



(12) BẢN MÔ TẢ SÁNG CHẾ THUỘC BẰNG ĐỘC QUYỀN SÁNG CHẾ

(19) Cộng hòa xã hội chủ nghĩa Việt Nam (VN)
CỤC SỞ HỮU TRÍ TUỆ

(11)



1-0052334

(51)^{2021.01} H03M 7/30

(13) B

(21) 1-2022-05575

(22) 31/08/2022

(45) 27/10/2025 451

(43) 25/11/2022 416A

(73) TẬP ĐOÀN CÔNG NGHIỆP - VIỄN THÔNG QUÂN ĐỘI (VN)

Lô D26 Khu đô thị mới Cầu Giấy, phường Yên Hoà, quận Cầu Giấy, thành phố Hà Nội

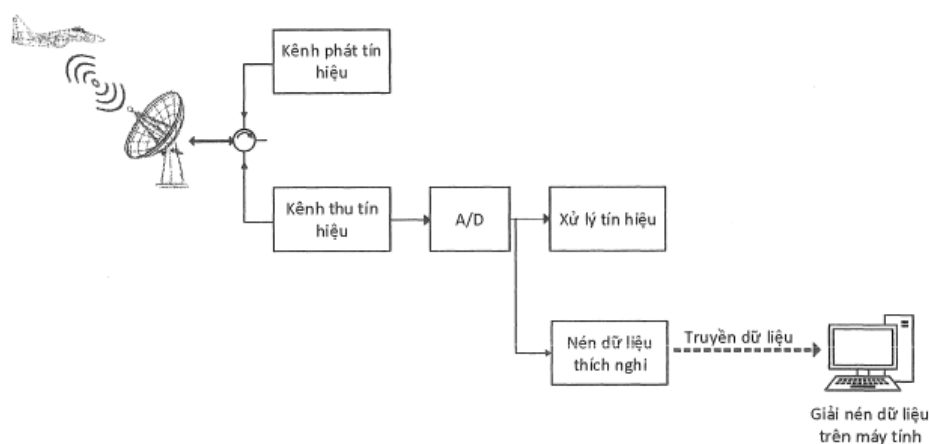
(72) Nguyễn Đình Khuê (VN); Nguyễn Như Thành (VN); Vũ Bá Dương (VN).

(74) Công ty TNHH NACILAW (NACILAW)

(54) HỆ THỐNG VÀ PHƯƠNG PHÁP NÉN VÀ GIẢI NÉN DỮ LIỆU THÍCH NGHI
TĂNG HIỆU SUẤT ĐƯỜNG TRUYỀN DỮ LIỆU

(21) 1-2022-05575

(57) Hệ thống nén và giải nén dữ liệu thích nghi được ứng dụng trong việc truyền dữ liệu và ghi lưu dữ liệu trong các đài ra đa, giúp làm giảm tốc độ đường truyền và giảm dung lượng cần ghi lưu dữ liệu, do đó tiết kiệm được tài nguyên sử dụng của hệ thống. Hệ thống được đề xuất bao gồm hai khối chức năng chính: khối nén dữ liệu thích nghi chạy trên FPGA và khối giải nén dữ liệu thích nghi chạy trên máy tính. Phương án được đề xuất bao gồm việc thực hiện nén và giải nén dữ liệu dựa trên thuật toán FOR và thực hiện thích nghi tổn hao khi nén dữ liệu thông qua việc thay đổi hệ số bit tổn hao dựa vào giá trị trung bình của dữ liệu cần nén và yêu cầu về tỉ số nén và mức độ tổn hao yêu cầu khi xử lý tín hiệu.



Hình 1

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến hệ thống và phương pháp nén và giải nén dữ liệu thích nghi. Cụ thể, hệ thống và phương pháp nén và giải nén dữ liệu thích nghi được đề cập trong sáng chế được ứng dụng trong việc truyền dữ liệu và ghi lưu dữ liệu trong các đài ra đa.

Tình trạng kỹ thuật của sáng chế

Hệ thống và phương pháp nén và giải nén dữ liệu thích nghi được sử dụng trong việc truyền dữ liệu và ghi lưu dữ liệu trong các đài ra đa để tăng hiệu suất đường truyền dữ liệu, giảm tài nguyên sử dụng của hệ thống.

Trong hệ thống các đài ra đa, tín hiệu cao tần phản xạ về từ mục tiêu được thu bởi ăng ten, qua các tầng lọc để đưa tín hiệu về trung tần. Tín hiệu trung tần sau đó được số hóa bằng cách trích mẫu bởi một bộ chuyển đổi tương tự sang số (Analog to Digital Converter -ADC). Dữ liệu được trích mẫu bởi ADC sau đó qua các tầng xử lý tín hiệu tiếp theo để đưa ra thông tin về cự ly, phương vị, vận tốc và độ cao của mục tiêu. Đồng thời, dữ liệu được trích mẫu bởi ADC có thể được truyền tới máy tính và ghi lưu lại để phục vụ cho việc phân tích dữ liệu về sau. Tùy thuộc vào băng thông của tín hiệu, tần số trích mẫu và số bit của bộ ADC, băng thông truyền dữ liệu ghi lưu có thể lên đến vài trăm Mbps đến vài Gbps. Để có thể tăng hiệu suất đường truyền và giảm được tài nguyên của hệ thống, dữ liệu có thể được nén trước khi được truyền tới máy tính. Hệ thống nén và giải nén dữ liệu thích nghi được đề xuất thực hiện nén và giải nén dữ liệu theo thuật toán nén Frame of Reference. Ngoài ra, hệ thống có một hệ số bit tổn hao có thể thay đổi thích nghi cho phép nén và giải nén dữ liệu không tổn hao hoặc có tổn hao để tăng được hiệu suất nén dữ liệu đồng thời vẫn đảm bảo được các chỉ tiêu tổn hao khi xử lý tín hiệu. Đề xuất sử dụng hệ thống và phương pháp nén và giải nén dữ liệu thích nghi giúp cho làm giảm băng thông truyền dữ liệu và giảm dung lượng ghi lưu dữ liệu, do đó tiết kiệm được tài nguyên của hệ thống, đồng thời đảm bảo được khả năng xử lý dữ liệu thời gian thực và đảm bảo chỉ tiêu tổn hao khi xử lý tín hiệu theo yêu cầu.

Bản chất kỹ thuật của sáng chế

Mục đích thứ nhất của sáng chế là đề xuất hệ thống nén và giải nén dữ liệu thích nghi được sử dụng trong việc truyền dữ liệu và ghi lưu dữ liệu trong hệ thống các đài ra đa.

Đầu vào của hệ thống là dữ liệu của tín hiệu thu về bởi ra đa bao gồm dữ liệu mang thông tin của tín hiệu và dữ liệu được số hóa bởi bộ ADC.

Chức năng hoạt động của hệ thống là thực hiện nén và giải nén dữ liệu thích nghi phần dữ liệu được số hóa bởi bộ ADC nhằm mục đích tiết kiệm tài nguyên khi truyền dữ liệu và ghi lưu dữ liệu. Tính thích nghi của hệ thống thể hiện ở chỗ dữ liệu được nén có thể được chuyển đổi một cách tự động giữa nén không tổn hao (tức là thông tin của dữ liệu không bị mất mát sau khi giải nén) hoặc nén có tổn hao (tức là thông tin của dữ liệu có thể bị mất mát sau khi giải nén) để đạt được hiệu suất truyền cao hơn và đồng thời đảm bảo được chỉ tiêu tổn hao khi xử lý tín hiệu thu về của ra đa.

Đầu ra của hệ thống là dữ liệu của tín hiệu thu về bởi ra đa trong đó phần dữ liệu được số hóa ADC vẫn đảm bảo được các chỉ tiêu tổn hao khi xử lý tín hiệu (tức là thông tin của dữ liệu có thể không bị mất mát hoặc có thể bị mất mát nhưng đảm bảo được chỉ tiêu tổn hao)

Để đạt được mục đích đề ra, hệ thống được đề cập trong sáng chế bao gồm hai khối chức năng chính, cụ thể:

Khối nén dữ liệu thích nghi, bao gồm các khối chức năng thành phần: khối xử lý dữ liệu đầu vào, khối nén dữ liệu, khối thích nghi tổn hao, khối truyền dữ liệu. Các khối chức năng này được thực hiện là các khối lô-gíc (Intellectual property core - IP core) chạy trên mạch tích hợp bán dẫn cấu trúc mảng khả trình (Field-programmable gate array - FPGA). Trong đó, khối xử lý dữ liệu đầu vào thực hiện xử lý dữ liệu số hóa bởi bộ ADC. Khối thích nghi tổn hao xác định giá trị hệ số bit tổn hao ứng với mỗi khối dữ liệu được nén dựa trên giá trị trung bình của khối dữ liệu trước đó và thay đổi thích nghi giá trị hệ số bit tổn hao để tăng được tỉ số nén dữ liệu trên đường truyền đồng thời đảm bảo được chỉ tiêu tổn hao khi xử lý tín hiệu. Khối nén dữ liệu nhận đầu vào là dữ liệu đã được xử lý và hệ số bit tổn hao, sau đó thực hiện nén dữ liệu theo thuật toán *Frame of Reference* (FOR).. Khối

truyền dữ liệu có nhiệm vụ đóng gói dữ liệu đã được nén và các thông tin của tín hiệu thu về và gửi tới máy tính.

Khôi ghi lưu và giải nén dữ liệu thích nghi: thực hiện chức năng lưu trữ dữ liệu được truyền đến từ FPGA vào bộ nhớ trên máy tính và thực hiện chức năng giải nén dữ liệu. Việc giải nén dữ liệu được thực hiện theo thuật toán FOR và dựa vào các thông tin đã được đóng gói trong dữ liệu.

Mục đích thứ hai của sáng chế là đề xuất phương pháp thực hiện nén và giải nén dữ liệu thích nghi. Để đạt được mục đích này, phương pháp được thực hiện theo các bước sau:

Bước 1: xử lý dữ liệu đầu vào; đầu vào của bước này là dữ liệu của tín hiệu thu về bởi ra đa bao gồm dữ liệu mang thông tin của tín hiệu và dữ liệu được số hóa từ bộ ADC (dữ liệu ADC). Dữ liệu ADC là giá trị kiểu số nguyên có dấu, có số bit tùy thuộc vào yêu cầu kỹ thuật khi số hóa tín hiệu của đài ra đa. Dữ liệu ADC trước khi được nén sẽ được xử lý để chuyển sang giá trị nguyên không dấu theo nguyên lý: dữ liệu số nguyên có dấu đầu vào có giá trị âm được chuyển thành dữ liệu số nguyên không dấu dương lẻ, dữ liệu số nguyên có dấu đầu vào có giá trị dương được chuyển thành dữ liệu số nguyên không dấu dương chẵn.

Bước 2: xác định giá trị hệ số bit tổn hao phục vụ cho việc nén dữ liệu có thích nghi; hệ số bit tổn hao là số bit thấp được lược bỏ của mỗi dữ liệu đầu vào nén dữ liệu. Dữ liệu ADC sau khi được xử lý sẽ được chia thành các khối, mỗi khối bao gồm N giá trị dữ liệu. Hệ số bit tổn hao sẽ được xác định dựa trên thông tin về giá trị trung bình của các khối. Việc hệ số bit tổn hao thay đổi theo dữ liệu đầu vào thể hiện tính thích nghi của phương pháp.

Bước 3: nén dữ liệu; tại bước này, các khối dữ liệu sẽ được nén theo thuật toán FOR và dựa trên giá trị hệ số bit tổn hao.

Bước 4: đóng gói dữ liệu và truyền dữ liệu; bước này tiến hành đóng gói dữ liệu thành các bản tin bao gồm dữ liệu mang thông tin của tín hiệu thu về bởi ra đa, dữ liệu sau khi nén và hệ số bit tổn hao. Truyền dữ liệu thực hiện truyền các bản tin dữ liệu trên đường truyền *Ethernet* giữa FPGA và máy tính.

Bước 5: ghi lưu dữ liệu; bước này tiến hành ghi lưu dữ liệu được truyền đến từ FPGA vào bộ nhớ của máy tính. Dữ liệu được ghi lưu bao gồm dữ liệu mang thông tin của tín hiệu thu về bởi ra đa, dữ liệu nén và hệ số bit tổn hao.

Bước 6: giải nén và khôi phục thông tin dữ liệu; bước này tiến hành giải nén phần dữ liệu nén đã được ghi lưu theo thuật toán FOR và dựa trên giá trị hệ số bit tổn hao. Dữ liệu sau khi giải nén là các giá trị dương chẵn hoặc giá trị dương lẻ. Khôi phục thông tin dữ liệu sẽ chuyển hóa thông tin dữ liệu về giá trị dương, giá trị âm như dữ liệu gốc ban đầu.

Mô tả vắn tắt các hình vẽ

Hình 1 là sơ đồ kết nối ghép nối hệ thống nén dữ liệu trong hệ thống đài ra đa;

Hình 2 là sơ đồ khối kết nối giữa các khối của hệ thống nén dữ liệu;

Hình 3 là cấu trúc dữ liệu sau khi nén theo thuật toán FOR của mỗi khối dữ liệu.

Mô tả chi tiết sáng chế

Hệ thống và phương pháp nén và giải nén dữ liệu thích nghi được sử dụng trong việc truyền dữ liệu và ghi lưu dữ liệu trong hệ thống đài ra đa. Tham chiếu Hình 1 và Hình 2.

Về đầu vào, đầu ra và chức năng của hệ thống:

Đầu vào của hệ thống là dữ liệu của tín hiệu thu về bởi ra đa bao gồm dữ liệu mang thông tin của tín hiệu và dữ liệu được số hóa bởi bộ ADC.

Chức năng hoạt động của hệ thống là thực hiện nén và giải nén dữ liệu thích nghi phần dữ liệu được số hóa bởi bộ ADC nhằm mục đích tiết kiệm tài nguyên khi truyền dữ liệu và ghi lưu dữ liệu. Tính thích nghi của hệ thống thể hiện ở chỗ dữ liệu được nén có thể được chuyển đổi một cách tự động giữa nén không tổn hao (tức là thông tin của dữ liệu không bị mất mát sau khi giải nén) hoặc nén có tổn hao (tức là thông tin của dữ liệu có thể bị mất mát sau khi giải nén) để đạt được hiệu suất truyền cao hơn và đồng thời đảm bảo được chỉ tiêu tổn hao khi xử lý tín hiệu thu về của ra đa.

Đầu ra của hệ thống là dữ liệu của tín hiệu thu về bởi ra đa trong đó phần dữ liệu được số hóa ADC vẫn đảm bảo được các chỉ tiêu tổn hao khi xử lý tín hiệu (tức là thông

tin của dữ liệu có thể không bị mất mát hoặc có thể bị mất mát nhưng đảm bảo được chỉ tiêu tổn hao).

Cụ thể, hệ thống được đề cập trong sáng chế bao gồm hai khối chức năng chính, cụ thể hơn:

Khối nén dữ liệu thích nghi, bao gồm các khối chức năng thành phần: khối xử lý dữ liệu đầu vào, khối nén dữ liệu, khối thích nghi tổn hao, khối truyền dữ liệu. Các khối chức năng này được thực hiện là các khối lô-gíc (IP core) chạy trên FGPA. Trong đó:

Khối xử lý dữ liệu đầu vào thực hiện xử lý dữ liệu số hóa bởi bộ ADC để dữ liệu thích hợp cho việc thực hiện nén dữ liệu.

Khối thích nghi tổn hao xác định giá trị hệ số bit tổn hao ứng với mỗi khối dữ liệu được nén dựa trên giá trị trung bình của khối dữ liệu trước đó và thay đổi thích nghi giá trị hệ số bit tổn hao để tăng được tỉ số nén dữ liệu trên đường truyền đồng thời đảm bảo được chỉ tiêu tổn hao khi xử lý tín hiệu.

Khối nén dữ liệu nhận đầu vào là dữ liệu đã được xử lý và hệ số bit tổn hao, sau đó thực hiện nén dữ liệu theo thuật toán *Frame of Reference* (FOR).

Khối truyền dữ liệu có nhiệm vụ đóng gói các dữ liệu thông tin của tín hiệu thu về, dữ liệu đã được nén và hệ số bit tổn hao và gửi tới máy tính.

Khối ghi lưu và giải nén dữ liệu thích nghi: thực hiện chức năng lưu trữ dữ liệu được truyền đến từ FGPA vào bộ nhớ trên máy tính và thực hiện chức năng giải nén dữ liệu. Việc giải nén dữ liệu được thực hiện theo thuật toán FOR và dựa vào các thông tin đã được đóng gói trong dữ liệu.

Phương án nén và giải nén dữ liệu thích nghi được đề cập trong sáng chế sử dụng thuật toán FOR để thực hiện nén và giải nén dữ liệu và sử dụng hệ số bit tổn hao để thực hiện việc thích nghi. Hệ số bit tổn hao là số bit thấp của dữ liệu được lược bỏ khi thực hiện nén dữ liệu, số bit lược bỏ càng lớn thì đạt được tỉ số nén càng lớn (do đó tài nguyên hệ thống được tiết kiệm nhiều hơn) nhưng tổn hao dữ liệu cũng lớn và ngược lại. Hệ số bit tổn hao sẽ thay đổi tự động phụ thuộc vào dữ liệu đầu vào để đảm bảo được tỉ số nén tối đa trong phạm vi chỉ tiêu tổn hao cho phép. Để đạt được mục tiêu này, phương pháp thực hiện của sáng chế bao gồm 6 bước cụ thể như sau:

Bước 1: xử lý dữ liệu đầu vào;

Đầu vào của hệ thống là dữ liệu của tín hiệu thu về bởi ra đa bao gồm dữ liệu mang thông tin của tín hiệu và dữ liệu được số hóa từ bộ ADC (dữ liệu ADC). Dữ liệu ADC là giá trị kiểu số nguyên có dấu, có số bit tùy thuộc vào yêu cầu kỹ thuật khi số hóa tín hiệu của đài ra đa. Dữ liệu ADC trước khi được nén sẽ được xử lý để chuyển sang giá trị nguyên không dấu, các bước thực hiện như sau:

- Giá trị dữ liệu là kiểu số nguyên có dấu. Chuyển giá trị dữ liệu thành kiểu số nguyên không dấu như sau:

$$b = \begin{cases} 2a, & \text{nếu } a \text{ là số dương} \\ -2a - 1, & \text{nếu } a \text{ là số âm} \end{cases}$$

Trong đó: a là dữ liệu ADC kiểu số nguyên có dấu, b là dữ liệu sau khi chuyển sang số nguyên không dấu.

Sau khi chuyển sang giá trị số nguyên không dấu, các giá trị dương ban đầu được chuyển thành số chẵn, các giá trị âm ban đầu được chuyển thành số lẻ. Bit thấp nhất của dữ liệu sau khi chuyển thể hiện số chẵn hoặc số lẻ.

Bước 2: xác định giá trị hệ số bit tổn hao phục vụ cho việc nén dữ liệu thích nghi; dữ liệu sau khi được xử lý sẽ được chia thành các khối dữ liệu, mỗi khối bao gồm N giá trị dữ liệu. Giá trị hệ số bit tổn hao ứng với mỗi khối dữ liệu được nén được xác định dựa trên giá trị trung bình của khối dữ liệu trước đó theo nguyên lý giá trị trung bình càng nhỏ sẽ có hệ số bit tổn hao càng nhỏ và ngược lại. Việc thực hiện xác định giá trị của hệ số bit tổn hao bằng cách so sánh giá trị trung bình của khối dữ liệu với các ngưỡng giá trị. Nếu giá trị trung bình nằm trong vùng ngưỡng giá trị nào thì hệ số bit tổn hao sẽ được gán một giá trị tương ứng với ngưỡng đó. Việc đặt giá trị các ngưỡng so sánh và giá trị hệ số bit tổn hao tương ứng có thể thay đổi cài đặt tùy ý để đánh giá, đảm bảo yêu cầu về tỉ số nén và yêu cầu về tổn hao khi xử lý tín hiệu. Theo nguyên lý này, việc nén dữ liệu có thể được thực hiện nén dữ liệu không tổn hao ở những vùng giá trị dữ liệu bé và thực hiện nén dữ liệu có tổn hao ở những vùng có giá trị dữ liệu lớn với tổn hao giá trị dữ liệu có để hạn chế được.

Bước 3: nén dữ liệu;

Bước này thực hiện nén dữ liệu theo thuật toán FOR và dựa vào giá trị hệ số bit tổn hao. Việc thực hiện lần lượt bao gồm các bước:

- Lược bỏ các bit thấp của dữ liệu dựa trên hệ số bit tổn hao. Ví dụ, dữ liệu ADC là 16 bit và được chuyển sang giá trị nguyên không dấu, hệ số bit tổn hao là 4, khi đó dữ liệu được lược bỏ 4 bit thấp và giữ lại bit thấp nhất như sau:

$$b = b_{15} b_{14} b_{13} b_{12} b_{11} b_{10} b_9 b_8 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0$$

chuyển thành:

$$c = b_{15} b_{14} b_{13} b_{12} b_{11} b_{10} b_9 b_8 b_7 b_6 b_5 b_0$$

- Nén dữ liệu theo thuật toán FOR: thuật toán FOR sẽ nén dữ liệu theo từng khối dữ liệu, các bước thực hiện như sau:
 - Thuật toán sẽ đi tìm giá trị lớn nhất (M) và nhỏ nhất (m) của khối dữ liệu;
 - Tìm độ rộng bit tương ứng để biểu diễn dữ liệu của mỗi khối:

$$k = \lceil \log_2(M - m + 1) \rceil$$

Trong đó $\lceil x \rceil$ là phép lấy giá trị nguyên nhỏ nhất lớn hơn x .

- Nén dữ liệu mỗi khối bởi một trường tiêu đề thông tin (header) bao gồm giá trị nhỏ nhất (m) của khối dữ liệu và độ rộng bit k , theo đó là N giá trị dữ liệu có độ rộng k -bit biểu diễn hiệu của mỗi số trong khối với giá trị nhỏ nhất (m).

Tham chiếu Hình 3 để xem cấu trúc dữ liệu sau khi nén của mỗi khối dữ liệu.

Bước 4: đóng gói dữ liệu và truyền dữ liệu;

Bước này tiến hành đóng gói dữ liệu thành các bản tin bao gồm dữ liệu mang thông tin của tín hiệu thu về bởi ra đa, dữ liệu sau khi nén và hệ số bit tổn hao. Truyền dữ liệu thực hiện truyền các bản tin dữ liệu trên đường truyền *Ethernet*.

Bước 5: ghi lưu dữ liệu;

Bước này tiến hành ghi lưu dữ liệu được truyền đến từ FPGA vào bộ nhớ của máy tính. Dữ liệu được ghi lưu bao gồm dữ liệu mang thông tin của tín hiệu thu về bởi ra đa, dữ liệu nén và hệ số bit tổn hao.

Bước 6: giải nén và khôi phục thông tin dữ liệu;

Bước này tiến hành giải nén và khôi phục phần dữ liệu nén đã được ghi lưu theo thuật toán FOR và dựa trên giá trị hệ số bit tổn hao, cụ thể:

- Với mỗi khối dữ liệu nén, lấy thông tin về giá trị nhỏ nhất (m) và số bit k ;
- Dịch dữ liệu mỗi lần k bit và cộng giá trị đó với giá trị nhỏ nhất (m), từ đó có được giá trị dữ liệu được lược bỏ các bit thấp. Tổng cộng cần thực hiện N lần dịch bit và cộng giá trị để giải nén mỗi khối dữ liệu.
 - Khôi phục và hiệu chỉnh giá trị các điểm dữ liệu dựa trên thông tin của hệ số bit tổn hao. Lấy ví dụ về việc giải nén dữ liệu 16-bit, hệ số tổn hao là 4: sau khi giải nén dữ liệu của mỗi khối, mỗi điểm dữ liệu sẽ có giá trị:

$$c = b_{15} b_{14} b_{13} b_{12} b_{11} b_{10} b_9 b_8 b_7 b_6 b_5 b_0$$

Từ giá trị này, sẽ khôi phục lại giá trị nguyên không dấu 16-bit trước khi nén như sau:

$$b' = b_{15} b_{14} b_{13} b_{12} b_{11} b_{10} b_9 b_8 b_7 b_6 b_5 0 0 0 0 b_0$$

lúc này, sai số giữa giá trị nguyên không dấu trước khi nén và sau khi giải nén là Δ , hay $b' = b - \Delta$. Trong đó, $\max(\Delta) = 2^{n+1} - 2$, với n là giá trị hệ số bit tổn hao. ($n = 4$ trong trường hợp ví dụ này).

Số nguyên có dấu được khôi phục lại như sau:

$$a' = \begin{cases} \frac{b'}{2} = \frac{b - \Delta}{2} = a - \frac{\Delta}{2}, & \text{nếu } b' \text{ là số chẵn} \\ \frac{b' + 1}{-2} = \frac{b - \Delta + 1}{-2} = a + \frac{\Delta}{2}, & \text{nếu } b' \text{ là số lẻ} \end{cases}$$

lúc này, sai số giữa giá trị nguyên có dấu ban đầu cần nén và sau khi giải nén là $(a - a') \in [-2^n - 1; 2^n - 1]$.

Để giảm khoảng sai lệch giữa dữ liệu gốc ban đầu và dữ liệu sau giải nén, một giá trị offset = 2^{n-1} được thêm vào khi khôi phục dữ liệu như sau:

$$a' = \begin{cases} \frac{b'}{2} + 2^{n-1} = \frac{b - \Delta}{2} + 2^{n-1} = a - \frac{\Delta}{2} + 2^{n-1}, & \text{nếu } b' \text{ là số chẵn} \\ \frac{b' + 1}{-2} - 2^{n-1} = \frac{b - \Delta + 1}{-2} - 2^{n-1} = a + \frac{\Delta}{2} - 2^{n-1}, & \text{nếu } b' \text{ là số lẻ} \end{cases}$$

lúc này, sai số giữa giá trị nguyên có dấu ban đầu cần nén và sau khi giải nén là $(a - a') \in [-2^{n-1}; 2^{n-1}]$.

Hiệu quả đạt được của sáng chế

Các kết quả thực hiện nén và giải nén dữ liệu với dữ liệu của một số đài ra đa cho thấy phương pháp nén và giải nén dữ liệu thích nghi có thể đạt được tỉ số nén hơn hai lần và đảm bảo được chỉ tiêu tổn hao theo yêu cầu. Kết quả thu được với dữ liệu của các đài ra đa khác nhau:

	Số bit dữ liệu đầu vào nén	Tỉ số nén đạt được	Tổn hao dữ liệu sau nén xung					
			Tổn hao SNR (dB)		Sai lệch biên độ tại đỉnh nén xung (dB)		Sai lệch góc pha tại đỉnh nén xung (độ)	
			Kết quả	Chỉ tiêu	Kết quả	Chỉ tiêu	Kết quả	Chỉ tiêu
Dữ liệu đài VRS-SRS	24	2,8	0,06	< 1	0,95e-13	< 0,1	5,96e-13	< 0,1
Dữ liệu đài VRS-MRS	24	2,6	0,025	< 1	3,6e-3	< 0,1	5,3e-3	< 0,1
Dữ liệu đài VRS-CSX	16	2,2	0,057	< 1	0,052	< 0,1	0,019	< 0,1

Từ các kết quả thử nghiệm với dữ liệu các đài ra đa khác nhau cho thấy phương pháp nén dữ liệu thích nghi được trình bày trong sáng chế có thể làm giảm băng thông dữ liệu đường truyền cũng như dung lượng cần lưu giữ dữ liệu ghi lưu, từ đó có thể làm tăng hiệu suất đường truyền dữ liệu và giảm tài nguyên sử dụng của hệ thống.

Yêu cầu bảo hộ

1. Hệ thống nén và giải nén dữ liệu thích nghi tăng hiệu suất đường truyền dữ liệu trong hệ thống ra đa bao gồm:

khối nén dữ liệu thích nghi, bao gồm các khối chức năng thành phần: khối xử lý dữ liệu đầu vào, khối nén dữ liệu, khối thích nghi tổn hao, khối truyền dữ liệu; các khối chức năng này được thực hiện là các khối lô-gíc (IP core) chạy trên FGPA; trong đó:

khối xử lý dữ liệu đầu vào thực hiện xử lý dữ liệu số hóa bởi bộ ADC để dữ liệu thích hợp cho việc thực hiện nén dữ liệu;

khối thích nghi tổn hao thay đổi thích nghi giá trị hệ số bit tổn hao phụ thuộc vào đầu vào của hệ thống để tăng được tỉ số nén dữ liệu trên đường truyền đồng thời đảm bảo được chỉ tiêu tổn hao khi xử lý tín hiệu;

khối nén dữ liệu nhận đầu vào là dữ liệu đã được xử lý và hệ số bit tổn hao, sau đó thực hiện nén dữ liệu theo thuật toán *frame of reference* (FOR);

khối truyền dữ liệu có nhiệm vụ đóng gói các dữ liệu thông tin của tín hiệu thu về, dữ liệu đã được nén và hệ số bit tổn hao và gửi tới máy tính;

khối ghi lưu và giải nén dữ liệu thích nghi: thực hiện chức năng lưu trữ dữ liệu được truyền đến từ FGPA vào bộ nhớ trên máy tính và thực hiện chức năng giải nén dữ liệu; việc giải nén dữ liệu được thực hiện theo thuật toán FOR và dựa vào các thông tin đã được đóng gói trong dữ liệu.

2. Phương pháp nén và giải nén dữ liệu thích nghi tăng hiệu suất đường truyền dữ liệu trong hệ thống ra đa bao gồm:

bước 1: xử lý dữ liệu đầu vào;

đầu vào của hệ thống là dữ liệu của tín hiệu thu về bởi ra đa bao gồm dữ liệu mang thông tin của tín hiệu và dữ liệu được số hóa từ bộ ADC (dữ liệu ADC); dữ liệu ADC trước khi được nén sẽ được xử lý để lược bỏ phần giá trị các bit thấp, các bước thực hiện như sau:

giá trị dữ liệu là kiểu số nguyên có dấu, chuyển giá trị dữ liệu thành kiểu số nguyên không dấu như sau:

$$b = \begin{cases} 2a, & \text{nếu } a \text{ là số dương} \\ -2a - 1, & \text{nếu } a \text{ là số âm} \end{cases}$$

trong đó: a là dữ liệu ADC kiểu số nguyên có dấu, b là dữ liệu sau khi chuyển sang số nguyên không dấu;

sau khi chuyển sang giá trị số nguyên không dấu, các giá trị dương ban đầu được chuyển thành số chẵn, các giá trị âm ban đầu được chuyển thành số lẻ; bit thấp nhất của dữ liệu sau khi chuyển thể hiện số chẵn hoặc số lẻ;

bước 2: xác định giá trị hệ số bit tổn hao phục vụ cho việc nén dữ liệu thích nghi;

dữ liệu sau khi được xử lý sẽ được chia thành các khối dữ liệu, mỗi khối bao gồm N giá trị dữ liệu; giá trị hệ số bit tổn hao ứng với mỗi khối dữ liệu được nén được xác định dựa trên giá trị trung bình của khối dữ liệu trước đó theo nguyên lý giá trị trung bình càng nhỏ sẽ có hệ số bit tổn thất càng nhỏ và ngược lại; trong đó, việc thực hiện xác định giá trị của hệ số bit tổn hao bằng cách so sánh giá trị trung bình của khối dữ liệu với các ngưỡng giá trị; nếu giá trị trung bình nằm trong vùng ngưỡng giá trị nào thì hệ số bit tổn hao sẽ được gán một giá trị tương ứng với ngưỡng đó; việc đặt giá trị các ngưỡng so sánh và giá trị hệ số bit tổn hao tương ứng có thể thay đổi cài đặt tùy ý để đánh giá, đảm bảo yêu cầu về tỉ số nén và yêu cầu về tổn hao khi xử lý tín hiệu; theo nguyên lý này, việc nén dữ liệu có thể được thực hiện nén dữ liệu không tổn hao ở những vùng giá trị dữ liệu bé và thực hiện nén dữ liệu có tổn hao ở những vùng có giá trị dữ liệu lớn với tổn hao giá trị dữ liệu có thể hạn chế được;

bước 3: nén dữ liệu;

thực hiện nén dữ liệu theo thuật toán FOR và dựa vào giá trị hệ số bit tổn hao; việc thực hiện lần lượt bao gồm các bước:

lược bỏ các bit thấp của dữ liệu dựa trên hệ số bit tổn hao;

nén dữ liệu theo thuật toán FOR: thuật toán FOR sẽ nén dữ liệu theo từng khối dữ liệu, các bước thực hiện như sau:

thuật toán sẽ đi tìm giá trị lớn nhất (M) và nhỏ nhất (m) của khối dữ liệu đó;

tìm độ rộng bit tương ứng để biểu diễn dữ liệu của mỗi khối:

$$k = \lceil \log_2(M - m + 1) \rceil$$

trong đó $\lceil x \rceil$ là phép lấy giá trị nguyên nhỏ nhất lớn hơn x ;

nén dữ liệu mỗi khối bởi một trường tiêu đề thông tin (header) bao gồm giá trị nhỏ nhất (m) của khối dữ liệu và độ rộng bit k , theo đó là N giá trị dữ liệu có độ rộng k -bit biểu diễn hiệu của mỗi số trong khối với giá trị nhỏ nhất (m);

bước 4: đóng gói dữ liệu và truyền dữ liệu;

bước này tiến hành đóng gói dữ liệu thành các bản tin bao gồm dữ liệu mang thông tin của tín hiệu thu về bởi ra đa, dữ liệu sau khi nén và hệ số bit tổn hao; truyền dữ liệu thực hiện truyền các bản tin dữ liệu trên đường truyền *ethernet*;

bước 5: ghi lưu dữ liệu;

bước này tiến hành ghi lưu dữ liệu được truyền đến từ FPGA vào bộ nhớ của máy tính; dữ liệu được ghi lưu bao gồm dữ liệu mang thông tin của tín hiệu thu về bởi ra đa, dữ liệu nén và hệ số bit tổn hao;

bước 6: giải nén và khôi phục thông tin dữ liệu;

giải nén và khôi phục phần dữ liệu nén đã được ghi lưu theo thuật toán FOR và dựa trên giá trị hệ số bit tổn hao:

với mỗi khối dữ liệu nén, lấy thông tin về giá trị nhỏ nhất (m) và số bit k ;

dịch dữ liệu mỗi lần k bit và cộng giá trị đó với giá trị nhỏ nhất (m), từ đó có được giá trị dữ liệu được lược bỏ các bit thấp; tổng cộng cần thực hiện N lần dịch bit và cộng giá trị để giải nén mỗi khối dữ liệu;

khôi phục và hiệu chỉnh giá trị các điểm dữ liệu dựa trên thông tin của hệ số bit tổn hao;

số nguyên có dấu được khôi phục lại như sau:

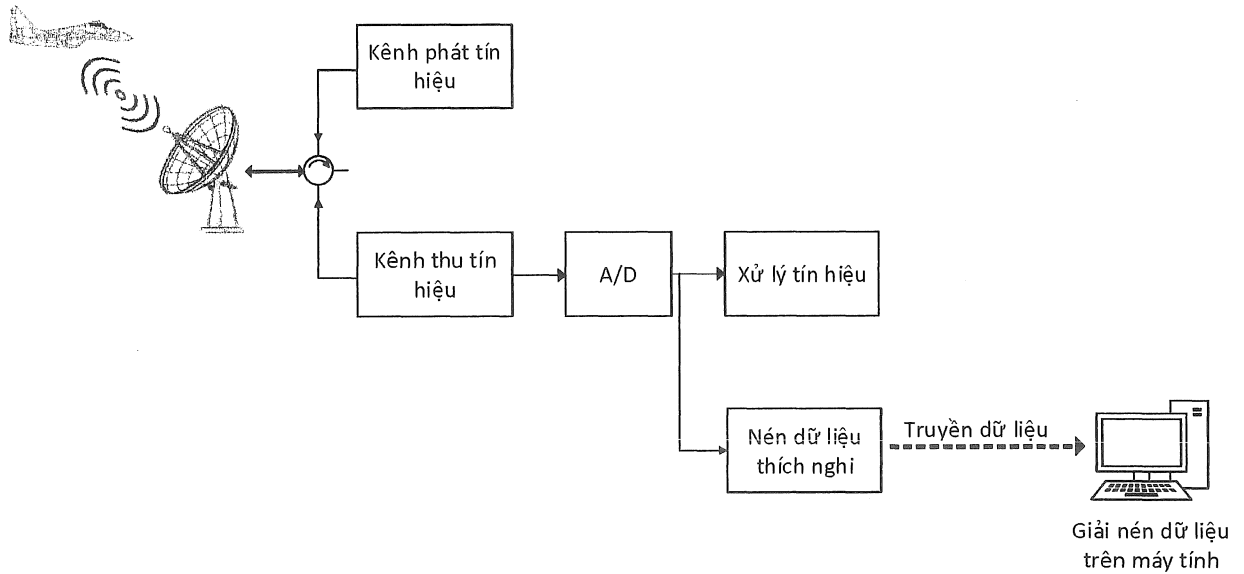
$$a' = \begin{cases} \frac{b'}{2} = \frac{b - \Delta}{2} = a - \frac{\Delta}{2}, & \text{nếu } b' \text{ là số chẵn} \\ \frac{b' + 1}{-2} = \frac{b - \Delta + 1}{-2} = a + \frac{\Delta}{2}, & \text{nếu } b' \text{ là số lẻ} \end{cases}$$

lúc này, sai số giữa giá trị nguyên có dấu ban đầu cần nén và sau khi giải nén là $(a - a') \in [-2^n - 1; 2^n - 1]$.

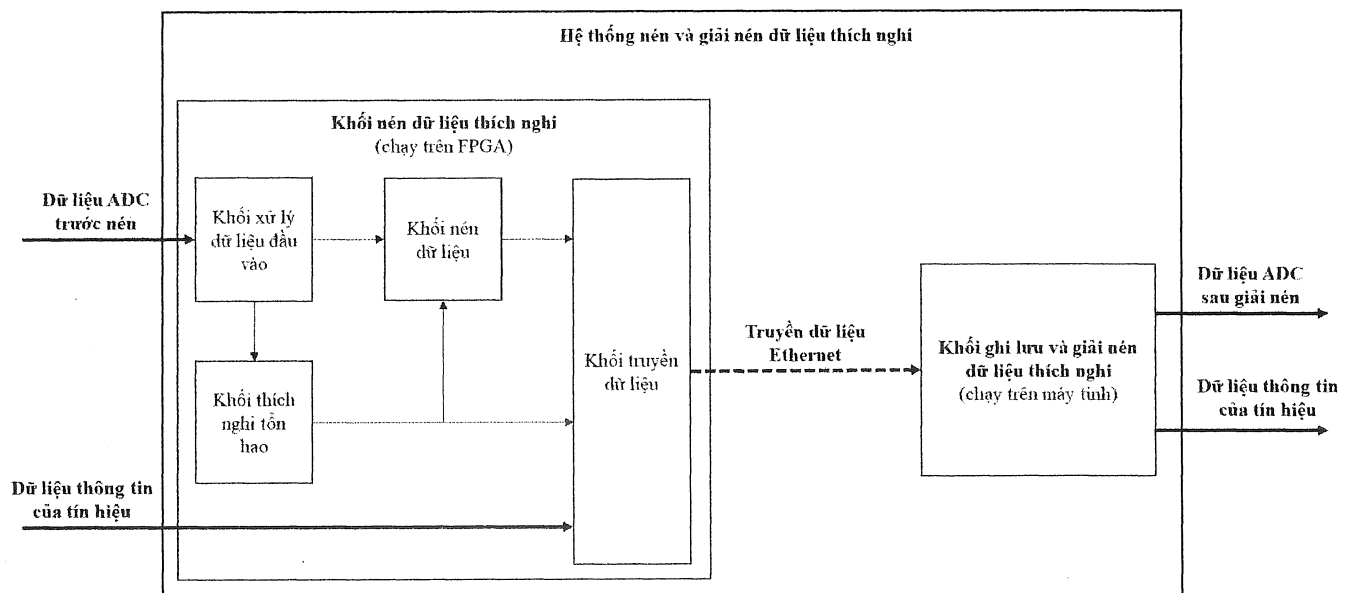
để giảm khoảng sai lệch giữa dữ liệu gốc ban đầu và dữ liệu sau giải nén, một giá trị offset = 2^{n-1} được thêm vào khi khôi phục dữ liệu như sau:

$$a' = \begin{cases} \frac{b'}{2} + 2^{n-1} = \frac{b - \Delta}{2} + 2^{n-1} = a - \frac{\Delta}{2} + 2^{n-1}, & \text{nếu } b' \text{ là số chẵn} \\ \frac{b' + 1}{-2} - 2^{n-1} = \frac{b - \Delta + 1}{-2} - 2^{n-1} = a + \frac{\Delta}{2} - 2^{n-1}, & \text{nếu } b' \text{ là số lẻ} \end{cases}$$

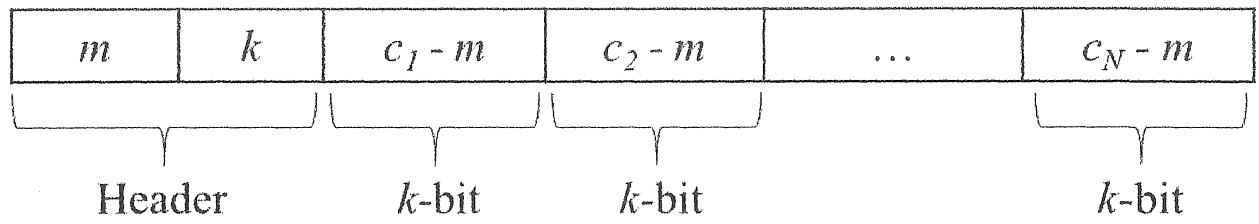
lúc này, sai số giữa giá trị nguyên có dấu ban đầu cần nén và sau khi giải nén là $(a - a') \in [-2^{n-1}; 2^{n-1}]$.



Hình 1



Hình 2



Hình 3