẽ được đánh dấu (A cho chức năng thứ nhất, B cho chức năng thứ hai).

Nhịp trễ Latency được định nghĩa là số đơn vị thời gian giữa hai sự khởi đầu độc lập.

Danh sách cấm: Mỗi bảng giới hạn Reservation với 2 hoặc nhiều điểm, trong một hàng sẽ có 1 hoặc nhiều nhịp trễ bị cấm. Danh sách cấm F là một danh sách liệt kê các số nguyên tương ứng với nhịp trễ bị cấm. Với pipeline, số 0 luôn luôn được coi là một nhịp trễ bị cấm.

Vecto xung đột: Một vecto xung đột là một chuỗi số nhị phân có chiều dài $N+1$, với N là nhịp trễ cấm lớn nhất trong danh sách cấm. Vecto xung đột khởi đầu $C(c_n, c_{n-1}, \dots, c_1, c_0)$ được tạo thành từ danh sách cấm F .

Graph trạng thái: Bao gồm các trạng thái có thể có của một Pipeline. Nút graph chứa vector xung đột. Nhánh graph là các cung định hướng, đi ra từ nút i , đi vào nút khác j hoặc chính nút i theo luật "OR với vecto xung đột khởi đầu".

Tiêu chuẩn MAL: MAL là độ trễ trung bình tối thiểu (Minimum Average Latency) của Pipeline cũng là tỉ số nhỏ nhất của tổng độ trễ / tổng số cung graph.

TỔNG HỢP PIPELINE THEO TIÊU CHUẨN ĐỘ TRỄ TỐI THIỂU ML

Cơ sở tổng hợp: Căn cứ lý thuyết mạch khoa (Switching theory) [5,6], ta có thể biểu diễn một phân hoạch 2 lớp cho một hàm logic bất kỳ $F(x_1, x_2, \dots, x_m)$ như sau:

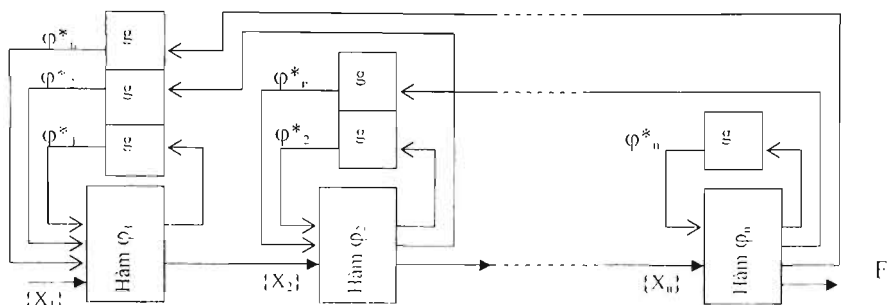
$$F(x_1, x_2, \dots, x_m) = \bigvee_{i=1}^n (\varphi_i(x_1, x_2, \dots, x_m) \wedge \psi_i(x_1, x_2, \dots, x_m))$$

ở đây $\{X\} = \{x_1, x_2, \dots, x_m\}$, $\{Y\} = \{y_1, y_2, \dots, y_k\}$, $\{Z\} = \{z_1, z_2, \dots, z_l\}$ và $\{Y\} \cup \{Z\} = \{X\}$.

Bây giờ mở rộng cho trường hợp F suy biến, tức là có hồi tiếp từ đầu ra của mỗi hàm cơ bản. Hơn nữa nếu hàm hồi tiếp là tuyến tính, thỏa mãn điều kiện $\varphi_i(\{F\}) = \varphi_i(\{F\} \cup I_n)$, với $\{F\} = \{v_1, v_2, \dots, v_k\}$, $I = 0, 1, 2, \dots, l_0$ là nhịp đồng bộ của hệ thống, lúc đó ta sẽ có quan hệ

$$F(x_1, x_2, \dots, x_n) = \bigvee_{i=1}^n (\varphi_i(\dots \varphi_i(\varphi_i(\{X\}, \varphi_1, \varphi_2, \dots, \varphi_n), \varphi_1, \varphi_2, \dots, \varphi_n), \varphi_1, \varphi_2, \dots, \varphi_n))$$

ở đây tập hợp hàm φ_i là các hàm cơ bản ràng buộc chặt, tức là chúng có chức năng không đổi còn tập hợp hàm φ_i có ràng buộc không chặt. Điều này dẫn tới kết luận là nếu thay đổi cấu trúc hàm φ_i sẽ cho phép tạo ra các chức năng khác nhau trong cùng một cấu trúc tập hợp hàm cơ bản. Một phân hoạch tốt là coi các φ_i là cấu trúc chức năng cố định (tương đương như các tầng của Pipeline chức năng) còn các φ_i là các bộ đệm FIFO (luôn luôn thỏa mãn điều kiện $\varphi_i(\{F\}) = \varphi_i(\{F\} \cup I_n)$, có kích thước tùy biến để tương thích với chức năng thứ i của hệ thống (Hình 3).



Hình 3 Phân hoạch hàm logic theo kiến trúc Pipeline chức năng

Để xác định các đặc tính của bảng giới hạn Reservation nhằm đạt được độ trễ tối thiểu mong muốn, chúng ta định nghĩa các thông số sau:

L_c là chuỗi nhịp trễ: là chuỗi thời gian giữa dữ liệu liên tục được đưa vào pipeline. Ví dụ với chu kỳ $C = (2)$ thì $L_c = 2, 2, 2, 2, \dots$

l_i là chuỗi thời gian khởi đầu: là thời gian bắt đầu cho mỗi dữ liệu. Phần tử thứ i ($i > 0$) trong chuỗi là thời gian bắt đầu của dữ liệu khởi đầu thứ i , do đó nó bằng tổng của các độ trễ của các khởi đầu trước đó. Trong ví dụ này, với l_i như trên ta có: $l_i = 0, 2, 4, 6, 8, 10, \dots$

G_c là tập hợp khoảng thời gian khởi đầu: là tập hợp các khoảng thời gian riêng biệt giữa các thời điểm khởi đầu. $G_c = \{t_i - t_j\}$ với mọi $i > j$, trong đó t_i và t_j là phần tử thứ i và thứ j trong chuỗi thời gian khởi đầu, như ví dụ trên có: $G_c = 2, 4, 6, 8, \dots$

Chú ý rằng G_c xác định đặc tính mà bảng giới hạn Reservation phải có để đưa ra chu kỳ C . Nếu một số nguyên i nằm trong G_c , bất kỳ một bảng giới hạn Reservation nào đưa ra chu kỳ C cũng không thể có hai điểm x trên bất kỳ I hàng nào có khoảng cách bằng i đơn vị thời gian (xung đồng hồ).

H_c là tập hợp các khoảng thời gian chấp nhận được được gọi là phần bù của G_c (nghĩa là $H_c \cup G_c = \mathbb{Z}$, trong đó $\mathbb{Z}$ là tập hợp tất cả các số nguyên dương). Với chu kỳ chúng ta đã đưa ra $H_c = \{0, 1, 3, 5, 7, \dots\}$. Vì vậy, bất kỳ

bảng giới hạn Reservation nào đưa ra chu kỳ C phải có các điểm đánh dấu x có khoảng cách cho phép nằm trong H_c . Như vậy H_c cho thấy khoảng cách cho phép giữa các điểm x . Với một cách khác, nếu bảng hạn chế có danh sách cấm I , thì chu kỳ C là hợp lệ nếu: $F \subset H_c$ hoặc $F \cap G_c = \emptyset$.

Vì tập hợp H_c là vô hạn nên khó xử lý trực tiếp. Như vậy, cần giới hạn nó bằng cách tính $H_c \pmod{p}$, trong đó p là khoảng thời gian của chu kỳ C , do chính là tổng các nhịp trễ. Đây là sự phân loại chính xác tất cả các khoảng thời gian cho phép vì chuỗi độ cấm lặp lại với khoảng thời gian p . Xét ví dụ đã đưa: $H_c \pmod{2} = \{0, 1\}$.

Để kiểm tra hoặc xây dựng bảng giới hạn Reservation, phải sử dụng định lý và định nghĩa sau [4]: Hai số nguyên i và $j \in \mathbb{Z}_p$ trong đó $\mathbb{Z}_p$ là tập hợp tất cả các số nguyên nhỏ hơn p , là tương thích đối với $H_c \pmod{p}$ nếu và chỉ nếu $i - j \pmod{p} \in H_c \pmod{p}$. Một tập hợp được gọi là lớp tương thích nếu mỗi cặp phần tử của nó là tương thích.

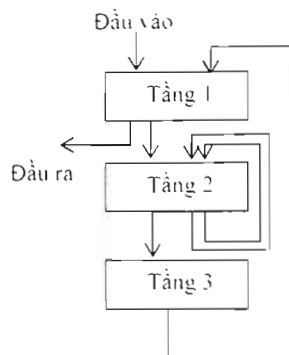
Từ đó tiêu chuẩn ML được xác định cho cấu trúc bảng hạn chế với chu kỳ C phải có các hàng với các điểm x tại các thời điểm sau: $z_1 + i_1 * p, z_2 + i_2 * p, \dots$ trong đó $\{z_1, z_2, \dots\}$ là lớp tương thích của $H_c \pmod{p}$ và $i_1, i_2, \dots$ là các số nguyên tùy ý.

Sử dụng định lý này tối ưu lớp tương thích để xây dựng lại bảng giới hạn Reservation tức là tái kiến trúc cấu trúc của Pipeline.

Bây giờ ta xét tiếp Pipeline có chức năng như được mô tả trên hình 4.

	t	t_1	t_2	t_3	t_4	t_5
Tầng 1	x					x
Tầng 2		x	x	x		
Tầng 3					x	

$F = \{0, 1, 2, 5\}$



Hình 4: Kiến trúc Pipeline và bảng giới hạn Reservation của Pipeline chức năng

Xét Pipeline trên, có: $L_C = 3, 3, 3, 3, \dots$

$I_C = 0, 3, 6, 9, 12, 15, \dots$

$G_C = \{t_i - t_j, \text{ với mọi } i > j\} = 3, 6, 9, 12, \dots$

$H_C = Z - G_C = 0, 1, 2, 4, 5, \dots$

$H_C \pmod 3 = \{0, 1, 2\}$.

Từ đó tiêu chuẩn ML cho Pipeline trên được xác định cho bảng hạn chế với chu kỳ C có các điểm x trong mỗi hàng phải xuất hiện ở các thời điểm $z_1 + i_1*3; z_2 + i_2*3, \dots$ trong đó $\{z_1, z_2, \dots\}$ là lớp tương thích của $H_C \pmod 3$.

Hàng thứ hai của bảng hạn chế ban đầu có hai điểm x tại thời điểm 1, 2 và 3. Để vị trí của các điểm này phù hợp với lớp tương thích $\{0, 1, 2\}$ thì vị trí các điểm phải tương ứng với các thời điểm $z_1 + i_1*3; z_2 + i_2*3; z_3 + i_3*3$ (trong đó $z_1 = 1, z_2 = 2, z_3 = 0$ là lớp tương thích của $H_C \pmod 3$). Có thể thấy rằng điểm x đầu tiên (vị trí 1) phù hợp (vì $z_1 + i_1*3 = 1 + 0*3 = 1$), điểm x thứ hai (vị trí 2) phù hợp (vì $z_1 + i_1*3 = 2 + 0*3 = 2$), điểm x thứ ba (vị trí 3) phù hợp (vì $z_1 + i_1*3 = 0 + 1*3 = 3$). Như vậy chúng ta không cần trì hoãn các điểm x. Đối với hàng thứ nhất, khả năng tuy chọn cao hơn vì chỉ có 2 điểm x nên chỉ cần kiểm tra điều kiện tương thích với $H_C \pmod 3$, cụ thể điểm x đầu tiên (vị trí 0) phù hợp (vì $z_1 + i_1*3 = 0 + 0*3 = 0$), điểm x thứ hai (vị trí 5) phù hợp (vì $z_1 + i_1*3 = 2 + 1*3 = 5$). Đối với hàng thứ cuối, không cần thay đổi. Kết quả là cấu trúc này đã thỏa mãn tiêu chuẩn ML.

KẾT LUẬN

Bài báo khẳng định khả năng luôn tìm được một cấu trúc Pipeline cho phép thỏa mãn tiêu chuẩn độ trễ tối thiểu ML (Minimum Latency) bằng quy trình tổng hợp: Xuất phát từ vectơ xung đột gốc C_0 của Pipeline để dàng xác định các thông số $L_C; I_C; G_C; H_C$, từ đó xác định lớp tương thích của $H_C \pmod p$; $Z_p = \{z_1, z_2, \dots, z_i < p\}$. Căn cứ vào lớp tương thích vừa tìm được, xác định lại vị trí các điểm đánh dấu trên bảng Reservation sao cho phù hợp với quy luật $z_1 + i_1*p; z_2 + i_2*p, \dots$

Khi tiêu chuẩn ML (Minimum Latency) được xác lập thì Pipeline thao tác đạt tốc độ cao nhất. Mặt khác do có cấu trúc tối giản nên độ tin cậy chung của cả hệ thống được cải thiện.

Kết quả này đã được áp dụng cho khâu thiết kế tổ hợp kiểm tra tham số khi tải chiến đấu X35E.

TÀI LIỆU THAM KHẢO

- [1]. Đỗ Xuân Tiên và cộng sự. Bảo cáo kết quả NCKH "Thiết kế, chế tạo khối kết xuất kết quả kiểm tra tên lửa URAN-E trên thiết bị tự động kiểm tra chẩn đoán tham số AKPA do LB Nga chế tạo"-Hà Nội 2008.
- [2]. Akshay Sharma, Carl Ebeling, Scott Hauck. PipeRoute: a pipelining-aware router for FPGAs. Proceedings of the 2003 ACM SIGDA eleventh international symposium on Field programmable gate arrays, February 23-25, 2003, Monterey California, USA

[3]. Deshanand P. Singh, Stephen D. Brown, The case for registered routing switches in field programmable gate arrays, Proceedings of the 2001 ACM SIGDA ninth international symposium on Field programmable gate arrays, pp. 161-169, February 2001, Monterey, California, United States.

[4]. Kai Hwang Perdue University, Faye A. Biggs Rice University, Computer Architecture and Parallel and Processing, McGraw-Hill Book Company, 1999.

[5]. Ashenhurst R.L., The Decomposition of Switching Functions, Ann. Computation Lab., Harvard University, vol. 29, pp.74-116, 1959

[6]. Akshay Sharma, Carl Ebeling, Scott Hauck, PipeRoute: a pipelining-aware router for FPGAs, Proceedings of the 2003 ACM SIGDA eleventh international symposium on Field programmable gate arrays, February 23-25, 2003, Monterey, California, USA

ABSTRACT

A CONTROL METHOD OF RECONFIGURATION FUNCTIONAL PIPELINE STRUCTURES ON THE MINIMUM LATENCY STANDARD

Chu Đức Toàn^{1*}, Trình Quang Kiên²,
Phạm Minh Toi², Hoàng Thị Phương³,
Phạm Xuân Bách³, Vũ Anh Tuấn⁴

¹ Electric Power University,

Academy of Technology and Military,

² Nam Dinh University of Technology Education,

³ Industrial Economic and Technology college

By using switching theory to make a valuation of the ability to reduce delays in pipeline operations achieving the minimum latency (ML), the paper proposed a Pipeline reconfiguration method via the partition using FPGA technology to get have quick configuration, applied in the design of a dedicated parallel processing with the aim of improving the computational speed

Key words. control of reconfiguration functional pipeline structures, improve the computational speed, FPGA technology, parallel processing