



(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)



1-0052780

(51)^{2022.01} G06F 12/02

(13) B

(21) 1-2023-04640

(22) 06/12/2021

(86) PCT/US2021/062064 06/12/2021

(87) WO 2022/159184 A1 28/07/2022

(30) 17/154,345 21/01/2021 US

(45) 27/10/2025 451

(43) 25/10/2023 427A

(73) QUALCOMM INCORPORATED (US)

ATTN: International IP Administration, 5775 Morehouse Drive, San Diego, CA
92121-1714, United States of America

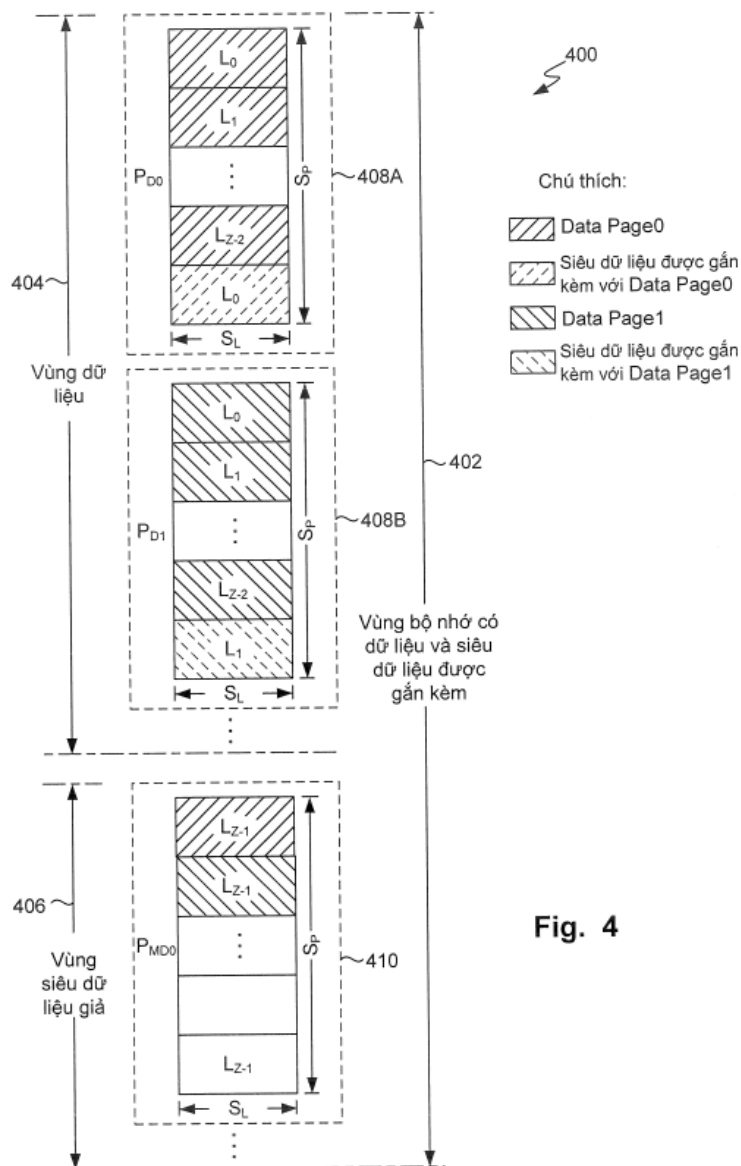
(72) TUMMALA, Gopi (IN); NANDU, Hiral (IN); PALACHARLA, Subbarao (US);
HASSAN, Syed Minhaj (PK); BHYRAVAJOSULA, Sai Ramesh (IN); NANNAKA,
Anurag (IN).

(74) Công ty TNHH Quốc tế D & N (D&N INTERNATIONAL CO.,LTD.)

(54) PHƯƠNG PHÁP, HỆ THỐNG VÀ PHƯƠNG TIỆN ĐỌC ĐƯỢC BẰNG MÁY
TÍNH ĐỂ THAY ĐỔI VỊ TRÍ SIÊU DỮ LIỆU

(21) 1-2023-04640

(57) Sáng chế nói chung đề cập đến phương pháp, hệ thống và phương tiện đọc được bằng máy tính để thay đổi vị trí siêu dữ liệu. Siêu dữ liệu được thay đổi vị trí động trong DRAM từ trang ảo không ánh xạ tới cùng hàng DRAM trong đó chứa dữ liệu mà nó gắn với, tới cùng hàng DRAM đó. Nếu đích của yêu cầu truy cập dữ liệu là vị trí trong trang thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu thay vì dữ liệu, thì vị trí thứ hai trong trang thứ hai có thể được xác định và dữ liệu được yêu cầu có thể được truy cập tại vị trí thứ hai. Siêu dữ liệu được gắn kèm có thể được truy cập tại vị trí trong trang thứ nhất, mà được tạo cấu hình trong miền ảo để lưu trữ dữ liệu nhưng được tạo cấu hình trong miền vật lý để lưu trữ siêu dữ liệu gắn với dữ liệu trong trang thứ nhất.



Lĩnh vực kỹ thuật được đề cập

Sáng chế nói chung đề cập đến kỹ thuật thay đổi vị trí siêu dữ liệu động trong bộ nhớ.

Tình trạng kỹ thuật của sáng chế

Thiết bị điện toán có thể bao gồm nhiều hệ thống con giao tiếp với nhau thông qua các bus hoặc các kết nối khác. Một thiết bị điện toán như vậy có thể là, ví dụ, thiết bị điện toán di động (portable computing device - PCD), chẳng hạn như máy tính xách tay, máy tính cầm tay, điện thoại di động hoặc điện thoại thông minh, thiết bị hỗ trợ kỹ thuật số cầm tay, máy chơi game cầm tay, v.v. Có thể bao gồm các hệ thống con giao tiếp trong cùng một chip mạch tích hợp hoặc trong các chip khác nhau. “Hệ thống trên chip” hay “SoC” (system-on-a-chip) là ví dụ về một chip như vậy tích hợp nhiều thành phần để cung cấp chức năng mức hệ thống. Ví dụ, SoC có thể bao gồm một hoặc nhiều loại bộ xử lý, chẳng hạn như bộ xử lý trung tâm (central processing unit - CPU), bộ xử lý đồ họa (graphics processing unit - GPU), bộ xử lý tín hiệu kỹ thuật số (digital signal processor - DSP) và bộ xử lý nơ-ron (neural processing unit - NPU). SoC có thể bao gồm các hệ thống con khác, chẳng hạn như bộ thu phát hoặc hệ thống con “mô-đem” cung cấp kết nối không dây, hệ thống con bộ nhớ, v.v. SoC có thể được ghép nối với một hoặc nhiều chip bộ nhớ thông qua liên kết truyền thông dữ liệu. Các công cụ xử lý khác nhau có thể thực hiện các giao dịch bộ nhớ với bộ nhớ. Bộ nhớ chính hoặc hệ thống trong các PCD và các thiết bị điện toán khác thường bao gồm bộ nhớ truy cập ngẫu nhiên động (dynamic random access memory - DRAM). DRAM được tổ chức theo mảng hàng và cột. Hàng phải được mở trước khi có thể truy cập dữ liệu của hàng đó. Chỉ có thể có một hàng được mở tại bất kỳ thời điểm nào. Hàng DRAM đôi khi được gọi là trang (vật lý).

Thuật ngữ “xung đột” đề cập đến giao dịch bộ nhớ hướng đến hàng DRAM bị đóng, trong khi cùng lúc đó một hàng khác (giao dịch không được hướng đến) đang mở.

Nếu giao dịch bộ nhớ xảy ra xung đột, thì đầu tiên hàng đang mở phải được đóng lại, sau đó phải mở hàng mà giao dịch hướng tới. Xung đột làm tăng thời gian trễ bộ nhớ, đây là khoảng thời gian để hoàn thành giao dịch bộ nhớ để đáp lại yêu cầu giao dịch của công cụ xử lý. Thời gian trễ bộ nhớ có khả năng ảnh hưởng đến hiệu suất của thiết bị điện toán và cuối cùng là trải nghiệm người dùng. Giảm thiểu thời gian trễ bộ nhớ cũng có thể là mục tiêu ý nghĩa trong các thiết bị điện toán thực hiện nhiệm vụ cơ yếu hoặc quan trọng về an toàn, chẳng hạn như những thiết bị được sử dụng để điều khiển phương tiện tự hành, thiết bị bay không người lái, v.v.

Một số loại dữ liệu có gắn với siêu dữ liệu. Ví dụ, "hội chứng" là siêu dữ liệu gắn với dữ liệu cần được xử lý bởi các thuật toán sửa lỗi. Các cờ và siêu dữ liệu khác có thể được gắn với dữ liệu cần được xử lý bởi thuật toán nén. Chữ ký, hàm băm, v.v., có thể được gắn với dữ liệu cần được xử lý bởi thuật toán xác thực. Vẫn còn những loại siêu dữ liệu khác được biết đến, chẳng hạn như tiện ích mở rộng gắn thẻ bộ nhớ (memory tagging extensions - MTE). Siêu dữ liệu thường được lưu trữ trong bộ nhớ theo cách mà nó có thể được truy cập khi dữ liệu mà nó gắn với được truy cập.

Bản chất kỹ thuật của sáng chế

Sáng chế đề cập đến các hệ thống, phương pháp, phương tiện đọc được bằng máy tính và các ví dụ khác cho việc thay đổi vị trí siêu dữ liệu trong DRAM. Siêu dữ liệu có thể được thay đổi vị trí từ một trang ảo không ánh xạ tới cùng hàng DRAM chứa dữ liệu mà nó gắn với, tới cùng hàng DRAM đó.

Phương pháp làm ví dụ để thay đổi vị trí siêu dữ liệu trong DRAM có thể bao gồm nhận yêu cầu thứ nhất để truy cập dữ liệu tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất của DRAM và siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu thứ nhất trong trang bộ nhớ thứ hai của DRAM. Phương pháp này có thể bao gồm việc xác định xem vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất có được tạo cấu hình để lưu trữ siêu dữ liệu hay không. Phương pháp này cũng có thể bao gồm việc truy cập dữ liệu tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu. Phương pháp này có thể còn bao gồm việc xác định vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu.

liệu. Phương pháp này vẫn có thể còn bao gồm việc truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu. Phương pháp này có thể bao gồm việc xác định vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu. Phương pháp này có thể còn bao gồm việc truy cập dữ liệu tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu. Phương pháp này có thể còn bao gồm việc truy cập siêu dữ liệu được gắn kèm tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu.

Hệ thống làm ví dụ để thay đổi vị trí siêu dữ liệu có thể bao gồm DRAM và bộ xử lý. Bộ xử lý có thể được tạo cấu hình để nhận yêu cầu thứ nhất để truy cập dữ liệu tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất và siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu thứ nhất trong trang bộ nhớ thứ hai. Bộ xử lý cũng có thể được tạo cấu hình để xác định xem vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất có được tạo cấu hình để lưu trữ siêu dữ liệu hay không. Bộ xử lý có thể còn được tạo cấu hình để truy cập dữ liệu tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu. Bộ xử lý vẫn có thể còn được tạo cấu hình để xác định vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu. Bộ xử lý có thể còn được tạo cấu hình để truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu. Bộ xử lý có thể được tạo cấu hình để xác định vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu. Bộ xử lý có thể còn được tạo cấu hình để truy cập dữ liệu tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu. Bộ xử lý có thể còn được tạo cấu hình để truy cập siêu dữ liệu được gắn kèm tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu.

Hệ thống khác làm ví dụ để thay đổi vị trí siêu dữ liệu trong DRAM có thể bao

gồm phương tiện nhận yêu cầu thứ nhất để truy cập dữ liệu tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất của DRAM và siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu thứ nhất trong trang bộ nhớ thứ hai của DRAM. Hệ thống có thể bao gồm phương tiện để xác định xem vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất có được tạo cấu hình để lưu trữ siêu dữ liệu hay không. Hệ thống cũng có thể bao gồm phương tiện để truy cập dữ liệu tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu. Hệ thống có thể còn bao gồm phương tiện để xác định vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu. Hệ thống vẫn có thể còn bao gồm phương tiện để truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu. Hệ thống có thể bao gồm phương tiện để xác định vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu. Hệ thống có thể còn bao gồm phương tiện để truy cập dữ liệu tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu. Hệ thống có thể còn bao gồm phương tiện để truy cập siêu dữ liệu được gắn kèm tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu.

Phương tiện đọc được bằng máy tính làm ví dụ để thay đổi vị trí siêu dữ liệu trong DRAM có thể bao gồm phương tiện bất biến đọc được bằng máy tính có các lệnh được lưu trữ trên đó ở dạng có thể thực thi được bằng máy tính. Các lệnh, khi được thực thi bởi bộ xử lý, có thể tạo cấu hình bộ xử lý để nhận yêu cầu thứ nhất để truy cập dữ liệu tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất và siêu dữ liệu được gắn kèm trong trang bộ nhớ thứ hai. Các lệnh có thể tạo cấu hình bộ xử lý để xác định xem vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất có được tạo cấu hình để lưu trữ siêu dữ liệu hay không. Các lệnh có thể còn tạo cấu hình bộ xử lý để truy cập dữ liệu tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu. Các lệnh có thể còn tạo cấu hình bộ xử lý để xác định vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ

nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu. Các lệnh có thể còn tạo cấu hình bộ xử lý để truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu. Các lệnh có thể tạo cấu hình bộ xử lý để xác định vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu. Các lệnh có thể còn tạo cấu hình bộ xử lý để truy cập dữ liệu tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu. Các lệnh có thể còn tạo cấu hình bộ xử lý để truy cập siêu dữ liệu được gắn kèm tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu.

Mô tả vắn tắt các hình vẽ

Trong các hình vẽ, các số chỉ dẫn giống nhau đề cập đến các phần giống nhau trong suốt các hình vẽ khác nhau trừ khi được chỉ ra khác. Đối với các số chỉ dẫn có ký hiệu ký tự chữ cái, chẳng hạn như “102A” hoặc “102B”, ký hiệu ký tự chữ cái có thể phân biệt hai phần hoặc thành phần giống nhau có trong cùng một hình. Các ký hiệu ký tự chữ cái cho các số chỉ dẫn có thể được bỏ qua khi có ý định rằng một số chỉ dẫn bao gồm tất cả các phần có cùng số chỉ dẫn trong tất cả các hình.

Fig.1 là sơ đồ khối minh họa về mặt khái niệm việc thay đổi vị trí siêu dữ liệu trong bộ nhớ truy cập ngẫu nhiên động (dynamic random access memory - DRAM) theo các phương án làm ví dụ.

Fig.2 là sơ đồ khối của hệ thống thay đổi vị trí siêu dữ liệu trong DRAM, theo các phương án làm ví dụ.

Fig.3 là bản đồ bộ nhớ minh họa các vị trí thông thường của dữ liệu và siêu dữ liệu được gắn kèm trong DRAM.

Fig.4 là bản đồ bộ nhớ minh họa việc thay đổi vị trí siêu dữ liệu trong DRAM, theo các phương án làm ví dụ.

Fig.5 là bản đồ bộ nhớ minh họa các vị trí của dữ liệu không có siêu dữ liệu được gắn kèm, theo các phương án làm ví dụ.

Fig.6 là lưu đồ minh họa phương pháp thay đổi vị trí siêu dữ liệu trong DRAM, theo các phương án làm ví dụ.

Fig.7 là sơ đồ địa chỉ minh họa ví dụ về dịch địa chỉ dữ liệu trong phương pháp thay đổi vị trí trên Fig.6, theo các phương án làm ví dụ.

Fig.8 là bản đồ bộ nhớ minh họa các khía cạnh của việc dịch trên Fig.7.

Fig.9 tương tự như Fig.8, hiển thị các ví dụ về địa chỉ bộ nhớ thực tế.

Fig.10 là sơ đồ địa chỉ minh họa ví dụ về dịch địa chỉ siêu dữ liệu trong phương pháp thay đổi vị trí trên Fig.6, theo các phương án làm ví dụ.

Fig.11 là bản đồ bộ nhớ minh họa việc thay đổi vị trí siêu dữ liệu trong DRAM, theo các phương án làm ví dụ trong đó DRAM bao gồm nhiều nhóm dãy.

Fig.12 là bản đồ bộ nhớ khác minh họa việc thay đổi vị trí siêu dữ liệu trong DRAM, theo các phương án làm ví dụ trong đó DRAM bao gồm nhiều nhóm dãy.

Fig.13 là sơ đồ khối của thiết bị điện toán, theo các phương án làm ví dụ.

Mô tả chi tiết sáng chế

Từ “ví dụ” được sử dụng ở đây có nghĩa là “làm ví dụ, ví dụ hoặc minh họa.” Từ “minh họa” có thể được sử dụng ở đây đồng nghĩa với “ví dụ”. Khía cạnh bất kỳ được mô tả ở đây dưới dạng “ví dụ” không nhất thiết phải được hiểu là ưu tiên hoặc có lợi hơn các khía cạnh khác.

Như được minh họa ở dạng khái niệm trên Fig.1, trong hệ thống 100, máy khách 102 có thể đưa ra các yêu cầu giao dịch bộ nhớ hướng đến bộ nhớ truy cập ngẫu nhiên động (“DRAM”) 104. Mặc dù không được thể hiện nhằm mục đích rõ ràng, hệ thống 100 có thể được bao gồm trong bất kỳ loại thiết bị điện toán nào. Máy khách 102 cũng có thể thuộc bất kỳ loại nào. Theo hiểu biết của người có hiểu biết trung bình trong lĩnh vực này, máy khách 102 có thể được thể hiện bằng việc thực thi ứng dụng, luồng, v.v., trên bộ xử lý. Máy khách 102 cũng có thể được xem là công cụ xử lý hoặc trạm chủ bus. Theo hiểu biết của người có hiểu biết trung bình về lĩnh vực này, trạm chủ bus là bất kỳ thiết bị nào khởi tạo giao dịch bus với tài nguyên hệ thống, trong đó DRAM 104 là một ví dụ. DRAM 104 có thể thuộc bất kỳ loại nào, chẳng hạn như DRAM đồng bộ tốc độ

dữ liệu kép (double data rate synchronous DRAM - “DDR-SDRAM”), đôi khi được gọi ngắn gọn là “DDR”. Khi công nghệ DDR phát triển, các phiên bản DDR như DDR công suất thấp thế hệ thứ tư (fourth generation low-power DDR - “LPDDR4”) và DDR công suất thấp thế hệ thứ năm (fifth generation low-power DDR - “LPDDR5”) đã được phát triển. Ví dụ, DRAM 104 có thể là LPDDR4, LPDDR5, v.v.

Máy khách 102 có thể đưa ra các yêu cầu giao dịch bộ nhớ khác nhau mà có thể bao gồm các yêu cầu đọc dữ liệu từ DRAM 104 và yêu cầu ghi dữ liệu vào (tức là, lưu trữ dữ liệu vào) DRAM 104. Các yêu cầu giao dịch bộ nhớ cũng có thể được gọi là yêu cầu truy cập DRAM 104. Mỗi yêu cầu giao dịch bộ nhớ có thể bao gồm địa chỉ hoặc vị trí ảo. Giao dịch đọc được hoàn thành khi dữ liệu là đối tượng của yêu cầu giao dịch đọc đã được trả lại cho máy khách 102. Giao dịch ghi được hoàn thành khi dữ liệu là đối tượng của yêu cầu giao dịch ghi đã được ghi vào DRAM 104.

Một số loại dữ liệu có thể được lưu trữ trong DRAM 104 có siêu dữ liệu gắn với dữ liệu đó. Bộ xử lý siêu dữ liệu 106 có thể nhận các yêu cầu giao dịch bộ nhớ từ máy khách 102 và xác định xem dữ liệu có phải là đối tượng của yêu cầu giao dịch có siêu dữ liệu gắn với hay không. Bộ xử lý siêu dữ liệu 106 có thể xác định xem dữ liệu có siêu dữ liệu gắn kèm hay không dựa vào địa chỉ ảo. Ví dụ, bộ xử lý siêu dữ liệu 106 có thể xác định rằng dữ liệu có siêu dữ liệu gắn kèm nếu địa chỉ ảo nằm trong dải địa chỉ thứ nhất và không có siêu dữ liệu gắn kèm nếu địa chỉ ảo nằm trong dải địa chỉ thứ hai. Các dải địa chỉ khác nhau có thể tương ứng với các loại siêu dữ liệu khác nhau. Tùy thuộc vào việc dữ liệu có siêu dữ liệu gắn kèm hay không có siêu dữ liệu gắn kèm và vào loại siêu dữ liệu, bộ xử lý siêu dữ liệu 106 có thể xử lý dữ liệu tương ứng. Ví dụ, nếu địa chỉ ảo tương ứng với dữ liệu thuộc loại tuân theo đặc điểm sửa lỗi, bộ xử lý siêu dữ liệu 106 có thể, trong trường hợp truy cập ghi, tạo siêu dữ liệu hội chứng được lưu trữ trong DRAM 104 cùng với dữ liệu hoặc trong trường hợp truy cập đọc, có thể thực hiện thuật toán sửa lỗi trên dữ liệu bằng cách sử dụng siêu dữ liệu hội chứng liên quan được đọc từ DRAM 104. Một vấn đề được giải quyết bằng các phương án được mô tả trong sáng chế là cách để truy cập siêu dữ liệu cùng với dữ liệu một cách hiệu quả.

Dữ liệu và siêu dữ liệu được gắn kèm có thể được lưu trữ trong vùng thứ nhất (nghĩa là, dải địa chỉ) 108 của DRAM 104. Mũi tên nét đứt 110 biểu thị về mặt khái

niệm mối liên hệ giữa dòng dữ liệu làm ví dụ và siêu dữ liệu của nó. Dữ liệu thuộc loại không có bất kỳ siêu dữ liệu gắn kèm nào có thể được lưu trữ trong vùng thứ hai 112 của DRAM 104. Mặc dù chỉ có một vùng “thứ nhất” 108 như vậy và một vùng “thứ hai” 112 như vậy được thể hiện trên Fig.1 cho mục đích ví dụ, có thể có nhiều hơn một vùng trong đó dữ liệu và siêu dữ liệu gắn kèm được lưu trữ, và nhiều hơn một vùng trong đó dữ liệu không có bất kỳ siêu dữ liệu được gắn kèm nào được lưu trữ. Mặc dù không được thể hiện trên FIG.1, các địa chỉ ảo (trong miền ảo) được cung cấp bởi các yêu cầu giao dịch bộ nhớ dịch hoặc ánh xạ tới các địa chỉ vật lý (trong miền vật lý) trong DRAM 104.

Cần hiểu rằng các vị trí dữ liệu và vị trí siêu dữ liệu trong miền ảo là nơi mà, như đôi khi được diễn tả một cách thông thường, máy khách 102 “thấy” hoặc “nhận thức” dữ liệu và siêu dữ liệu được lưu trữ. Như được mô tả chi tiết hơn bên dưới, từ góc độ máy khách (nghĩa là trong miền ảo), các phần của DRAM 104 có thể được tạo cấu hình để lưu trữ dữ liệu, trong khi các phần khác của DRAM 104 có thể được tạo cấu hình để lưu trữ siêu dữ liệu. Nói cách khác, miền ảo hoặc miền máy khách liên quan đến sơ đồ địa chỉ được máy khách 102 sử dụng để tham chiếu gián tiếp các vị trí lưu trữ dữ liệu và siêu dữ liệu trong DRAM 104. Ngược lại, miền vật lý liên quan đến sơ đồ địa chỉ được sử dụng trong DRAM 104 để truy cập vật lý các vị trí lưu trữ trong đó dữ liệu và siêu dữ liệu có thể được lưu trữ. Tương ứng, có thể tuyên bố rằng trong miền vật lý, các phần của DRAM 104 có thể được tạo cấu hình để lưu trữ dữ liệu, trong khi các phần khác của DRAM 104 có thể được tạo cấu hình để lưu trữ siêu dữ liệu.

Như được mô tả trên Fig.2, hệ thống 200 có thể bao gồm máy khách 202, bộ nhớ đệm hệ thống 204, bộ đồng xử lý 206, bộ điều khiển bộ nhớ 208 và DRAM 210. Hệ thống 200 có thể là ví dụ về hệ thống 100 được mô tả ở trên (Fig.1). Máy khách 202, bộ đồng xử lý 206 và DRAM 210 có thể là các ví dụ tương ứng về máy khách 102, bộ xử lý siêu dữ liệu 106 và DRAM 104 được mô tả ở trên. Bộ nhớ đệm hệ thống 204 có thể được xen vào luồng dữ liệu giữa máy khách 202 và bộ đồng xử lý 206. Bộ nhớ đệm hệ thống 204 có thể hoạt động theo cách thông thường, như được hiểu bởi người có hiểu biết trung bình trong lĩnh vực này. Theo đó, một số yêu cầu giao dịch bộ nhớ có thể được hoàn thành bằng cách sử dụng bộ nhớ đệm hệ thống 204, loại bỏ nhu cầu truy cập DRAM 210. Bộ nhớ đệm hệ thống 204 có thể tổ chức dữ liệu theo các đơn vị được gọi

là dòng bộ nhớ đệm hoặc ngắn gọn là “dòng”. Hoàn thành các yêu cầu giao dịch bộ nhớ bằng cách sử dụng bộ nhớ đệm hệ thống 204 thay vì DRAM 210 không liên quan trực tiếp đến sáng chế. Thay vào đó, sáng chế đề cập đến các yêu cầu giao dịch bộ nhớ được hoàn thành bằng cách truy cập DRAM 210. Tuy nhiên, nên hiểu rằng máy khách 202, bộ nhớ đệm hệ thống 204 và bộ đồng xử lý 206 đều xử lý dữ liệu trong miền ảo và do đó xử lý dữ liệu theo địa chỉ trang ảo (tức là số trang) và địa chỉ dòng ảo (tức là số dòng) thay vì địa chỉ vật lý trong DRAM 210. Máy khách 202 đưa ra các yêu cầu giao dịch bộ nhớ mà cho biết địa chỉ ảo theo số trang ảo và theo số dòng trong trang ảo.

Bộ đồng xử lý 206 có thể, bên cạnh các chức năng khác, xử lý siêu dữ liệu, chẳng hạn như một hoặc nhiều trong số, ví dụ, sửa lỗi, gắn thẻ bộ nhớ, xác thực DRAM, làm mới DRAM, nén DRAM, v.v. Bộ đồng xử lý 206 có thể xác định liệu máy khách có đang yêu cầu truy cập dữ liệu thuộc loại có siêu dữ liệu được gắn kèm hay không. Yêu cầu truy cập dữ liệu có siêu dữ liệu được gắn kèm có thể sinh ra yêu cầu truy cập siêu dữ liệu được gắn kèm. Do đó, bộ đồng xử lý 206 có thể cung cấp cho bộ điều khiển bộ nhớ 208 yêu cầu không chỉ để truy cập dữ liệu do máy khách 202 yêu cầu mà còn để truy cập siêu dữ liệu được gắn kèm. Ví dụ, để đáp lại yêu cầu từ máy khách 202 ghi dữ liệu thuộc loại tuân theo sửa lỗi vào DRAM 210, bộ đồng xử lý 206 có thể tạo siêu dữ liệu hội chứng và yêu cầu tới bộ điều khiển bộ nhớ 208 để lưu trữ siêu dữ liệu hội chứng trong DRAM 210 cùng với dữ liệu. Tương tự, để đáp lại yêu cầu đọc loại dữ liệu tuân theo sửa lỗi từ DRAM 210, bộ đồng xử lý 206 có thể tạo yêu cầu tới bộ điều khiển bộ nhớ 208 để đọc siêu dữ liệu hội chứng có liên quan từ DRAM 210. Sau đó, bộ đồng xử lý 206 có thể thực hiện thuật toán sửa lỗi trên dữ liệu bằng cách sử dụng siêu dữ liệu hội chứng được đọc từ DRAM 210.

Bộ điều khiển bộ nhớ 208 có thể, bên cạnh các chức năng khác, dịch các yêu cầu giao dịch bộ nhớ thành các giao dịch DRAM, bao gồm các lệnh DRAM và địa chỉ vật lý, theo cách thông thường. Vì chức năng thông thường này của bộ điều khiển bộ nhớ 208 và các chức năng liên quan được hiểu rõ bởi người có hiểu biết trung bình trong lĩnh vực kỹ thuật, nên các khía cạnh như vậy của bộ điều khiển bộ nhớ 208 không được mô tả ở đây.

DRAM 210 có thể có cấu trúc thông thường và hoạt động theo cách thông

thường. Ví dụ, DRAM 210 có thể là LPDDR4, LPDDR5, v.v. Mặc dù cấu trúc và hoạt động của DRAM như vậy được hiểu rõ bởi người có hiểu biết trung bình trong lĩnh vực này, phần mô tả ngắn gọn sau đây được cung cấp làm thông tin cơ bản.

DRAM 210 có thể bao gồm một hoặc nhiều dãy 212. Ví dụ, DRAM 210 có thể có M dãy 212, có thể được gọi là “Dãy_0” đến “Dãy_M-1”. Mỗi dãy 212 có thể bao gồm mảng hai chiều 213 các vị trí lưu trữ, trong đó các vị trí lưu trữ trong mảng được truy cập bằng cách chọn các hàng và cột. Mảng 213 được thể hiện theo cách khái niệm trên Fig.2 cho mục đích rõ ràng; mảng 213 có thể có nhiều hàng và cột hơn được thể hiện và mỗi vị trí lưu trữ có thể được tạo cấu hình để lưu trữ một số lượng dữ liệu, chẳng hạn như một byte (byte - “B”). Hàng và cột ví dụ được đánh dấu (ở dạng gạch chéo) trên Fig.2. Do đó, địa chỉ vật lý DRAM do bộ điều khiển bộ nhớ 208 cung cấp có thể bao gồm địa chỉ hàng, địa chỉ cột và địa chỉ dãy. Để đáp ứng địa chỉ dãy, logic giải mã địa chỉ dãy (không được thể hiện) có thể chọn một trong các dãy 212. Để đáp ứng địa chỉ hàng, bộ giải mã địa chỉ hàng 214 có thể chọn một trong các hàng trong dãy đã chọn 212. Tương tự, để đáp lại địa chỉ cột, bộ giải mã địa chỉ cột 216 có thể chọn một trong các cột trong dãy đã chọn 212. Chốt đọc 218 có thể đệm dữ liệu đọc và chốt ghi 220 để đệm dữ liệu ghi. Logic đầu vào/đầu ra (Input/output - “I/O”) 222 có thể hướng việc đọc và ghi dữ liệu từ và đến các vị trí bộ nhớ đã chọn.

Mỗi dãy 212 có thể có bộ đệm hàng 224. Bộ đệm hàng 224 lưu trữ các nội dung của hàng được lựa chọn (đôi khi được gọi là "trang" vật lý). Hàng phải được kích hoạt hoặc "được mở" trước khi nó có thể được ghi vào hoặc được đọc từ đó. Sau khi hàng được mở, DRAM 210 có thể đọc hoặc ghi vào bất kỳ số lượng cột nào trong bộ đệm hàng 224 để đáp lại các lệnh đọc hoặc ghi. Theo lệnh đọc hoặc ghi, dữ liệu được truyền nối tiếp giữa bộ điều khiển bộ nhớ 208 và DRAM 210 theo đơn vị được gọi là “cụm”, bao gồm một số cột dữ liệu được xác định trước. Ví dụ, kích thước cụm có thể giống với kích thước dòng được mô tả ở trên. Hàng phải được khôi phục hoặc "được đóng" sau khi ghi vào hoặc đọc từ bộ đệm hàng 224. Chỉ có một hàng trong mỗi dãy 212 có thể vẫn mở tại một thời điểm.

Bộ điều khiển bộ nhớ 208 có thể nhận dạng mỗi giao dịch là "trúng", "không trúng" hoặc "xung đột" trên cơ sở trạng thái hiện hành của các dãy DRAM 212. Giao

dịch trúng (hit) là giao dịch đọc hoặc ghi vào hàng (trong dãy 212) mà bộ điều khiển bộ nhớ 208 xác định là đã mở tại thời điểm bộ điều khiển bộ nhớ 208 xử lý yêu cầu giao dịch. Giao dịch trúng có độ trễ thấp. Nghĩa là, giao dịch trúng có thể được hoàn thành trong khoảng thời gian tương đối ngắn. Giao dịch không trúng (miss) là giao dịch đọc hoặc ghi vào hàng (trong dãy 212) mà bộ điều khiển bộ nhớ 208 xác định là đóng tại thời điểm bộ điều khiển bộ nhớ 208 xử lý yêu cầu giao dịch. Nếu bộ điều khiển bộ nhớ 208 nhận dạng giao dịch đọc hoặc ghi là giao dịch không trúng, thì bộ điều khiển bộ nhớ 208 trước tiên phải mở hàng trước khi đọc từ hoặc ghi vào hàng đó. Giao dịch không trúng có độ trễ cao hơn giao dịch trúng. Nghĩa là, giao dịch không trúng mất nhiều thời gian hơn để hoàn thành so với giao dịch trúng. Giao dịch xung đột (conflict) là giao dịch đọc hoặc ghi vào hàng (trong dãy 212) mà bộ điều khiển bộ nhớ 208 xác định là đóng tại thời điểm bộ điều khiển bộ nhớ 208 xử lý yêu cầu giao dịch, trong khi bộ điều khiển bộ nhớ 208 xác định rằng một hàng khác ở cùng dãy (mà giao dịch bộ nhớ không được hướng đến) đang mở tại thời điểm đó. Nếu bộ điều khiển bộ nhớ 208 nhận dạng giao dịch đọc hoặc ghi là giao dịch xung đột, thì bộ điều khiển bộ nhớ 208 trước tiên đóng hàng mở trong dãy 212, sau đó mở hàng mà giao dịch được hướng đến trước khi đọc từ hoặc ghi vào hàng đó. Giao dịch xung đột có độ trễ cao hơn giao dịch không trúng. Nghĩa là, giao dịch xung đột mất nhiều thời gian hơn để hoàn thành so với giao dịch không trúng hoặc trúng.

Nếu dữ liệu và siêu dữ liệu được gắn với nó không được lưu trữ trong cùng hàng trong DRAM 210, thì việc truy cập bộ nhớ có thể dẫn đến xung đột và do đó có độ trễ cao. Các phương án làm ví dụ về hệ thống, phương pháp, phương tiện đọc được bằng máy tính, v.v., được mô tả trong sáng chế liên quan đến ánh xạ động hoặc thay đổi vị trí siêu dữ liệu theo cách tối đa hóa khả năng truy cập bộ nhớ của dữ liệu và siêu dữ liệu được gắn kèm sẽ không dẫn đến xung đột. Bằng cách lưu trữ dữ liệu và siêu dữ liệu được gắn với nó trong cùng hàng DRAM, dữ liệu và siêu dữ liệu có thể được truy cập đồng thời khi hàng DRAM được mở, do đó tránh được xung đột.

Như được mô tả trên Fig.3, sơ đồ địa chỉ thông thường hoặc cấu hình 300 có thể tạo cấu hình dải địa chỉ bộ nhớ hoặc vùng 302 để bao gồm vùng dữ liệu 304 và vùng siêu dữ liệu 306. Nghĩa là, vùng 302 được tạo cấu hình để lưu trữ dữ liệu trong vùng dữ liệu 304 và siêu dữ liệu được gắn với nó trong vùng siêu dữ liệu 306. Vùng dữ liệu 304

có thể được tạo cấu hình là hai hoặc nhiều trang dữ liệu 308A (“P_{D0}”), 308B (“P_{D1}”), v.v., thông qua trang dữ liệu cuối cùng (không được thể hiện). (Ký hiệu dấu ba chấm (“...”) theo sau trang dữ liệu 308B cho biết rằng có thể có bất kỳ số lượng trang dữ liệu nào khác.) Tương tự, vùng siêu dữ liệu 306 có thể được tạo cấu hình là một hoặc nhiều trang siêu dữ liệu 310.

Mỗi trang dữ liệu 308A, 308B, v.v., có thể được tạo cấu hình thành số (“Z”) các dòng dữ liệu: dòng dữ liệu thứ nhất L₀, dòng dữ liệu thứ hai L₁, v.v., đến dòng dữ liệu cuối cùng L_{Z-1}. Tương tự, mỗi trang trong số một hoặc nhiều trang siêu dữ liệu 310 có thể được tạo cấu hình thành Z dòng siêu dữ liệu: dòng siêu dữ liệu thứ nhất L₀, dòng siêu dữ liệu thứ hai L₁, v.v., đến dòng siêu dữ liệu cuối cùng L_{Z-1}. Tất cả siêu dữ liệu được gắn với dữ liệu được lưu trữ trong trang dữ liệu thứ nhất 308A có thể được lưu trữ trong dòng siêu dữ liệu thứ nhất L₀ của trang siêu dữ liệu 310 (“P_{MD0}”). Tương tự, tất cả siêu dữ liệu gắn với dữ liệu được lưu trữ trong trang dữ liệu thứ hai 308B có thể được lưu trữ trong dòng siêu dữ liệu thứ hai L₁ của trang siêu dữ liệu 310. Các vị trí tương đối của dữ liệu và siêu dữ liệu được gắn kèm như mô tả ở trên đối với hai trang dữ liệu thứ nhất 308A và 308B áp dụng cho tất cả các trang dữ liệu tiếp theo (không được thể hiện), và do đó, siêu dữ liệu gắn với dữ liệu được lưu trữ trong trang dữ liệu cuối cùng (không được thể hiện) có thể được lưu trữ ở dòng cuối cùng L_{Z-1} của trang siêu dữ liệu 310. Mỗi trang dữ liệu 308A, 308B, v.v. và mỗi trang siêu dữ liệu 310 có thể có kích thước trang (“S_P”) và kích thước dòng (“S_L”), ví dụ về các kích thước này được mô tả bên dưới. Mỗi trang trong miền ảo có thể được dịch sang (ví dụ, bởi bộ điều khiển bộ nhớ 208) hàng DRAM khác trong miền vật lý. Mỗi dòng của trang trong miền ảo có thể được dịch sang nhóm gồm một hoặc nhiều cột DRAM trong hàng tương ứng với trang đó.

Để truy cập dữ liệu và siêu dữ liệu gắn với nó, máy khách có thể đưa ra các yêu cầu giao dịch bao gồm các địa chỉ ảo cho biết đích của yêu cầu theo số trang và số dòng. Nghĩa là, máy khách có thể sử dụng sơ đồ địa chỉ miền ảo để xử lý dữ liệu trong các trang dữ liệu 308A, 308B, v.v. và xử lý siêu dữ liệu được gắn kèm trong các trang siêu dữ liệu 310. Trong cấu hình làm ví dụ 300, mỗi trang trong số các trang dữ liệu 308A, 308B, v.v. và các trang siêu dữ liệu 310 ánh xạ hoặc dịch trong miền vật lý sang hàng DRAM khác. Cần phải hiểu từ cấu hình làm ví dụ 300, rằng vì dữ liệu và siêu dữ liệu gắn với nó không được lưu trữ trong cùng hàng DRAM, truy cập bộ nhớ có thể dẫn đến

xung đột.

Như được mô tả trên Fig.4, sơ đồ địa chỉ làm ví dụ hoặc cấu hình 400 có thể tạo cấu hình dải địa chỉ hoặc vùng 402 để bao gồm vùng dữ liệu 404 và vùng siêu dữ liệu giả 406. Để truy cập dữ liệu và siêu dữ liệu gắn với nó, máy khách có thể đưa ra các yêu cầu giao dịch bao gồm các địa chỉ ảo trong vùng 402 nhận dạng đích của yêu cầu theo số trang và số dòng, theo cách tương tự được mô tả ở trên đối với Fig.3. Nghĩa là, máy khách có thể sử dụng sơ đồ địa chỉ miền ảo để xử lý dữ liệu trong các trang dữ liệu 408A, 408B, v.v. của vùng dữ liệu 404 và xử lý siêu dữ liệu được gắn kèm trong các trang siêu dữ liệu 410 của vùng siêu dữ liệu giả 406. Trong cấu hình làm ví dụ 400, mỗi trang trong số các trang dữ liệu 408A, 408B, v.v. và các trang siêu dữ liệu 410 ánh xạ hoặc dịch trong miền vật lý sang hàng DRAM khác. Tuy nhiên, trái ngược với vùng siêu dữ liệu 306 trong cấu hình thông thường 300 được mô tả ở trên (Fig.3), vùng siêu dữ liệu giả 406 được tạo cấu hình trong miền ảo (tức là máy khách) dưới dạng vùng siêu dữ liệu, nhưng như được trình bày trên Fig.4 được tạo cấu hình trong miền vật lý (tức là bộ điều khiển bộ nhớ và DRAM) dưới dạng vùng dữ liệu bổ sung, để lưu trữ dữ liệu đã được "thay đổi vị trí" khỏi vùng dữ liệu 404. Nói cách khác, máy khách 202 "thấy" hoặc "nhận thức" siêu dữ liệu như được lưu trữ trong vùng siêu dữ liệu giả 406, theo cách tương tự được mô tả ở trên đối với Fig.3, mặc dù siêu dữ liệu không thực sự (trong miền vật lý) được lưu trữ trong các hàng DRAM ánh xạ tới vùng siêu dữ liệu giả 406.

Thay vào đó, trong miền vật lý, siêu dữ liệu có thể được lưu trữ trong dòng được xác định trước, chẳng hạn như dòng cuối cùng của mỗi trang dữ liệu 408A, 408B, v.v., trong vùng dữ liệu 404. Nói cách khác, các phương án làm ví dụ được mô tả trong sáng chế hoạt động trong miền vật lý như thể dòng cuối cùng của trang dữ liệu được hoán đổi với dòng của trang siêu dữ liệu chứa siêu dữ liệu cho trang dữ liệu đó trong miền ảo. Ví dụ, trong miền vật lý, siêu dữ liệu được gắn với dữ liệu được lưu trữ từ dòng thứ nhất đến dòng áp chót (nghĩa là dòng dữ liệu L_0 đến L_{Z-2}) của trang dữ liệu thứ nhất 408A có thể được lưu trữ ở dòng cuối cùng của trang dữ liệu thứ nhất 408A. Ví dụ, trong miền vật lý, siêu dữ liệu gắn với dữ liệu được lưu trữ từ dòng thứ nhất đến dòng áp chót (nghĩa là dòng dữ liệu L_0 đến L_{Z-2}) của trang dữ liệu thứ hai 408B có thể được lưu trữ ở dòng cuối cùng của trang dữ liệu thứ hai 408B. Cùng các vị trí tương đối của dữ liệu và siêu dữ liệu được gắn kèm như mô tả ở trên đối với hai trang dữ liệu thứ nhất 408A và 408B

áp dụng cho tất cả các trang dữ liệu khác (không được thể hiện) và do đó, trong miền vật lý, siêu dữ liệu được gắn với dữ liệu được lưu trữ trong dòng thứ nhất đến dòng áp chót (nghĩa là dòng dữ liệu L_0 đến L_{Z-2}) của trang dữ liệu cuối cùng (không được thể hiện) có thể được lưu trữ ở dòng cuối cùng của trang dữ liệu cuối cùng đó.

Vùng siêu dữ liệu giả 406 có thể được tạo cấu hình trong miền vật lý dưới dạng một hoặc nhiều trang dữ liệu bổ sung 410 để lưu trữ, thay vì siêu dữ liệu, dữ liệu trong miền ảo được lưu trữ trong (nghĩa là máy khách “thấy” hoặc “nhận thức” như đang được lưu trữ trong) dòng cuối cùng (nghĩa là dòng dữ liệu L_{Z-1}) của mỗi trang dữ liệu 408A, 408B, v.v. Ví dụ, dòng thứ nhất của trang dữ liệu bổ sung 410 có thể được tạo cấu hình trong miền vật lý để lưu trữ dữ liệu có liên quan trong miền ảo với dòng cuối cùng của trang dữ liệu thứ nhất 408A, dòng thứ hai của trang dữ liệu bổ sung 410 có thể được tạo cấu hình để lưu trữ dữ liệu có liên quan trong miền ảo với dòng cuối cùng của trang dữ liệu thứ hai 408B, v.v..

Tóm lại, trong miền ảo hoặc từ góc nhìn của máy khách 202 (Fig.2), dòng cuối cùng của mỗi trang dữ liệu 408A, 408B, v.v., được tạo cấu hình để lưu trữ dữ liệu, nhưng trong miền vật lý, hoặc từ góc độ của bộ điều khiển bộ nhớ 208 và DRAM 210 (Fig.2), dòng cuối cùng của mỗi trang dữ liệu 408A, 408B, v.v., được tạo cấu hình để lưu trữ tất cả siêu dữ liệu gắn với một trong các trang 408A, 408B, v.v.. Hơn nữa, trong miền ảo hoặc từ góc độ của máy khách 202 (Fig.2), vùng siêu dữ liệu giả 406 được tạo cấu hình để lưu trữ siêu dữ liệu gắn với các trang dữ liệu 408A, 408B, v.v., nhưng trong miền vật lý hoặc từ góc độ của bộ điều khiển bộ nhớ 208 và DRAM 210, vùng siêu dữ liệu giả 406 được tạo cấu hình để lưu trữ dữ liệu, từ góc độ của máy khách 202 (Fig.2), được lưu trữ trong dòng cuối cùng của mỗi trang dữ liệu 408A, 408B, v.v.. Nói cách khác, dòng cuối cùng của mỗi trang dữ liệu 408A, 408B, v.v., là vị trí dữ liệu miền ảo đồng thời là vị trí siêu dữ liệu miền vật lý được tạo cấu hình để lưu trữ tất cả siêu dữ liệu gắn với một trong các trang dữ liệu 408A, 408B, v.v.. Hơn nữa, mỗi dòng của trang dữ liệu bổ sung 410 (trong vùng siêu dữ liệu giả 406) là vị trí siêu dữ liệu miền ảo, cùng lúc đó vị trí siêu dữ liệu miền vật lý tương ứng với dòng cuối cùng của mỗi trang dữ liệu 408A, 408B, v.v.. Cần hiểu rằng dòng cuối cùng là ví dụ về dòng được xác định trước trong đó siêu dữ liệu có thể được lưu trữ và trong các phương án khác siêu dữ liệu có thể được lưu trữ trong bất kỳ dòng được xác định trước nào khác. Ví dụ, trong một phương án

khác (không được thể hiện) dòng thứ nhất có thể là dòng được tạo cấu hình để lưu trữ siêu dữ liệu. Trong một phương án khác nữa (không được thể hiện), dòng thứ hai có thể là dòng được tạo cấu hình để lưu trữ siêu dữ liệu, v.v.

Như được mô tả trên Fig.5, có thể lưu ý rằng sơ đồ địa chỉ hoặc cấu hình 400 được mô tả ở trên (Fig.4), liên quan hoặc ánh xạ các vị trí dữ liệu và siêu dữ liệu, có thể cùng tồn tại với sơ đồ địa chỉ hoặc cấu hình 500 liên quan đến dữ liệu không có bất kỳ siêu dữ liệu được gắn kèm. Cấu hình 500 có thể tạo cấu hình dải địa chỉ hoặc vùng 502 của DRAM 210 để chỉ bao gồm vùng dữ liệu 504 và không có vùng siêu dữ liệu được gắn kèm. Dải địa chỉ hoặc vùng 502 và 402 (Fig.4) có thể là hai dải địa chỉ hoặc vùng của cùng DRAM 210 (Fig.2. Vùng dữ liệu 504 có thể được tạo cấu hình là hai hoặc nhiều trang dữ liệu DRAM 508A, 508B, v.v..

Như được mô tả trên Fig.6, phương pháp làm ví dụ 600 để thay đổi vị trí siêu dữ liệu trong hệ thống DRAM có thể bao gồm các sự kiện hoặc hành động được mô tả bên dưới liên quan đến các khối 602-614. Phương pháp 600 phản ánh rằng việc ánh xạ hoặc thay đổi vị trí dữ liệu và siêu dữ liệu được mô tả ở trên liên quan đến Fig.4 có thể xảy ra động, nghĩa là, để đáp lại các yêu cầu giao dịch bộ nhớ từ máy khách 202 (Fig.2. Phương pháp 600 có thể được thực hiện hoặc theo cách khác được điều khiển bởi, ví dụ, bộ đồng xử lý 206 được mô tả ở trên, hoặc bởi bộ đồng xử lý 206 và bộ điều khiển bộ nhớ 208 kết hợp. Bộ đồng xử lý 206, bộ điều khiển bộ nhớ 208, hoặc bộ đồng xử lý 206 và bộ điều khiển bộ nhớ 208 kết hợp có thể là ví dụ về phương tiện để thực hiện các chức năng được mô tả dưới đây liên quan đến phương pháp 600. Trong phần mô tả dưới đây của phương pháp 600, các thuật ngữ “thứ nhất,” “thứ hai”, v.v., có thể được sử dụng để phân biệt các phần tử hoặc các bước và không được mô tả hoặc ám chỉ bất kỳ thứ tự, trình tự, xếp hạng, v.v.. Ví dụ, tham chiếu trong phương pháp 600 đến các trang “thứ nhất” và “thứ hai” không bị giới hạn ở các trang dữ liệu tương ứng 408A và 408B được mô tả ở trên đối với Fig.4 hoặc bất kỳ trang nào khác có thể liên tiếp, nhưng có thể đề cập đến bất kỳ hai trong số các trang dữ liệu 408A, 408B, v.v. hoặc bất kỳ hai trang nào khác. Tương tự, tham chiếu đến các vị trí dữ liệu “thứ nhất” và “thứ hai” không bị giới hạn ở các dòng tương ứng L_0 và L_1 hoặc bất kỳ vị trí dữ liệu nào khác có thể liên tiếp, mà có thể tham chiếu đến hai vị trí bất kỳ.

Như được chỉ ra bởi khối 602, phương pháp 600 có thể bao gồm bước nhận yêu cầu truy cập dữ liệu tại vị trí dữ liệu thứ nhất trong trang thứ nhất và siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu thứ nhất trong trang thứ hai. Yêu cầu có thể bao gồm hai yêu cầu truy cập: một để truy cập dữ liệu và một để truy cập siêu dữ liệu gắn với dữ liệu đó. Ví dụ, các yêu cầu truy cập có thể được nhận bởi một phần của bộ đồng xử lý 206.

Như được chỉ ra bởi khối 604, có thể xác định liệu vị trí dữ liệu thứ nhất trong trang thứ nhất có được tạo cấu hình để lưu trữ siêu dữ liệu hay không. Ví dụ, như được mô tả ở trên liên quan đến Fig.4, dòng cuối cùng của mỗi trang dữ liệu 408A, 408B, v.v., có thể được tạo cấu hình để lưu trữ siêu dữ liệu. Do đó, trong phương án làm ví dụ trong đó dòng cuối cùng của mỗi trang dữ liệu được tạo cấu hình để lưu trữ siêu dữ liệu chứ không phải dữ liệu, có thể xác định (khối 604) liệu vị trí dữ liệu thứ nhất có phải là dòng cuối cùng của trang dữ liệu hay không. Nghĩa là, có thể xác định đích của yêu cầu truy cập dữ liệu có phải là dòng cuối cùng của trang dữ liệu hay không. Cần hiểu rằng theo phương án làm ví dụ được mô tả trong sáng chế, bộ đồng xử lý 206 được tạo cấu hình để sử dụng ánh xạ hoặc cấu hình 400 được mô tả ở trên đối với Fig.4, và do đó có thể xác định xem vị trí dữ liệu miền vật lý cụ thể được tạo cấu hình để lưu trữ siêu dữ liệu (chứ không phải dữ liệu) hay được tạo cấu hình để lưu trữ dữ liệu (chứ không phải siêu dữ liệu). Nghĩa là, trong phương án làm ví dụ, bộ đồng xử lý 206 có thể xác định xem đích của yêu cầu truy cập dữ liệu có phải là dòng cuối cùng của trang dữ liệu hay không. Vị trí dữ liệu thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu cũng có thể được xem là vị trí siêu dữ liệu thứ hai.

Như được chỉ ra bởi khối 606, nếu xác định được (khối 604) rằng vị trí dữ liệu thứ nhất trong trang thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu, thì dữ liệu tại vị trí dữ liệu thứ nhất trong trang thứ nhất có thể được truy cập. Ví dụ, bộ điều khiển bộ nhớ 208 (Fig.2) có thể truy cập dữ liệu bằng cách đưa ra các lệnh cho DRAM 210, truyền thông dữ liệu với DRAM 210, v.v., như mô tả ở trên đối với Fig.2. Đề cập lại đến ánh xạ hoặc cấu hình 400 được mô tả ở trên đối với Fig.4, nếu vị trí dữ liệu thứ nhất tương ứng với một trong các dòng thứ nhất đến dòng áp chót (nghĩa là dòng dữ liệu L_0 đến L_{Z-2}) của một trong các trang dữ liệu 408A, 408B, v.v., sau đó vị trí dữ liệu miền vật lý có thể được truy cập.

Như được chỉ ra bởi khối 612, nếu xác định được (khối 604) rằng vị trí dữ liệu thứ nhất trong trang thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu thì vị trí siêu dữ liệu (“thứ hai”) trong trang thứ nhất có thể được xác định. Nhắc lại ví dụ được mô tả ở trên liên quan đến Fig.4, mỗi dòng của trang dữ liệu bổ sung 410 của vùng siêu dữ liệu giả 406 được ánh xạ tới dòng cuối cùng của một trong các trang dữ liệu 408A, 408B, v.v.. Do đó, trong ví dụ này, vị trí siêu dữ liệu “thứ hai” trong trang thứ nhất là dòng cuối cùng của trang thứ nhất. Ví dụ, nếu trang thứ nhất là trang dữ liệu 408A thì vị trí siêu dữ liệu thứ hai có thể được xác định là dòng cuối cùng của trang dữ liệu 408A. Tương tự, nếu trang thứ nhất là trang dữ liệu 408B thì vị trí siêu dữ liệu thứ hai có thể được xác định là dòng cuối cùng của trang dữ liệu 408B. Vị trí siêu dữ liệu thứ hai là vị trí của siêu dữ liệu đã được "thay đổi vị trí" khỏi vùng siêu dữ liệu giả 406. Như được chỉ ra bởi khối 613, sau khi vị trí siêu dữ liệu (“thứ hai”) trong trang thứ nhất đã được xác định (khối 612), siêu dữ liệu được gắn kèm có thể được truy cập tại vị trí siêu dữ liệu đó trong trang thứ nhất.

Việc truy cập vị trí dữ liệu theo khối 606 có thể bao gồm việc dịch (ví dụ, bởi bộ điều khiển bộ nhớ 208) vị trí dữ liệu miền ảo thành vị trí dữ liệu miền vật lý. Yêu cầu truy cập dữ liệu có thể cung cấp vị trí dữ liệu thứ nhất dưới dạng số trang ảo và số dòng. Tương tự, yêu cầu truy cập siêu dữ liệu có thể cung cấp vị trí siêu dữ liệu thứ nhất dưới dạng số trang ảo và số dòng. Số trang trong miền ảo có thể tương ứng với, và do đó được dịch sang (ví dụ, bởi bộ điều khiển bộ nhớ 208), hàng DRAM trong miền vật lý. Số dòng trong miền ảo có thể tương ứng với, và do đó được dịch sang nhóm gồm một hoặc nhiều cột DRAM trong hàng đó. (Số cột DRAM trên mỗi dòng dữ liệu có thể được xác định trước, tức là hằng số trong suốt hoạt động của phương án).

Như được chỉ ra bởi khối 608, nếu xác định được (khối 604) rằng vị trí dữ liệu thứ nhất trong trang thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu, thì vị trí dữ liệu thứ hai trong trang thứ hai có thể được xác định dựa vào vị trí dữ liệu thứ nhất. Đề cập lại đến ánh xạ hoặc cấu hình ví dụ được mô tả ở trên đối với Fig.4, nếu được xác định rằng vị trí dữ liệu được yêu cầu hoặc đích là dòng cuối cùng (tức là, dòng dữ liệu $LZ-1$) của một trong các trang dữ liệu 408A, 408B, v.v., thì có thể xác định được dòng nào của trang dữ liệu bổ sung 410 trong vùng siêu dữ liệu giả 406 tương ứng với trang dữ liệu đó. Theo ví dụ này, trang dữ liệu bổ sung 410 là trang "thứ hai" được tham chiếu ở trên

và dòng trong trang dữ liệu bổ sung 410 là vị trí dữ liệu "thứ hai".

Như được chỉ ra bởi khối 610, khi vị trí dữ liệu thứ hai trong trang thứ hai đã được xác định (khối 608), vị trí dữ liệu thứ hai đó có thể được truy cập. Trong trường hợp truy cập đọc, việc truy cập dữ liệu tại vị trí dữ liệu thứ hai có thể bao gồm bộ điều khiển bộ nhớ 208 (Fig.2) đọc dữ liệu từ DRAM 210. Trong trường hợp truy cập ghi, việc truy cập dữ liệu có thể bao gồm bộ điều khiển bộ nhớ 208 để đảm bảo rằng dữ liệu đã được ghi vào, nghĩa là được lưu trữ, trong DRAM 210.

Như được chỉ ra bởi khối 614, nếu xác định được (khối 604) rằng vị trí dữ liệu thứ nhất trong trang thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu, siêu dữ liệu được gắn kèm sau đó có thể được truy cập tại vị trí dữ liệu thứ nhất đó. Việc truy cập siêu dữ liệu được gắn kèm có thể bao gồm việc dịch (ví dụ, bởi bộ điều khiển bộ nhớ 208) vị trí dữ liệu thứ nhất trong miền ảo sang miền vật lý. Số trang trong miền ảo có thể tương ứng với, và do đó được dịch đến, hàng DRAM trong miền vật lý. Số dòng trong miền ảo có thể tương ứng với, và do đó được dịch sang nhóm gồm một hoặc nhiều cột DRAM trong hàng đó.

Lưu ý rằng trong sự kiện vị trí dữ liệu thứ nhất trong trang thứ nhất được tạo cấu hình để lưu trữ dữ liệu (nghĩa là không được tạo cấu hình để lưu trữ siêu dữ liệu), thì vị trí dữ liệu thứ nhất nằm trong cùng hàng DRAM với vị trí siêu dữ liệu. Với tham chiếu đến ánh xạ ví dụ trên Fig.4, trong sự kiện vị trí dữ liệu đích là bất kỳ dòng dữ liệu nào từ L_0 đến L_{Z-2} của trang dữ liệu 408A, trong miền vật lý tương ứng với hàng DRAM thứ nhất, sau đó bộ điều khiển bộ nhớ 208 (Fig.2) không gặp phải xung đột vì vị trí siêu dữ liệu đích nằm trong cùng hàng DRAM thứ nhất đó. Tương tự, trong sự kiện vị trí dữ liệu đích là bất kỳ dòng dữ liệu nào từ L_0 đến L_{Z-2} của trang dữ liệu 408B, mà trong miền vật lý tương ứng với hàng DRAM thứ hai, thì bộ điều khiển bộ nhớ 208 (Fig.2) không gặp phải xung đột vì vị trí siêu dữ liệu đích nằm trong cùng hàng DRAM thứ hai đó. Tuy nhiên, trong sự kiện vị trí dữ liệu thứ nhất trong trang thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu (nghĩa là vị trí dữ liệu thứ nhất đã được tạo cấu hình là vị trí siêu dữ liệu thứ hai), thì vị trí dữ liệu thứ hai (nơi dữ liệu đích đã được thay đổi vị trí) sẽ không nằm trong cùng hàng DRAM với vị trí siêu dữ liệu thứ hai. Một lần nữa tham chiếu đến ánh xạ ví dụ trên Fig.4, trong sự kiện vị trí dữ liệu được dịch hoặc được thay đổi vị trí

nằm trong trang dữ liệu DRAM bổ sung 410, thì bộ điều khiển bộ nhớ 208 gặp phải xung đột do vị trí siêu dữ liệu được dịch hoặc được thay đổi vị trí là dòng cuối cùng của một trong các trang dữ liệu 408A, 408B, v.v.. Giao dịch xung đột ít có khả năng xảy ra hơn so với giao dịch trùng (nghĩa là không có xung đột) vì trong mỗi trang DRAM chỉ có một dòng (ví dụ, dòng cuối cùng) trong số Z dòng được thay đổi vị trí đến trang dữ liệu DRAM bổ sung 410. Trung bình, giao dịch xung đột có thể xảy ra chỉ một lần trong mỗi Z yêu cầu truy cập dữ liệu và siêu dữ liệu được gắn kèm.

Lưu ý rằng phương pháp 600 được mô tả ở trên (Fig.6) liên quan đến yêu cầu truy cập dữ liệu có siêu dữ liệu được gắn kèm. Mặc dù không được thể hiện trên Fig. 6, nếu nhận được yêu cầu truy cập dữ liệu không có bất kỳ siêu dữ liệu được gắn kèm nào, vị trí dữ liệu trong một trang bộ nhớ khác, chẳng hạn như trang 508A hoặc 508B được mô tả ở trên (Fig.5), có thể được xác định và truy cập theo cách thông thường. Nghĩa là, phương pháp 600 có thể được thực hiện có điều kiện, tùy thuộc vào việc dữ liệu được yêu cầu thuộc loại có siêu dữ liệu được gắn kèm hay loại khác không có siêu dữ liệu được gắn kèm.

Trên Fig.7, ví dụ về phép xác định trong khối 608 của phương pháp 600 (Fig.6) có thể được mô tả. Tức là Fig.7 minh họa ví dụ về dịch địa chỉ 702 trong vùng dữ liệu 404 (Fig.4) sang địa chỉ trong vùng siêu dữ liệu giả 406. Theo ví dụ này, kích thước trang DRAM S_P là 2kB và kích thước dòng S_L là 64B. Theo ví dụ này, kích thước của vùng dữ liệu 402 (Fig.4) có thể là 32GB, bao gồm 16k trang 2kB đã nói ở trên và kích thước của vùng siêu dữ liệu 404 có thể là 1 GB. Tức là tỷ lệ dữ liệu so với siêu dữ liệu đối với loại dữ liệu mà ví dụ liên quan là 32:1.

Theo ví dụ này, các địa chỉ có độ dài 36 bit. Các bit địa chỉ có thể được tham chiếu bằng số vị trí của chúng, từ bit ít quan trọng nhất (least-significant bit - "LSB") ở vị trí "0" hoặc, để cho ngắn gọn, "bit 0," đến bit quan trọng nhất (most-significant bit - "MSB") ở vị trí "35" hoặc ngắn gọn là "bit 35". Địa chỉ có thể được tổ chức theo nhóm bit. Kích thước hoặc số lượng bit trong nhóm bit có thứ tự thấp nhất là sáu vì kích thước của nhóm bit này là logarit cơ số 2 ("log2") của kích thước dòng S_L (Fig.4). Kích thước hoặc số bit trong nhóm bit là năm vì kích thước của nhóm bit này là log2 của số dòng trên mỗi trang, mà trong ví dụ này là số dòng 64B trên mỗi trang 2kB. Kích thước hoặc

số bit trong nhóm bit tiếp theo là 25 vì kích thước của nhóm bit này là $\log_2$ của kích thước vùng dữ liệu là 32GB.

Trên Fig.8, minh họa việc dịch địa chỉ trong vùng dữ liệu 404 (Fig.4) của vùng bộ nhớ 402 sang địa chỉ vật lý trong vùng siêu dữ liệu giả 406 của vùng bộ nhớ 402. Vùng bộ nhớ 402 có thể được phân biệt bởi địa chỉ bắt đầu vùng dữ liệu ("DataStartAddr") và địa chỉ kết thúc vùng dữ liệu ("DataEndAddr") của vùng dữ liệu 404 và địa chỉ bắt đầu vùng siêu dữ liệu ("MdataStartAddr") và địa chỉ kết thúc vùng siêu dữ liệu ("MdataEndAddr") trong vùng siêu dữ liệu 406. "DataAddr" là địa chỉ của dòng 64B trong một trong các trang dữ liệu 408A, 408B, v.v. (không được thể hiện riêng lẻ trên Fig.8) tương ứng với vị trí dữ liệu miền ảo được yêu cầu. Mặc dù không được thể hiện trên Fig. 8, địa chỉ của dòng cuối cùng của trang dữ liệu 408A, 408B, v.v., có thể được gọi là "LastLineDataAddr." Trong các phương trình dưới đây, biểu tượng mũi tên " $\rightarrow$ " có thể được sử dụng để chỉ báo rằng kết quả (nghĩa là vế phải của phương trình) được lấy từ vế trái của phương trình. Trong một số phương trình, mũi tên có thể ám chỉ phép toán.

Theo ví dụ này, việc xác định theo khối 604 (Fig.6) có thể được thực hiện bằng cách xác định xem các bit từ 6 đến 10 của địa chỉ dữ liệu có phải đều là bit một hay không, nghĩa là liệu $\text{DataAddr}[10:6] = '11111'$ hay không. Nếu các bit từ 6 đến 10 của địa chỉ dữ liệu là '11111' thì dòng tương ứng với vị trí dữ liệu được yêu cầu hoặc đích là dòng cuối cùng của một trong các trang dữ liệu 408A, 408B, v.v.. Nếu các bit từ 6 đến 10 của địa chỉ dữ liệu là '11111' thì địa chỉ được yêu cầu trong vùng dữ liệu 404 (Fig.4) có thể được dịch sang địa chỉ trong vùng siêu dữ liệu giả 406.

Tiếp theo ví dụ này, việc xác định theo khối 608 (Fig.6) có thể được thực hiện bằng cách trước tiên xác định phần bù địa chỉ của dòng cuối cùng từ địa chỉ bắt đầu ("LastLineDataAddrOffset"). Phần bù địa chỉ có thể được xác định bằng cách trừ địa chỉ bắt đầu của vùng dữ liệu từ LastLineDataAddr:

$$\text{LastDataLineAddr} - \text{DataStartAddr} \rightarrow \text{LastLineDataAddrOffset} \quad (\text{Phương trình 1})$$

Phần bù địa chỉ của dòng cuối cùng trong trang dữ liệu có thể được xác định từ LastLineDataAddrOffset:

LastLineDataAddrOffset -> DataPageAddrOffset

(Phương trình

2)

Trong Phương trình 2 ở trên, ký hiệu mũi tên (“->”) ngụ ý một hoạt động theo đó các bit địa chỉ từ 6 đến 10 (nghĩa là phần bù dòng trong trang) bị loại bỏ. Kết quả, LastLineDataAddrOffset là phần bù địa chỉ của dòng dữ liệu cần được thêm vào địa chỉ bắt đầu vùng siêu dữ liệu giả, MdataStartAddr. Sau đó, địa chỉ của dòng dữ liệu được thay đổi vị trí (“NewDataAddr”) trong vùng siêu dữ liệu giả 406 có thể được xác định bằng cách thêm phần bù địa chỉ của dòng cuối cùng trong trang dữ liệu vào địa chỉ bắt đầu của vùng siêu dữ liệu:

DataPageAddrOffset + MdataStartAddr -> NewDataAddr

(Phương trình

3)

Trên Fig.9, các ví dụ về giá trị địa chỉ trong vùng bộ nhớ 402 được thể hiện. Vì giá trị của DataAddr[10:6] là '11111' nên Phương trình 1 được đánh giá: $0x8_000107C0 = 0xc000_0000 + 0x7_4001_07C0$. Phương trình 2 được đánh giá: $0x7_4001_07C0 = 0x3A00_0800$. Phương trình 3 được đánh giá: $0x3A00_0800 = 0x8000_0000 + 0xBA00_0800$. Do đó, địa chỉ của dòng dữ liệu được thay đổi vị trí (NewDataAddr) trong vùng siêu dữ liệu giả 406 trong ví dụ này là $0xBA00_0800$.

Tham chiếu đến Fig.10, ví dụ về phép xác định trong khối 612 của phương pháp 600 (Fig.6) có thể được mô tả. Tức là Fig.10 minh họa ví dụ về dịch địa chỉ 802 trong vùng siêu dữ liệu giả 406 sang địa chỉ 804 trong vùng dữ liệu 404. Vì DataAddr, mà là vị trí dữ liệu được yêu cầu hoặc đích, là dòng cuối cùng của một trong các trang dữ liệu 408A, 408B, v.v., trong ví dụ này, các phép xác định trong khối 612 (Fig.6) có thể được thực hiện như được chỉ ra trên Fig.10 và Phương trình 4:

DataAddr -> LastLineDataAddrOffset (MdataAddr)

(Phương trình

4)

Trong Phương trình 4 ở trên, ký hiệu mũi tên (“->”) ngụ ý phép toán theo đó các bit địa chỉ từ 6 đến 10 buộc phải là '11111'. Sử dụng các giá trị từ ví dụ được mô tả ở trên đối với Fig.9, địa chỉ của siêu dữ liệu được gắn kèm trong vùng dữ liệu 404 (MdataAddr) sẽ là $0x8_0001_07FE$.

Phương pháp 600 được mô tả ở trên (Fig.6) cũng có thể áp dụng cho các phương án trong đó DRAM 210 (Fig.2) được tổ chức thành nhiều nhóm dây (bank group - “BG”) trong đó dữ liệu được xen kẽ, chẳng hạn như LPDDR5 được tạo cấu hình trong “chế độ nhóm dây” (còn được gọi là “mặt nạ nhóm dây”). Như được minh họa trên Fig.11, vùng bộ nhớ làm ví dụ 1100 (ví dụ, trong DRAM 210) có thể được tổ chức trong miền ảo dưới dạng nhóm gồm bốn trang ảo (Virtual Page), được đề cập trên Fig.11 như: “Virtual Page0,” “Virtual Page1,” “Virtual Page2” và “Virtual Page3.” Tương tự như phương án trước được mô tả ở trên (ví dụ, liên quan đến Fig.4), trong đó DRAM 210 không được tổ chức thành nhiều BG, trong phương án này, dữ liệu có thể được đánh địa chỉ trong miền ảo theo số trang ảo và số dòng (không được thể hiện). Ví dụ, máy khách 202 (Fig.2) có thể yêu cầu truy cập dữ liệu bằng cách chỉ ra một trong số Virtual Page0 - Virtual Page3 và một số dòng trong trang ảo đó.

Trong sơ đồ đánh địa chỉ LPDDR5, các bit 14-13 đánh địa chỉ dây, các bit [10:8] đánh địa chỉ BG và các bit [5:0] đánh địa chỉ cột bắt đầu trong mảng DRAM (không được thể hiện trên Fig.11). Trong ví dụ minh họa, DRAM có thể được tạo cấu hình trong bốn BG, được đề cập trên Fig.11 như: “BG0,” “BG1,” “BG2” và “BG3.” Mỗi nhóm dây có thể bao gồm bốn dây. Mỗi trang ảo có thể được tổ chức theo dòng, được thể hiện trên Fig.11 bởi các khối tô bóng mờ. Các kiểu tô bóng khác nhau phân biệt các nhóm dây với nhau, như được nêu trong Chú thích trên Fig.11. Mỗi trang ảo và mỗi dòng có thể có kích thước tương ứng, ví dụ, 2kB và 64B, như trong phương án trước được mô tả ở trên (ví dụ, đối với Fig.4). Lưu ý rằng bốn trang ảo, Virtual Page0 - Virtual Page3 là vùng bộ nhớ miền máy khách hoặc ảo, trong khi BG và dây là cấu trúc miền vật lý hoặc DRAM.

Tương tự như phương án trước, mỗi dòng dữ liệu có siêu dữ liệu được gắn kèm. Trang siêu dữ liệu ảo (không được thể hiện) có thể được tạo cấu hình (nghĩa là trong miền ảo) để lưu trữ siêu dữ liệu được gắn kèm. Do đó, yêu cầu truy cập dữ liệu và siêu dữ liệu được gắn kèm có thể xử lý dữ liệu ở một trong số Virtual Page0 - Virtual Page3 và xử lý siêu dữ liệu trong trang siêu dữ liệu ảo.

Chế độ nhóm dây LPDDR5 xen kẽ các dây, có thể được gọi là “B0,” “B1,” “B2” và “B3,” trên các nhóm dây. Lưu ý trên Fig.11 rằng mỗi cột trang ảo biểu diễn truy cập

64B, được đánh địa chỉ bằng bit [7:6]. Mẫu trong đó xen kẽ dây xảy ra, nghĩa là mẫu trong đó các truy cập liên tiếp di chuyển từ dây này sang dây khác trên các BG, không được chỉ ra trên Fig.11 và có thể là bất kỳ mẫu nào, theo cách hiểu của người có hiểu biết trung bình trong lĩnh vực này. Vấn đề ánh xạ hoặc thay đổi vị trí siêu dữ liệu đặt ra thách thức vì việc xen kẽ không phù hợp với kích thước trang. Ví dụ, trong phương án chế độ nhóm dây LPDDR5, trong đó các trang ảo có kích thước 2kB, các nhóm dây không được xen kẽ ở cùng ranh giới kích thước (ví dụ, 2kB) làm trang ảo. Vấn đề về ánh xạ hoặc thay đổi vị trí siêu dữ liệu đặt ra thách thức vì ngay cả khi siêu dữ liệu được thay đổi vị trí đến dòng cuối cùng trong trang dữ liệu ảo (tương tự như LPDDR5), không có gì đảm bảo rằng dữ liệu và siêu dữ liệu sẽ được đặt ở cùng trang nhóm dây vật lý trong DDR. Khi các nhóm dây được xen kẽ ở ranh giới nhỏ hơn kích thước trang, trang ảo sẽ được ánh xạ tới các trang vật lý khác nhau trong các nhóm Dây khác nhau.

Thách thức này có thể được giải quyết bằng cách tạo cấu hình một dòng của mỗi nhóm dây (nghĩa là trong miền vật lý) để lưu trữ siêu dữ liệu (xem Chú thích trên Fig.11). Lưu ý trong ví dụ được thể hiện trên Fig.11 rằng dòng cuối cùng của dây B3 trong mỗi nhóm dây BG0, BG1, BG2 và BG3 được tạo cấu hình để lưu trữ siêu dữ liệu.

Trên Fig.12, ánh xạ được mô tả ở trên (Fig.11) được mô tả theo cách khác. Trên Fig.12, ánh xạ làm ví dụ giữa trang ảo 1200, chẳng hạn như bất kỳ từ Virtual Page0 - Virtual Page3 nào được mô tả ở trên và các vị trí lưu trữ vật lý hoặc DRAM, được biểu thị bằng các mũi tên đường đứt nét. Mỗi phần nhóm dây được ánh xạ 1202 có cấu trúc lưu trữ tương tự như cấu trúc của DRAM 210 được mô tả ở trên đối với Fig.2. Nghĩa là, mỗi dây B0-B3 bao gồm mảng các vị trí lưu trữ được đánh địa chỉ theo hàng và cột. Trong mỗi hàng của mảng, nhóm gồm một số vị trí lưu trữ hoặc cột được xác định trước có thể tương ứng với dòng (ví dụ, 64B) trong miền ảo.

Các khối được tô bóng mờ trong các nhóm dây từ BG0 đến BG3 biểu diễn ánh xạ của tất cả bốn trang ảo Virtual Page0 - Virtual Page3 tới miền vật lý (DRAM). Do xen kẽ nhóm dây, trong mỗi nhóm dây từ BG0 đến BG3, một phần của mỗi trang ảo từ Virtual Page0 - Virtual Page3 ánh xạ tới một phần của một trong các dây từ B0 đến B3, chẳng hạn như, các phần được khoanh tròn 1202 của dây B0. Quan trọng là, hãy lưu ý rằng tất cả các dòng siêu dữ liệu (xem Chú thích trên Fig.11) ánh xạ tới cùng trang

DRAM, được mô tả trên Fig.12 nhằm mục đích rõ ràng là hàng thứ tư từ trên xuống trong dãy B0 của mỗi nhóm dây từ BG0 đến BG3. Tuy nhiên, trong các ví dụ khác, các dòng siêu dữ liệu có thể ánh xạ tới bất kỳ trang DRAM nào.

Trong ví dụ được minh họa trên các Fig.11 đến Fig.12, yêu cầu truy cập dữ liệu của máy khách ở bất kỳ dòng nào trong trang ảo 1200 ngoài dòng cuối cùng của dãy B3 trong cùng nhóm dây với dữ liệu được yêu cầu sẽ không dẫn đến xung đột, và do đó không thực hiện thay đổi vị trí của địa chỉ dữ liệu được yêu cầu. Lưu ý rằng mặc dù chỉ có một trang trong mỗi dãy B0-B3 trong nhóm dây có thể được mở tại một thời điểm, các trang có thể được mở ở các dãy khác nhau cùng một lúc. Ví dụ, trang chứa dòng siêu dữ liệu (ví dụ, bốn hàng tạo thành trang vật lý của dãy B3 trên Fig. 12) có thể được mở trong khi trang chứa dòng dữ liệu được yêu cầu trong B0, B1 hoặc B2 cũng đang mở. Trong ví dụ minh họa, chỉ yêu cầu truy cập dữ liệu hướng đến dòng cuối cùng của dãy B3 sẽ dẫn đến xung đột, bởi vì dòng đó được tạo cấu hình trong miền vật lý để lưu trữ siêu dữ liệu thay vì dữ liệu. Nếu xác định được rằng yêu cầu truy cập dữ liệu được chuyển đến dòng cuối cùng của dãy B3, việc thay đổi vị trí địa chỉ dữ liệu được yêu cầu sẽ được thực hiện. Vì có 32 dòng trong mỗi trang vật lý cho bất kỳ dãy nào trong ví dụ minh họa, xác suất yêu cầu truy cập dữ liệu bị xung đột là 32. Với tỷ lệ dữ liệu trên siêu dữ liệu là 1/32 (trình bày trong ví dụ trên Fig.11), 31 trong tổng số 32 lần truy cập vào trang vật lý DRAM có siêu dữ liệu nằm trong cùng một trang vật lý trong nhóm Dây. Bất kỳ truy cập siêu dữ liệu nào sau lần truy cập dữ liệu cho 31 dòng đó sẽ không dẫn đến xung đột trang vì truy cập siêu dữ liệu sẽ dẫn đến lần truy cập trùng trang. Truy cập dữ liệu vào dòng cuối cùng của trang được thay đổi vị trí đến dòng trong trang siêu dữ liệu giả (ánh xạ tới trang DRAM khác) dẫn đến không trùng/xung đột trang. Do đó, xác suất xung đột trang là 1/32.

Việc xác định theo khối 604 (Fig.6) có thể được thực hiện bằng cách thứ nhất xác định địa chỉ ("DataAddrSwapped") có các bit mặt nạ nhóm dây (Addr[10:8]) được hoán đổi với các bit trang có thứ tự thấp hơn (Addr[12:11]). Nếu các bit từ 6 đến 10 của DataAddrSwapped là '1111' thì dòng tương ứng với vị trí dữ liệu được yêu cầu hoặc đích là dòng cuối cùng của một trong các trang từ Virtual Page0 - Virtual Page3. Nếu các bit từ 6 đến 10 của địa chỉ dữ liệu là '1111' thì việc xác định theo khối 608 (Fig.6) có thể được thực hiện bằng cách dịch địa chỉ dữ liệu được yêu cầu sang dòng cuối cùng

của nhóm dây cuối cùng (xem Phương trình 1 ở trên). Phần bù địa chỉ của dòng cuối cùng trong trang có thể được xác định từ `LastLineDataAddrOffset` (xem Phương trình 2 ở trên). Sau đó, địa chỉ của dòng dữ liệu được thay đổi vị trí (`NewDataAddr`) có thể được xác định bằng cách thêm phần bù địa chỉ của dòng cuối cùng trong trang dữ liệu vào địa chỉ bắt đầu của vùng siêu dữ liệu (xem Phương trình 3 ở trên).

Sử dụng cùng giá trị địa chỉ làm ví dụ được thể hiện trên Fig.9 của `0x8_0001_07C0`, vì giá trị của `DataAddr[10:6]` không phải là '11111' sau khi hoán đổi các bit của nhóm dây với các bit của trang có thứ tự thấp hơn, nên có thể được xác định rằng địa chỉ dữ liệu không phải là dòng cuối cùng của trang nhóm dây. Do đó, không thực hiện thay đổi vị trí địa chỉ dữ liệu. Địa chỉ siêu dữ liệu có thể được xác định bằng cách hoán đổi các bit mặt nạ nhóm dây (`Addr[10:8]`) với các bit trang có thứ tự thấp hơn (`Addr[12:11]`) để tạo thành `DataAddrSwapped`. Sau đó, địa chỉ siêu dữ liệu có thể được xác định (xem Phương trình 4). Sử dụng giá trị cho `DataAddr` là `0x8_001_07C0`, địa chỉ được thay đổi vị trí của siêu dữ liệu được gắn kèm (`MdataAddr`) sẽ là `0x8_0001_1FD6`.

Theo một ví dụ khác, giá trị địa chỉ có thể là `0x0_F000_FFC0`. Vì giá trị của `DataAddr[10:6]` không phải là '11111' sau khi hoán đổi các bit của nhóm dây với các bit của trang có thứ tự thấp hơn, nên có thể xác định được rằng địa chỉ dữ liệu là dòng cuối cùng của trang nhóm dây. Việc xác định theo khối 612 (Fig.6) có thể được thực hiện bằng cách trước tiên xác định phần bù địa chỉ của dòng cuối cùng từ địa chỉ bắt đầu. Phần bù địa chỉ có thể được xác định bằng cách trừ địa chỉ bắt đầu của vùng dữ liệu từ `LastLineDataAddr` (xem Phương trình 1 ở trên). Sau đó, phần bù địa chỉ của dòng cuối cùng trong trang có thể được xác định (xem Phương trình 2 ở trên). Địa chỉ của dòng dữ liệu được thay đổi vị trí ("`NewDataAddr`") có thể được xác định bằng cách thêm phần bù địa chỉ của dòng cuối cùng trong trang dữ liệu vào địa chỉ bắt đầu của vùng siêu dữ liệu (xem Phương trình 3 ở trên). Theo ví dụ này, trong đó giá trị địa chỉ là `0x0_F000_FFC0`, Phương trình 1 được tính thành: `0x0_F000_FFC0 - 0xC000_0000 -> 0x3000_FFC0`. Phương trình 2 được tính thành: `0x3000_FFC0 -> 0x180_07C0`. Phương trình 3 được tính thành: `0x3000_FFC0 + 0x8000_0000 -> 0x8180_FFC0`. Như vậy địa chỉ của dòng dữ liệu đã thay đổi vị trí là `0x8180_FFC0`.

Địa chỉ siêu dữ liệu đã thay đổi vị trí có thể được xác định bằng cách hoán đổi

các bit mặt nạ nhóm dãy (Addr[10:8]) với các bit trang có thứ tự thấp hơn (Addr[12:11]) để tạo thành DataAddrSwapped. Sau đó, địa chỉ siêu dữ liệu có thể được xác định (xem Phương trình 4). Sử dụng giá trị cho DataAddr là 0x0_F000_FFC0, địa chỉ được thay đổi vị trí của siêu dữ liệu được gắn kèm (MdataAddr) sẽ là 0xF000_FFFE.

Như được minh họa trên Fig.13, các phương án làm ví dụ về hệ thống và phương pháp thay đổi vị trí siêu dữ liệu trong hệ thống DRAM có thể được cung cấp trong thiết bị điện toán di động (“PCD”) 1300. PCD 1300 có thể là ví dụ về hệ thống 100 (Fig.1) hoặc 200 (Fig.2).

PCD 1300 có thể gồm SoC 1302. SoC 1302 có thể gồm CPU 1304, GPU 1306, DSP 1307, bộ xử lý tín hiệu tương tự 1308, hoặc các bộ xử lý khác. CPU 1304 có thể bao gồm nhiều lõi, chẳng hạn như lõi thứ nhất 1304A, lõi thứ hai 1304B, v.v., tới lõi thứ N 1304N.

Bộ điều khiển hiển thị 1310 và bộ điều khiển màn hình cảm ứng 1312 có thể được ghép nối với CPU 1304. Màn hình cảm ứng 1314 bên ngoài SoC 1302 có thể được ghép nối với bộ điều khiển màn hình 1310 và bộ điều khiển màn hình cảm ứng 1312. PCD 1300 có thể còn bao gồm bộ giải mã video 1316 được ghép nối với CPU 1304. Bộ khuếch đại video 1318 được ghép nối với bộ giải mã video 1316 và màn hình cảm ứng 1314. Cổng video 1320 có thể được ghép nối với bộ khuếch đại video 1318. Bus nối tiếp đa năng (universal serial bus - USB) có thể cũng được ghép nối với CPU 1304, và cổng USB 1324 có thể được ghép nối với bộ điều khiển USB 1322. Thẻ module nhận dạng thuê bao (subscriber identity module - SIM) 1326 cũng có thể được ghép nối với CPU 1304.

Một hoặc nhiều bộ nhớ có thể ghép nối với CPU 1304. Một hoặc nhiều bộ nhớ có thể bao gồm cả bộ nhớ khả biến và bất biến. Ví dụ về bộ nhớ khả biến bao gồm bộ nhớ truy cập ngẫu nhiên tĩnh (static random access memory - SRAM) 1328 và các RAM động (dynamic RAM - DRAM) 1330 và 1331. Các bộ nhớ như vậy có thể nằm bên ngoài SoC 1302, chẳng hạn như DRAM 1330, hoặc bên trong SoC 1302, chẳng hạn như DRAM 1331. Bộ điều khiển DRAM 1332 được ghép nối với CPU 1304 có thể điều khiển việc ghi dữ liệu vào, và đọc dữ liệu từ, các DRAM 1330 và 1331. Theo các phương án khác, bộ điều khiển DRAM như vậy có thể được gồm trong bộ xử lý, chẳng hạn như

CPU 1304. Bộ điều khiển DRAM 1332 có thể là ví dụ về bộ điều khiển bộ nhớ 208 (Fig. 2) hoặc bộ điều khiển bộ nhớ 208 kết hợp với bộ đồng xử lý 206. Một trong hai DRAM 1330 và 1331 có thể là ví dụ về DRAM 210 (Fig.2).

CODEC âm thanh lập thể 1334 có thể được ghép nối với bộ xử lý tín hiệu tương tự 1308. Hơn nữa, bộ khuếch đại âm thanh 1336 có thể được ghép nối với CODEC âm thanh lập thể 1334. Loa âm thanh lập thể thứ nhất và thứ hai 1338 và 1340 có thể lần lượt được ghép nối với bộ khuếch đại âm thanh 1336. Ngoài ra, bộ khuếch đại micro 1342 có thể được ghép nối với CODEC âm thanh lập thể 1334, và micro 1344 có thể được ghép nối với bộ khuếch đại micro 1342. Bộ dò sóng vô tuyến điều chế tần số (frequency modulation - FM) 1346 có thể được ghép nối với CODEC âm thanh lập thể 1334. Anten FM 1348 có thể được ghép nối với bộ dò sóng vô tuyến FM 1346. Ngoài ra, tai nghe âm thanh lập thể 1350 có thể được ghép nối với CODEC âm thanh lập thể 1334. Các thiết bị khác mà có thể được ghép nối với CPU 1304 bao gồm một hoặc nhiều camera kỹ thuật số (ví dụ, CCD hoặc CMOS) 1352.

Modem hoặc bộ thu phát RF 1354 có thể được ghép nối với bộ xử lý tín hiệu tương tự 1308. Bộ chuyển đổi RF 1356 có thể được ghép nối với bộ thu phát RF 1354 và anten RF 1358. Ngoài ra, bàn phím 1360, tai nghe đơn âm có micro 1362, và thiết bị rung 1364 có thể được ghép nối với bộ xử lý tín hiệu tương tự 1308.

SoC 1302 có thể có một hoặc nhiều cảm biến nhiệt nằm trên chip hoặc bên trong 1370A và có thể được ghép nối với một hoặc nhiều cảm biến nhiệt nằm ngoài chip hoặc bên ngoài 1370B. Bộ điều khiển bộ chuyển đổi tương tự sang kỹ thuật số (“analog-to-digital converter - ADC”) 1372 có thể chuyển đổi tình trạng sụt áp do cảm biến nhiệt 1370A và 1370B tạo ra thành tín hiệu số. Nguồn điện 1374 và mạch tích hợp quản lý nguồn điện (“power management integrated circuit - PMIC”) 1376 có thể cung cấp điện cho SoC 1302.

Firmware hoặc phần mềm có thể được lưu trữ trong bất kỳ bộ nhớ nào được mô tả ở trên, chẳng hạn như DRAM 1330 hoặc 1331, SRAM 1328, v.v., hoặc có thể được lưu trữ trong bộ nhớ cục bộ mà có thể truy cập trực tiếp bởi phần cứng bộ xử lý mà phần mềm hoặc firmware thực thi trên đó. Việc thực thi firmware hoặc phần mềm như vậy có thể điều khiển các khía cạnh của bất kỳ phương pháp nào được mô tả ở trên hoặc tạo

cấu hình các khía cạnh của bất kỳ hệ thống nào được mô tả ở trên. Bất kỳ bộ nhớ hoặc phương tiện lưu trữ bất biến nào khác có firmware hoặc phần mềm được lưu trữ trong đó ở dạng đọc được bằng máy tính để thực thi bởi phần cứng bộ xử lý có thể là ví dụ về "phương tiện đọc được bằng máy tính", như thuật ngữ này được hiểu trong từ điển của sáng chế.

Các phương án thay thế sẽ trở nên rõ ràng đối với một người có hiểu biết trung bình trong lĩnh vực kỹ thuật sáng chế thuộc về mà không rời khỏi tinh thần và phạm vi của nó. Do đó, mặc dù các khía cạnh được lựa chọn đã được minh họa và được mô tả chi tiết, nhưng có thể hiểu rằng các thay thế và sửa đổi khác nhau có thể được thực hiện trong đó mà không nằm ngoài bản chất và phạm vi của sáng chế, như được định nghĩa bởi các yêu cầu bảo hộ sau đây.

Ví dụ về các phương án triển khai được mô tả trong các mệnh đề được đánh số sau đây.

1. Phương pháp thay đổi vị trí siêu dữ liệu trong bộ nhớ truy cập ngẫu nhiên động (dynamic random access memory - DRAM), bao gồm:

nhận yêu cầu thứ nhất để truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất của DRAM và siêu dữ liệu được gắn kèm trong trang bộ nhớ thứ hai của DRAM;

xác định xem vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu hay không;

truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

xác định vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

xác định vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ

nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu;

truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu; và

truy cập siêu dữ liệu được gắn kèm tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu.

2. Phương pháp theo mệnh đề 1, trong đó việc truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất xảy ra đồng thời với truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất là không được tạo cấu hình để lưu trữ siêu dữ liệu.

3. Phương pháp theo mệnh đề 2, trong đó:

truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất bao gồm truy cập hàng DRAM; và

truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất bao gồm truy cập hàng DRAM.

4. Phương pháp theo mệnh đề 1, còn bao gồm:

nhận yêu cầu thứ hai để truy cập dữ liệu thứ hai tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ ba, dữ liệu thứ hai không có siêu dữ liệu được gắn kèm; và

truy cập dữ liệu thứ hai tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ ba.

5. Phương pháp theo mệnh đề 1, trong đó việc xác định xem vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất có được tạo cấu hình để lưu trữ siêu dữ liệu hay không bao gồm việc xác định xem vị trí dữ liệu thứ nhất có phải là dòng cuối cùng trong trang bộ nhớ thứ nhất hay không.

6. Phương pháp theo mệnh đề 1, trong đó truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất bao gồm việc truy cập dòng cuối cùng trong số nhiều dòng trong trang bộ nhớ thứ nhất.

7. Phương pháp theo mệnh đề 1, trong đó việc truy cập siêu dữ liệu được gắn

kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất bao gồm việc truy cập dòng cuối cùng của nhiều dòng trong trang bộ nhớ thứ nhất, dòng cuối cùng bao gồm siêu dữ liệu gắn với dữ liệu trong tất cả các dòng khác trong trang bộ nhớ thứ nhất.

8. Phương pháp theo mệnh đề 1, trong đó bộ nhớ có hai hoặc nhiều nhóm dây, mỗi nhóm dây bao gồm hai hoặc nhiều dây, và vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất là dòng cuối cùng của dây cuối cùng trong mỗi nhóm dây.

9. Hệ thống thay đổi vị trí siêu dữ liệu, bao gồm:

bộ nhớ truy cập ngẫu nhiên động (dynamic random access memory - DRAM);

và

bộ xử lý được tạo cấu hình để:

nhận yêu cầu thứ nhất để truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất và siêu dữ liệu được gắn kèm trong trang bộ nhớ thứ hai;

xác định xem vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất có được tạo cấu hình để lưu trữ siêu dữ liệu hay không;

truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

xác định vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

xác định vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu;

truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu; và

truy cập siêu dữ liệu được gắn kèm tại vị trí dữ liệu thứ nhất trong trang bộ nhớ

thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu.

10. Hệ thống theo mệnh đề 9, trong đó bộ xử lý được tạo cấu hình để truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất đồng thời với truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu.

11. Hệ thống theo mệnh đề 10, trong đó:

bộ xử lý được tạo cấu hình để truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất bằng cách truy cập hàng DRAM; và

bộ xử lý được tạo cấu hình để truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất bằng cách truy cập hàng DRAM.

12. Hệ thống theo mệnh đề 9, trong đó bộ xử lý còn được tạo cấu hình để:

nhận yêu cầu thứ hai để truy cập dữ liệu thứ hai tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ ba, dữ liệu thứ hai không có siêu dữ liệu được gắn kèm; và

truy cập dữ liệu thứ hai tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ ba.

13. Hệ thống theo mệnh đề 9, trong đó bộ xử lý được tạo cấu hình để xác định xem vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất có được tạo cấu hình để lưu trữ siêu dữ liệu hay không bằng cách xác định xem vị trí dữ liệu thứ nhất có phải là dòng cuối cùng trong trang bộ nhớ thứ nhất hay không.

14. Hệ thống theo mệnh đề 9, trong đó bộ xử lý được tạo cấu hình để truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất bằng cách truy cập dòng cuối cùng trong số nhiều dòng trong trang bộ nhớ thứ nhất.

15. Hệ thống theo mệnh đề 9, trong đó bộ xử lý được tạo cấu hình để truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất bằng việc truy cập dòng cuối cùng trong số nhiều dòng trong trang bộ nhớ thứ nhất, dòng cuối cùng bao gồm siêu dữ liệu gắn với dữ liệu trong tất cả các dòng khác trong trang bộ nhớ thứ nhất.

16. Hệ thống theo mệnh đề 9, trong đó bộ nhớ có hai hoặc nhiều nhóm dây, mỗi nhóm dây bao gồm hai hoặc nhiều dây, và vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất là dòng cuối cùng của dây cuối cùng trong mỗi nhóm dây.

17. Hệ thống thay đổi vị trí siêu dữ liệu trong bộ nhớ truy cập ngẫu nhiên động (DRAM), bao gồm:

phương tiện nhận yêu cầu thứ nhất để truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất của DRAM và siêu dữ liệu được gắn kèm trong trang bộ nhớ thứ hai của DRAM;

phương tiện để xác định xem vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất có được tạo cấu hình để lưu trữ siêu dữ liệu hay không;

phương tiện để truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

phương tiện để xác định vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

phương tiện truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

phương tiện để xác định vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu;

phương tiện để truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu; và

phương tiện để truy cập siêu dữ liệu được gắn kèm tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu.

18. Hệ thống theo mệnh đề 17, trong đó phương tiện để truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất bao gồm phương tiện để truy cập siêu dữ liệu được gắn kèm đồng thời với truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu.

19. Hệ thống theo mệnh đề 18, trong đó:

phương tiện để truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất bao gồm phương tiện để truy cập hàng DRAM; và

phương tiện để truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất bao gồm phương tiện để truy cập hàng DRAM.

20. Hệ thống theo mệnh đề 17, còn bao gồm:

phương tiện để nhận yêu cầu thứ hai để truy cập dữ liệu thứ hai tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ ba, dữ liệu thứ hai không có siêu dữ liệu được gắn kèm; và

phương tiện để truy cập dữ liệu thứ hai tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ ba.

21. Hệ thống theo mệnh đề 17, trong đó phương tiện để xác định xem vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất có được tạo cấu hình để lưu trữ siêu dữ liệu hay không bao gồm phương tiện để xác định xem vị trí dữ liệu thứ nhất có phải là dòng cuối cùng trong trang bộ nhớ thứ nhất hay không.

22. Hệ thống theo mệnh đề 17, trong đó phương tiện để truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất bao gồm phương tiện để truy cập dòng cuối cùng trong số nhiều dòng trong trang bộ nhớ thứ nhất.

23. Hệ thống theo mệnh đề 17, trong đó phương tiện để truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất bao gồm phương tiện để truy cập dòng cuối cùng trong số nhiều dòng trong trang bộ nhớ thứ nhất, dòng cuối cùng bao gồm siêu dữ liệu gắn với dữ liệu trong tất cả các dòng khác trong trang bộ nhớ thứ nhất.

24. Hệ thống theo mệnh đề 17, trong đó bộ nhớ có hai hoặc nhiều nhóm dây, mỗi nhóm dây bao gồm hai hoặc nhiều dây, và vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất là dòng cuối cùng của dây cuối cùng trong mỗi nhóm dây.

25. Phương tiện đọc được bằng máy tính để thay đổi vị trí siêu dữ liệu trong bộ nhớ truy cập ngẫu nhiên động (DRAM), bao gồm phương tiện bất biến đọc được bằng máy tính lưu trữ trên đó trong các lệnh dạng có thể thực thi được bằng máy tính, khi được thực thi bởi bộ xử lý, sẽ tạo cấu hình bộ xử lý để:

nhận yêu cầu thứ nhất để truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất và siêu dữ liệu được gắn kèm trong trang bộ nhớ thứ hai;

xác định xem vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất có được tạo cấu hình để lưu trữ siêu dữ liệu hay không;

truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

xác định vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

xác định vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu;

truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu; và

truy cập siêu dữ liệu được gắn kèm tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu.

26. Phương tiện đọc được bằng máy tính theo mệnh đề 25, trong đó các lệnh

tạo cấu hình bộ xử lý để truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất đồng thời với truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu.

27. Phương tiện đọc được bằng máy tính theo mệnh đề 26, trong đó các lệnh tạo cấu hình bộ xử lý để:

truy cập siêu dữ liệu được gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất bằng cách truy cập hàng DRAM; và

truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất bằng cách truy cập hàng DRAM.

28. Phương tiện đọc được bằng máy tính theo mệnh đề 25, trong đó các lệnh còn tạo cấu hình bộ xử lý để:

nhận yêu cầu thứ hai để truy cập dữ liệu thứ hai tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ ba, dữ liệu thứ hai không có siêu dữ liệu được gắn kèm; và

truy cập dữ liệu thứ hai tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ ba.

29. Phương tiện đọc được bằng máy tính theo mệnh đề 25, trong đó các lệnh tạo cấu hình bộ xử lý để xác định xem vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất có được tạo cấu hình để lưu trữ siêu dữ liệu hay không bằng cách xác định xem vị trí dữ liệu thứ nhất có phải là dòng cuối cùng trong trang bộ nhớ thứ nhất hay không.

30. Phương tiện đọc được bằng máy tính theo mệnh đề 25, trong đó các lệnh tạo cấu hình bộ xử lý để truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất bằng cách truy cập dòng cuối cùng trong số nhiều dòng trong trang bộ nhớ thứ nhất.

YÊU CẦU BẢO HỘ

1. Phương pháp thay đổi vị trí siêu dữ liệu trong bộ nhớ truy cập ngẫu nhiên động, dynamic random access memory - DRAM, phương pháp này bao gồm các bước:

nhận (602) yêu cầu thứ nhất để truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất của DRAM và siêu dữ liệu gắn kèm trong trang bộ nhớ thứ hai của DRAM;

xác định (604) xem vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất có được tạo cấu hình để lưu trữ siêu dữ liệu hay không;

truy cập (606) dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

xác định (612) vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

truy cập (613) siêu dữ liệu gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

xác định (608) vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu;

truy cập (610) dữ liệu thứ nhất tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu; và

truy cập (614) siêu dữ liệu gắn kèm tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu.

2. Phương pháp theo điểm 1, trong đó bước truy cập (613) siêu dữ liệu gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất xảy ra đồng thời với bước truy cập (606) dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

và hơn nữa, trong đó:

truy cập (613) siêu dữ liệu gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất bao gồm truy cập hàng DRAM; và

truy cập (606) dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất bao gồm truy cập hàng DRAM.

3. Phương pháp theo điểm 1, phương pháp này còn bao gồm các bước:

nhận yêu cầu thứ hai để truy cập dữ liệu thứ hai tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ ba, dữ liệu thứ hai không có siêu dữ liệu gắn kèm; và

truy cập dữ liệu thứ hai tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ ba.

4. Phương pháp theo điểm 1, trong đó bước xác định (604) xem vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất có được tạo cấu hình để lưu trữ siêu dữ liệu hay không bao gồm xác định xem vị trí dữ liệu thứ nhất có ở dòng cuối trong trang bộ nhớ thứ nhất hay không.

5. Phương pháp theo điểm 1, trong đó bước truy cập (606) dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất bao gồm truy cập dòng cuối trong số nhiều dòng trong trang bộ nhớ thứ nhất.

6. Phương pháp theo điểm 1, trong đó bước truy cập (613) siêu dữ liệu gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất bao gồm truy cập dòng cuối trong số nhiều dòng trong trang bộ nhớ thứ nhất, dòng cuối bao gồm siêu dữ liệu gắn kèm với dữ liệu trong tất cả các dòng khác trong trang bộ nhớ thứ nhất.

7. Phương pháp theo điểm 1, trong đó bộ nhớ có hai hoặc nhiều nhóm dây, mỗi nhóm dây bao gồm hai hoặc nhiều dây, và vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất là dòng cuối cùng của dây cuối cùng trong nhóm dây.

8. Hệ thống (200) thay đổi vị trí siêu dữ liệu, hệ thống này bao gồm:

bộ nhớ truy cập ngẫu nhiên động, DRAM (210); và

bộ xử lý (202) được tạo cấu hình để:

nhận (602) yêu cầu thứ nhất để truy cập dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất và siêu dữ liệu gắn kèm trong trang bộ nhớ

thứ hai;

xác định (604) xem vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất có được tạo cấu hình để lưu trữ siêu dữ liệu hay không;

truy cập (606) dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

xác định (612) vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

truy cập (613) siêu dữ liệu gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

xác định (608) vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu;

truy cập (610) dữ liệu thứ nhất tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ hai nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu; và

truy cập (614) siêu dữ liệu gắn kèm tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất được tạo cấu hình để lưu trữ siêu dữ liệu.

9. Hệ thống theo điểm 8, trong đó bộ xử lý (202) được tạo cấu hình để truy cập (613) siêu dữ liệu gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất đồng thời với truy cập (606) dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất nếu vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất không được tạo cấu hình để lưu trữ siêu dữ liệu;

và hơn nữa trong đó:

bộ xử lý (202) được tạo cấu hình để truy cập (613) siêu dữ liệu gắn kèm tại vị trí

siêu dữ liệu trong trang bộ nhớ thứ nhất bằng cách truy cập hàng DRAM; và

bộ xử lý được tạo cấu hình để truy cập (606) dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất bằng cách truy cập hàng DRAM.

10. Hệ thống theo điểm 8, trong đó bộ xử lý (202) còn được tạo cấu hình để:

nhận yêu cầu thứ hai để truy cập dữ liệu thứ hai tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ ba, dữ liệu thứ hai không có siêu dữ liệu được gắn kèm; và

truy cập dữ liệu thứ hai tại vị trí dữ liệu thứ hai trong trang bộ nhớ thứ ba.

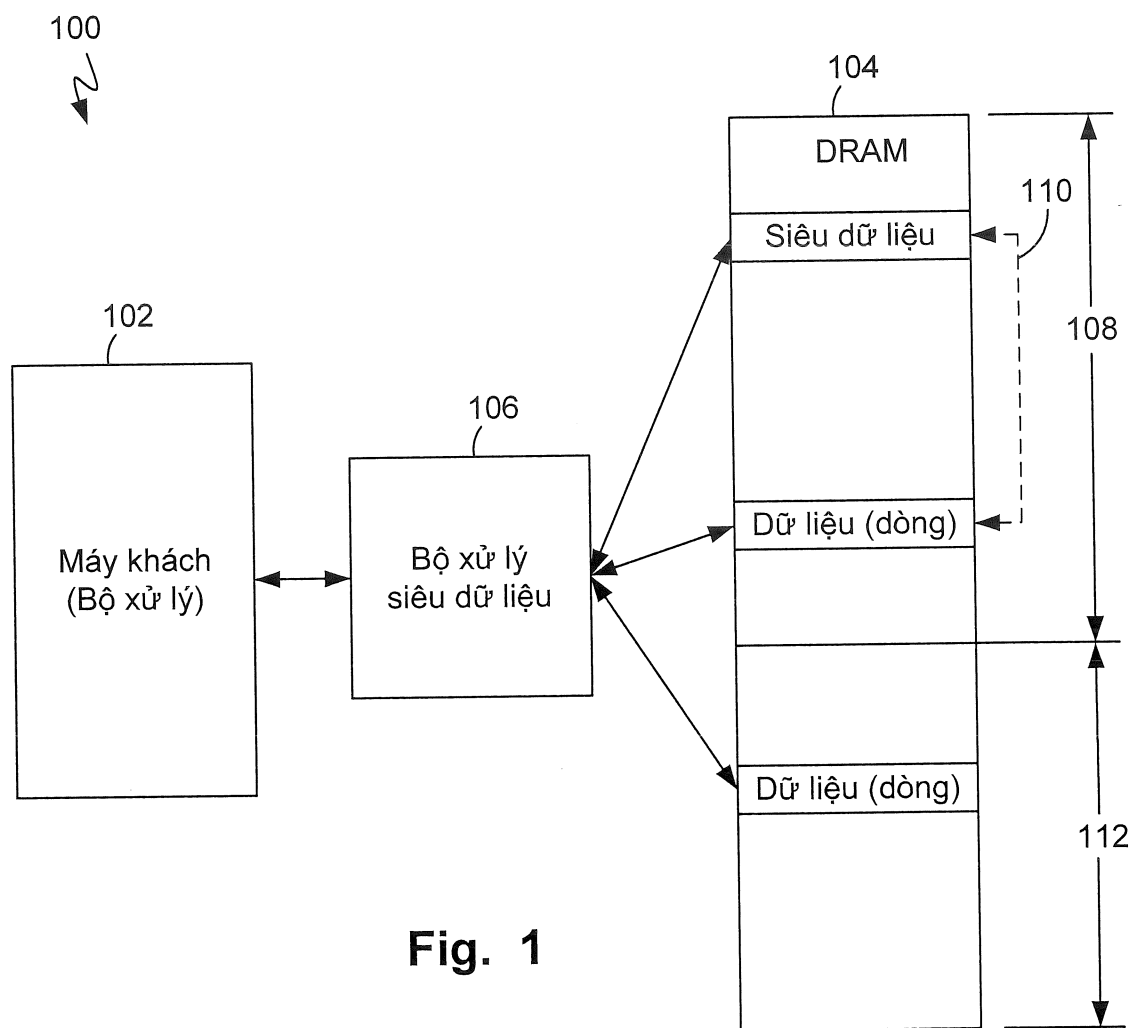
11. Hệ thống theo điểm 8, trong đó bộ xử lý (202) được tạo cấu hình để xác định (604) xem vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất có được tạo cấu hình để lưu trữ siêu dữ liệu hay không bằng cách xác định xem vị trí dữ liệu thứ nhất có phải dòng cuối trong trang bộ nhớ thứ nhất hay không.

12. Hệ thống theo điểm 8, trong đó bộ xử lý (202) được tạo cấu hình để truy cập (606) dữ liệu thứ nhất tại vị trí dữ liệu thứ nhất trong trang bộ nhớ thứ nhất bằng cách truy cập dòng cuối trong số nhiều dòng trong trang bộ nhớ thứ nhất.

13. Hệ thống theo điểm 8, trong đó bộ xử lý (202) được tạo cấu hình để truy cập (613) siêu dữ liệu gắn kèm tại vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất bằng cách truy cập dòng cuối trong số nhiều dòng trong trang bộ nhớ thứ nhất, dòng cuối bao gồm siêu dữ liệu gắn kèm với dữ liệu trong tất cả các dòng khác trong trang bộ nhớ thứ nhất.

14. Hệ thống theo điểm 8, trong đó bộ nhớ truy cập ngẫu nhiên động (210) có hai hoặc nhiều nhóm dây, mỗi nhóm dây bao gồm hai hoặc nhiều dây, và vị trí siêu dữ liệu trong trang bộ nhớ thứ nhất là dòng cuối cùng của dây cuối cùng trong nhóm dây.

15. Phương tiện đọc được bằng máy tính để thay đổi vị trí siêu dữ liệu trong bộ nhớ truy cập ngẫu nhiên động, DRAM (210), bao gồm phương tiện bất biến đọc được bằng máy tính có lưu trữ trên đó các lệnh dạng thực thi được bằng máy tính, khi được thực thi bằng máy tính, sẽ khiến cho máy tính thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 7.



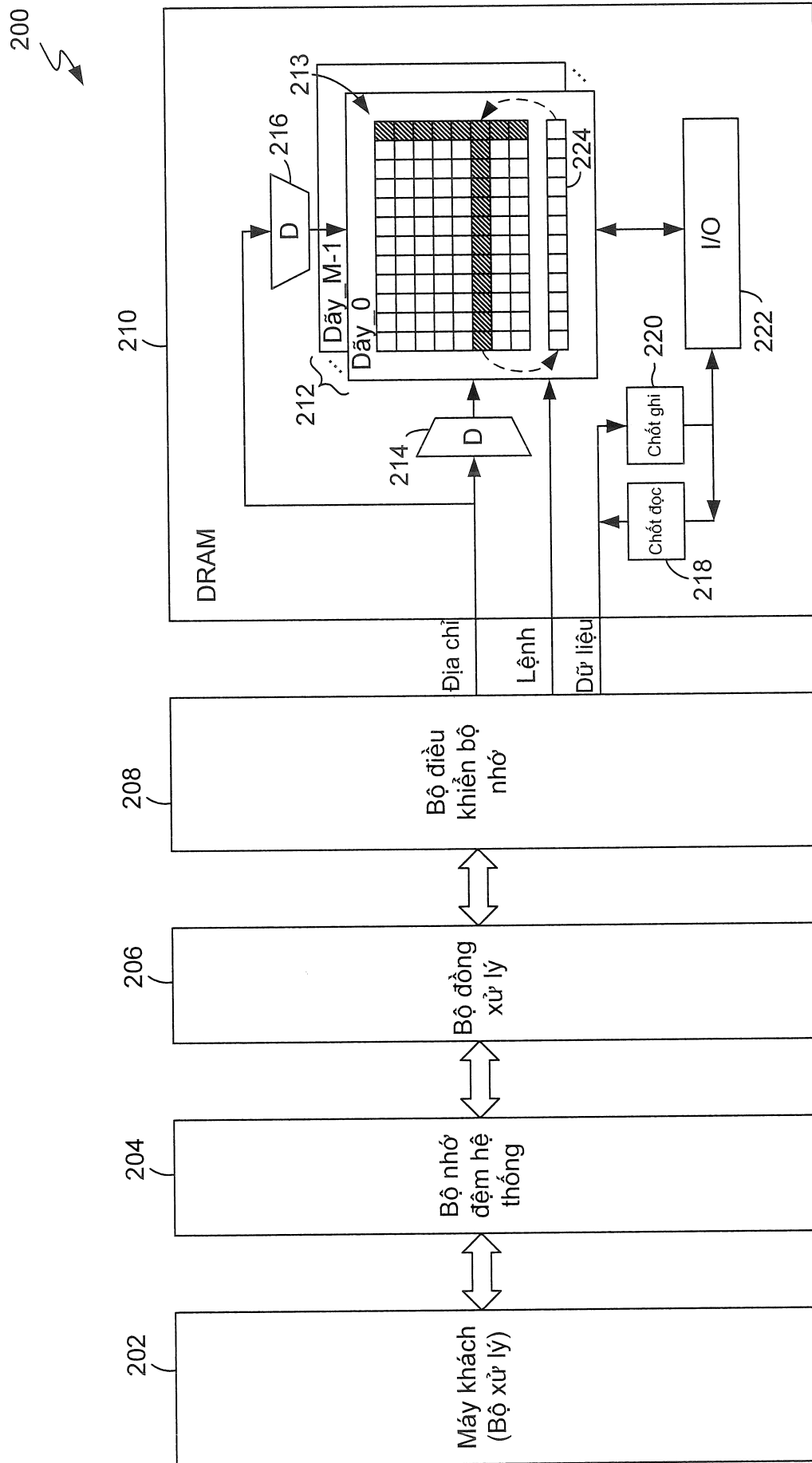


Fig. 2

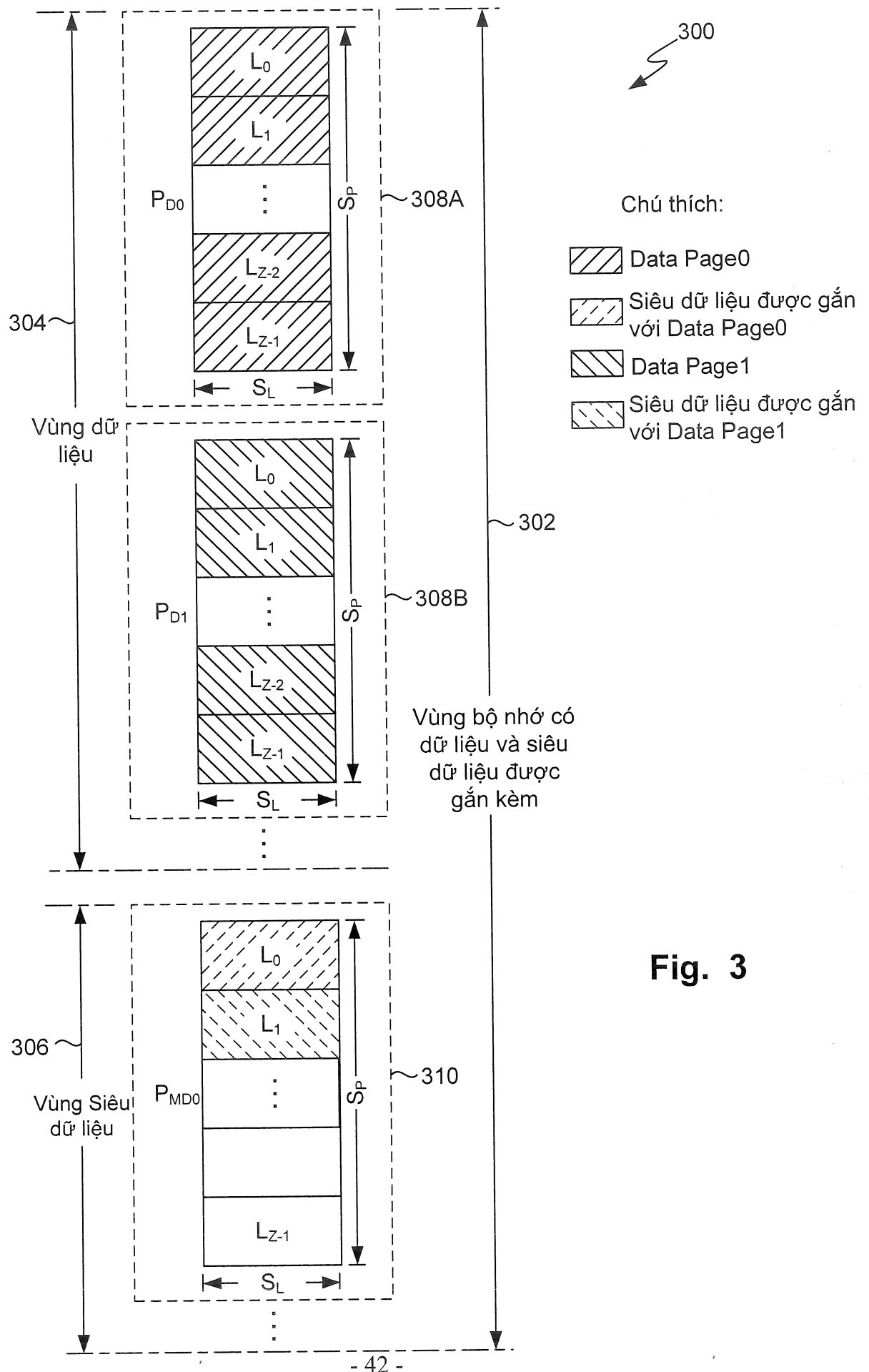
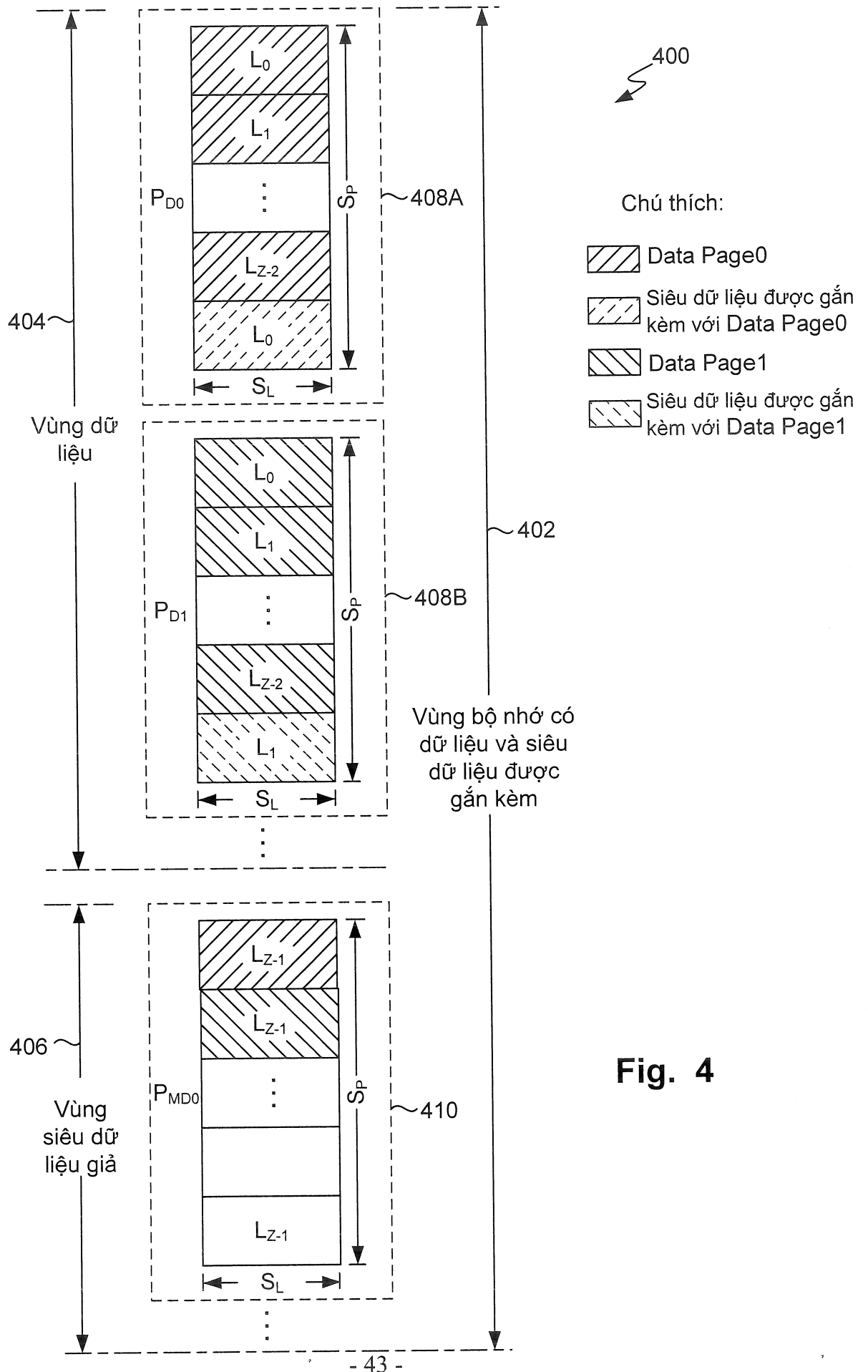


Fig. 3



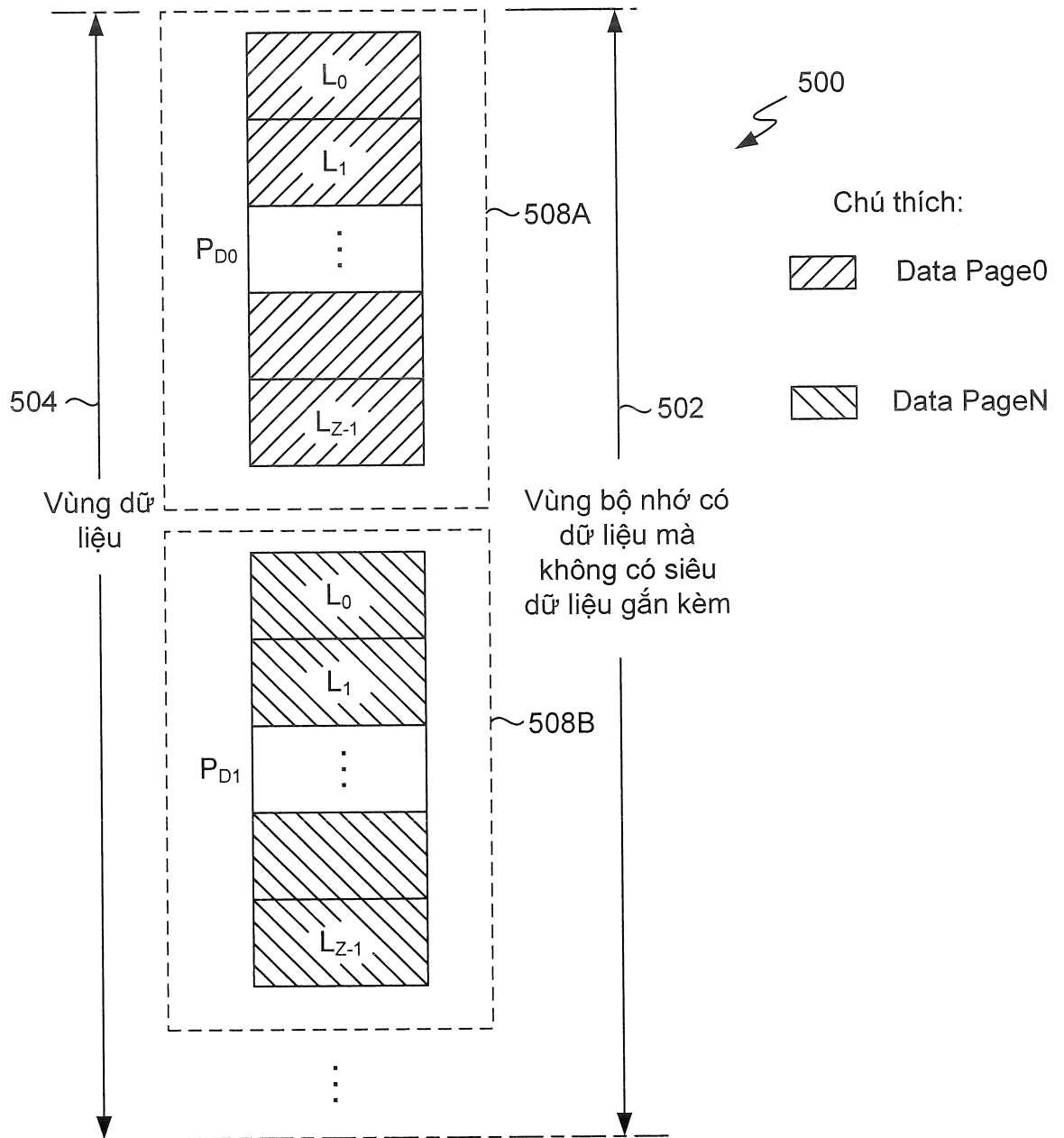


Fig. 5

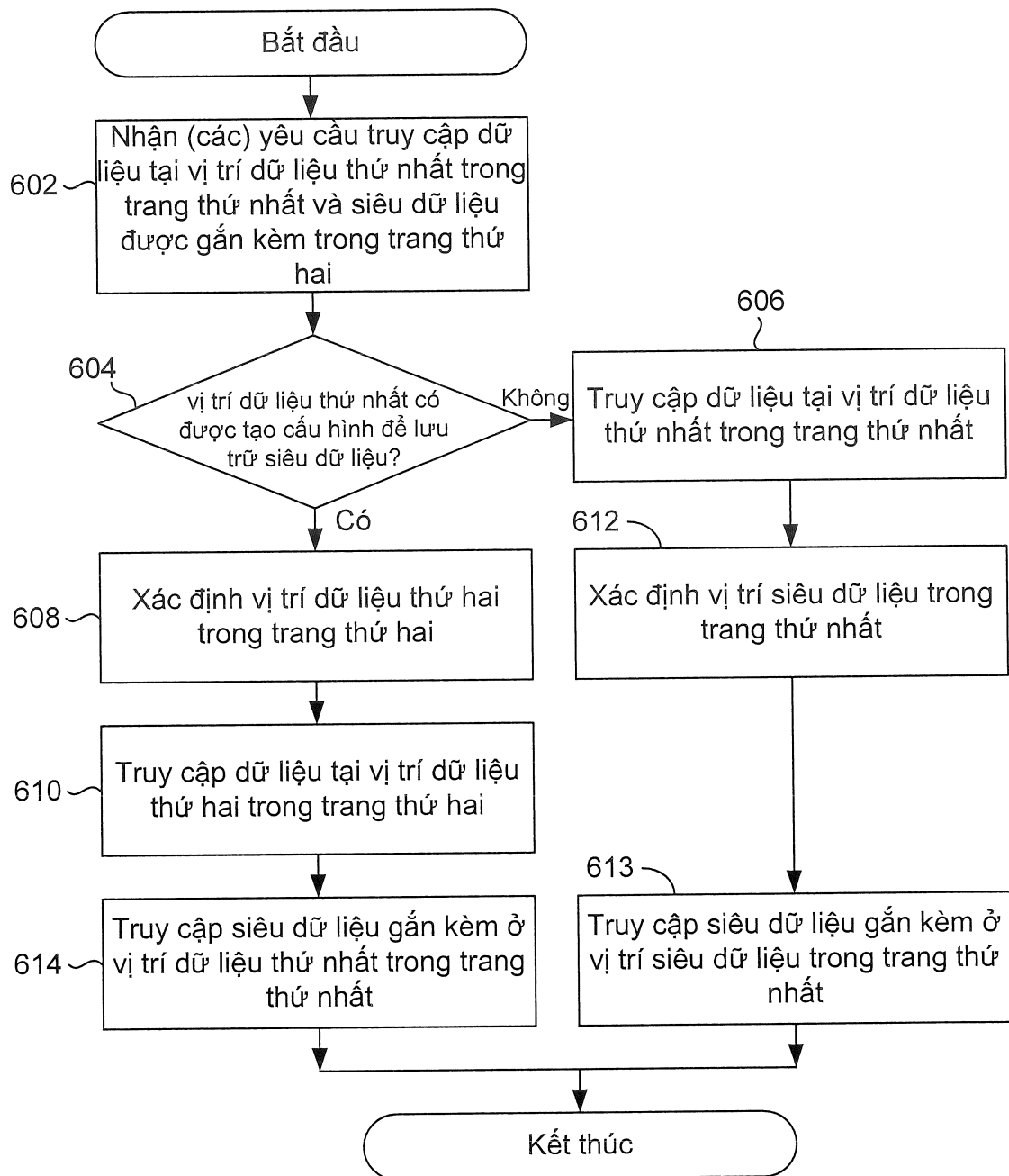


Fig. 6

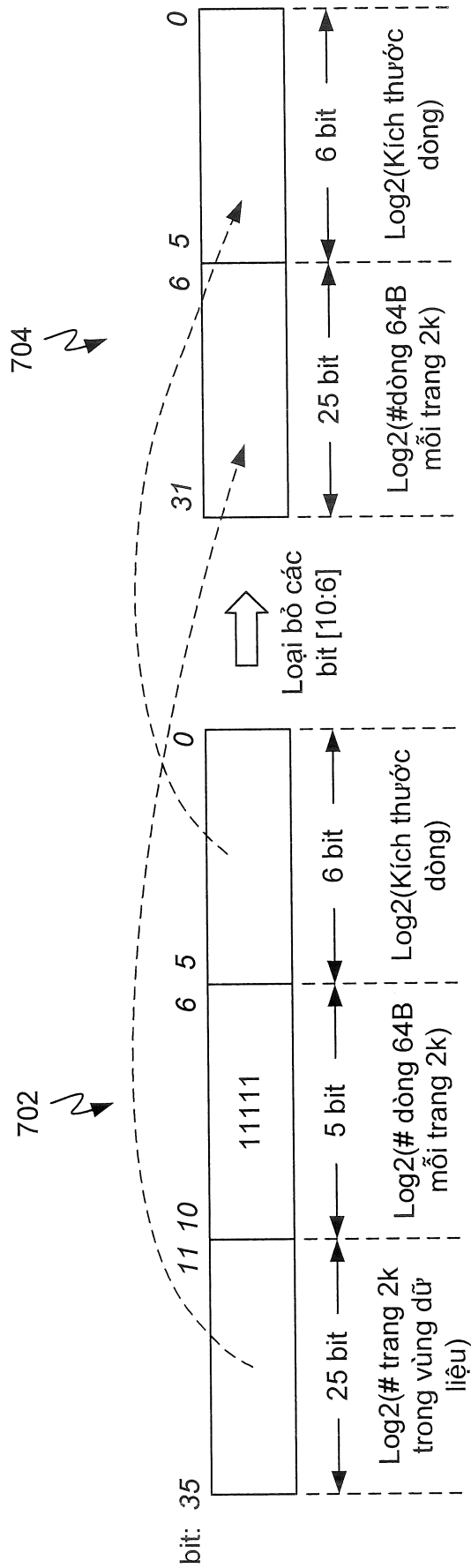
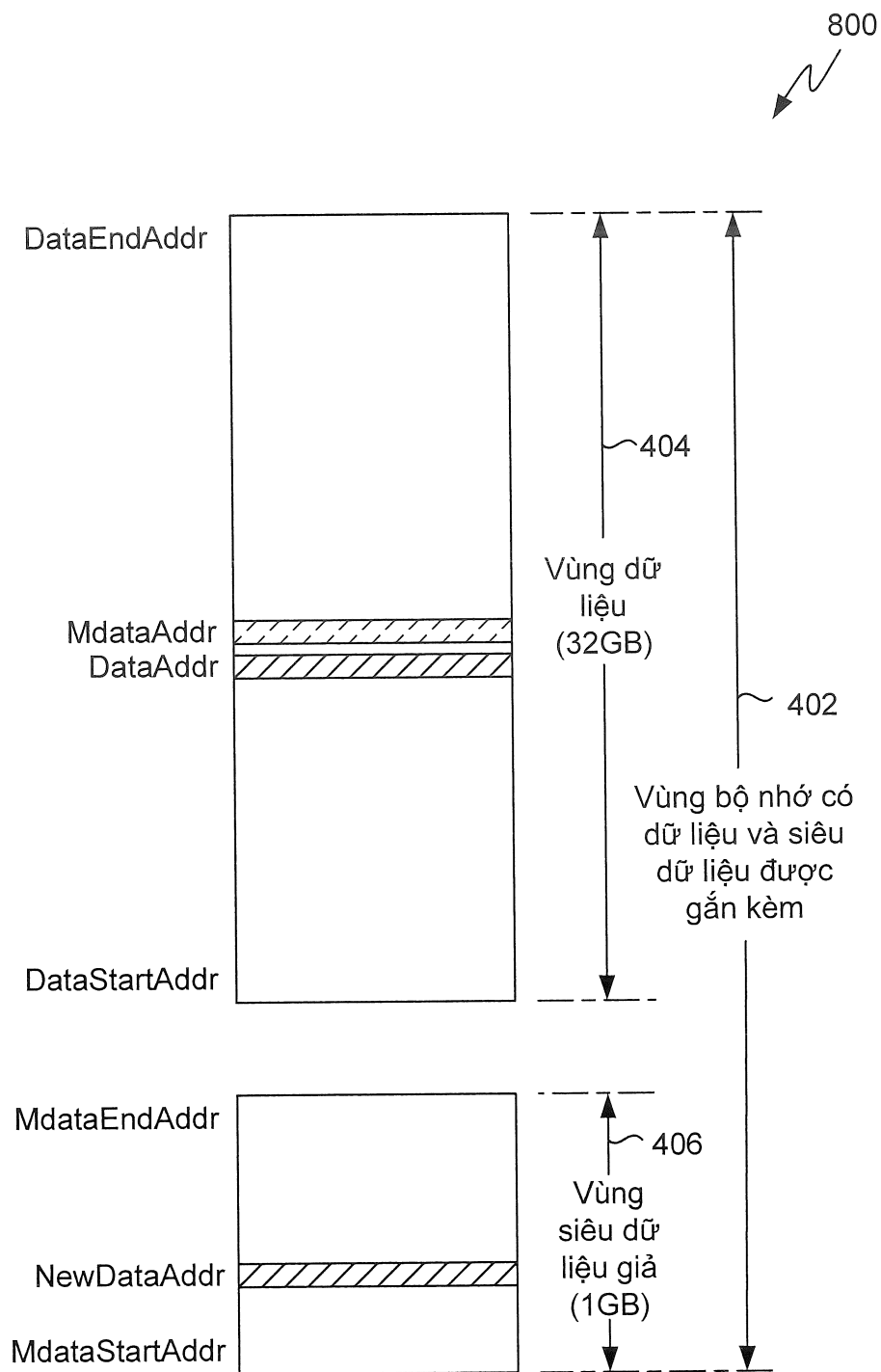
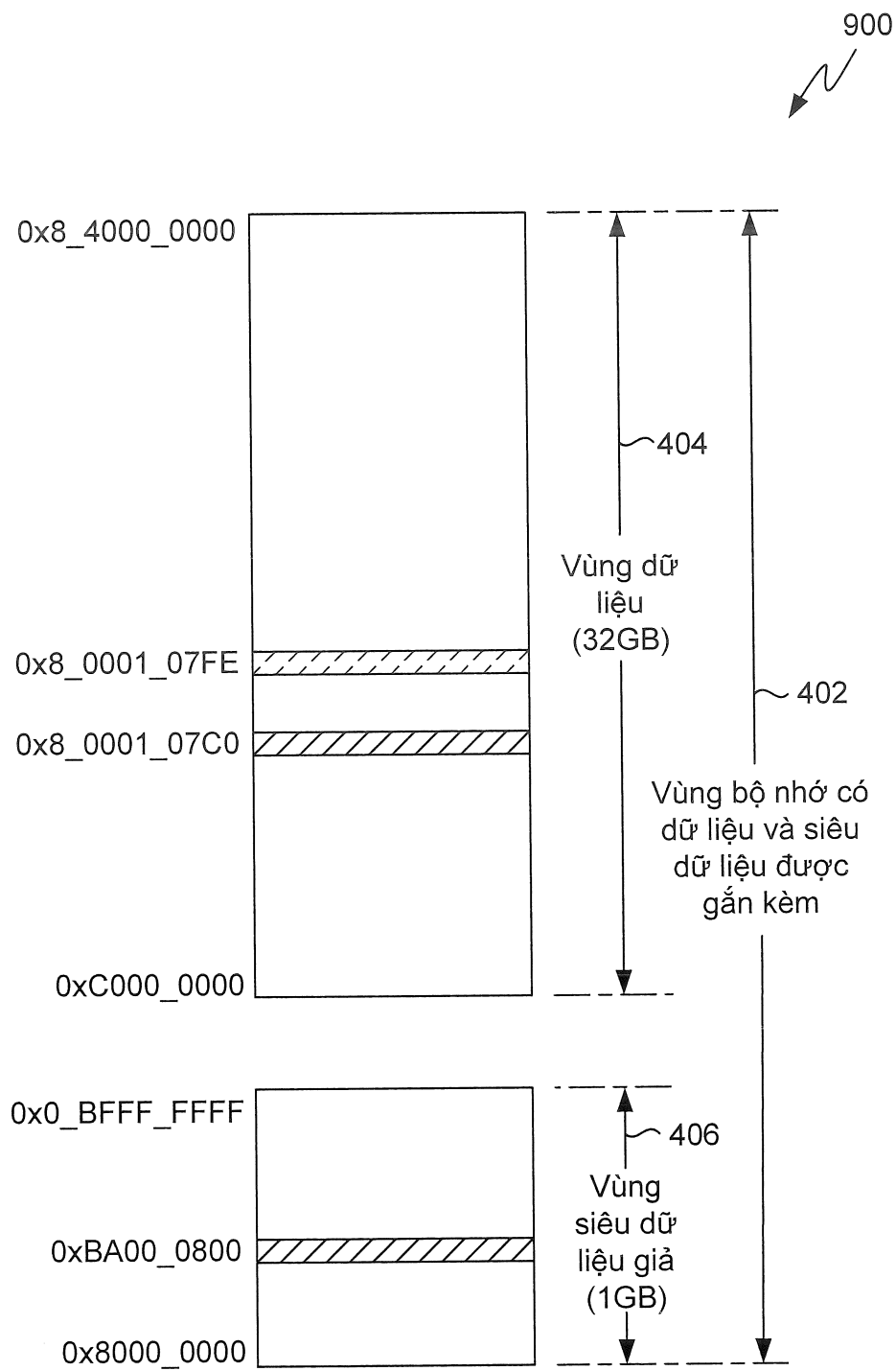


Fig. 7

**Fig. 8**

**Fig. 9**

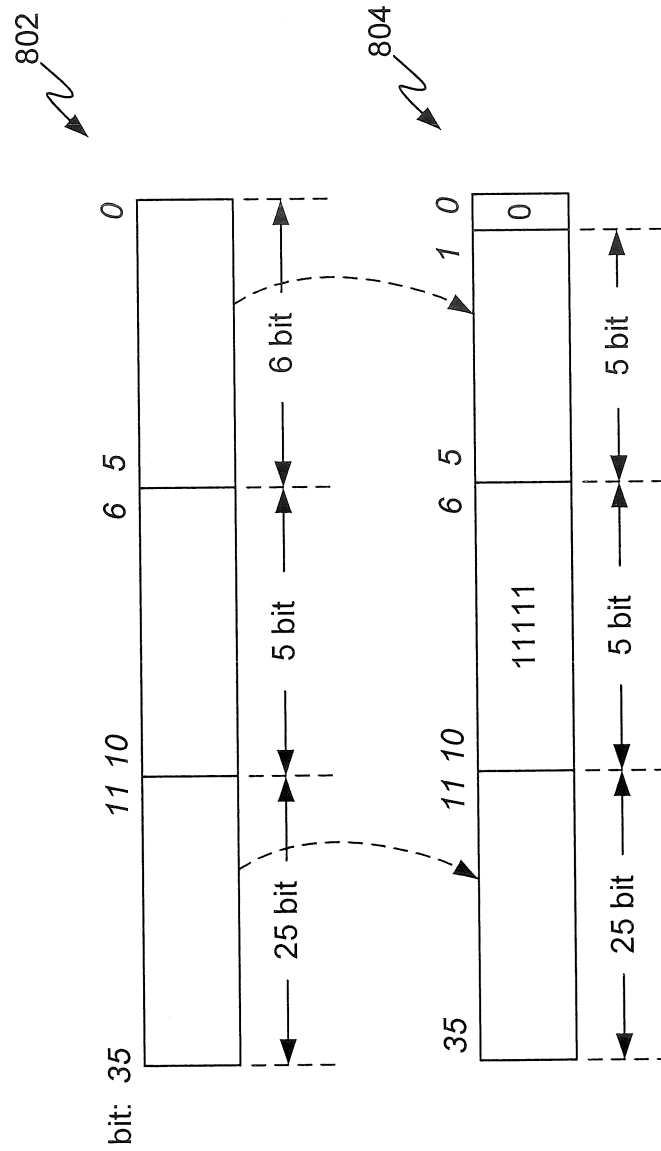
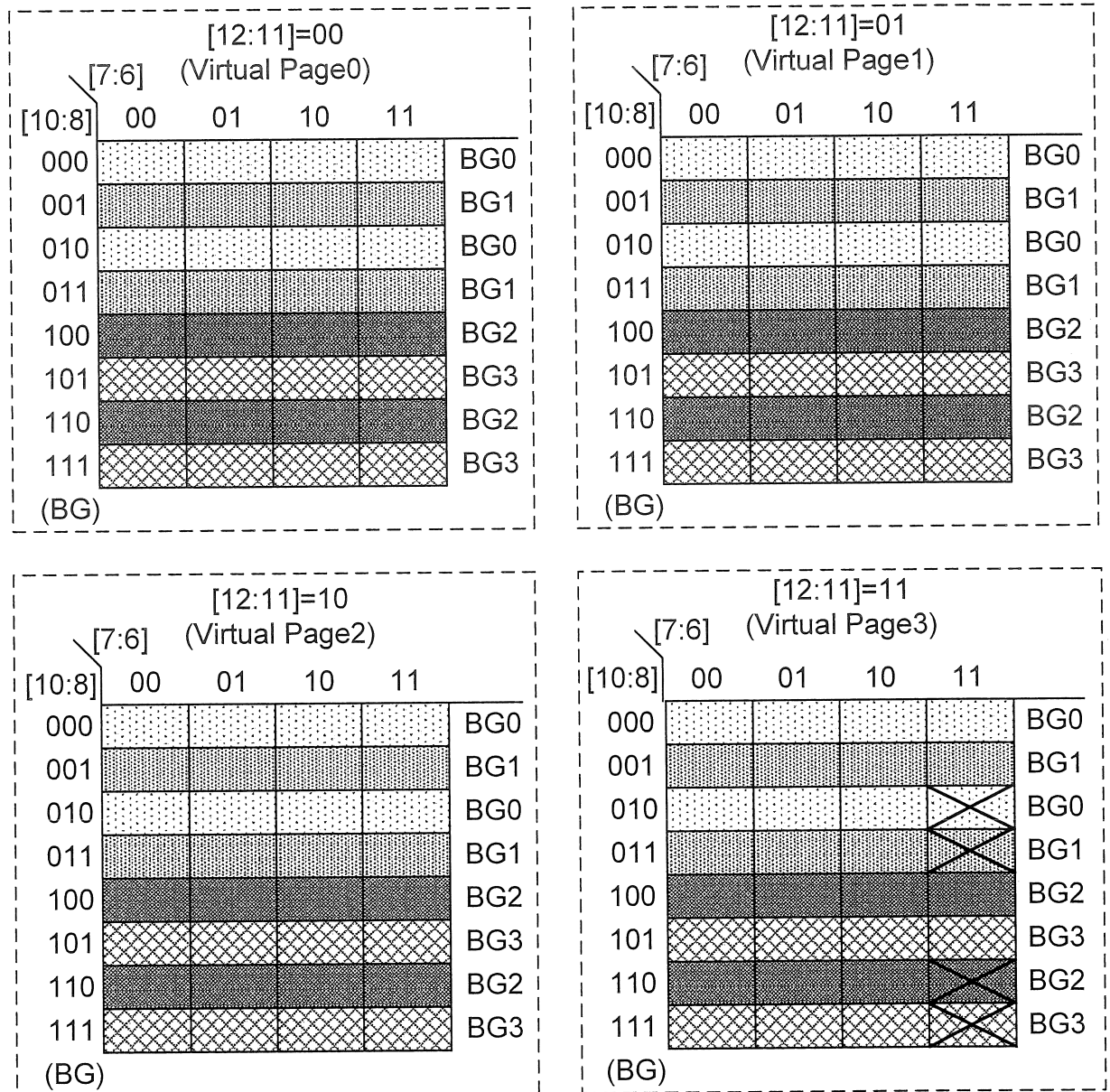


Fig. 10

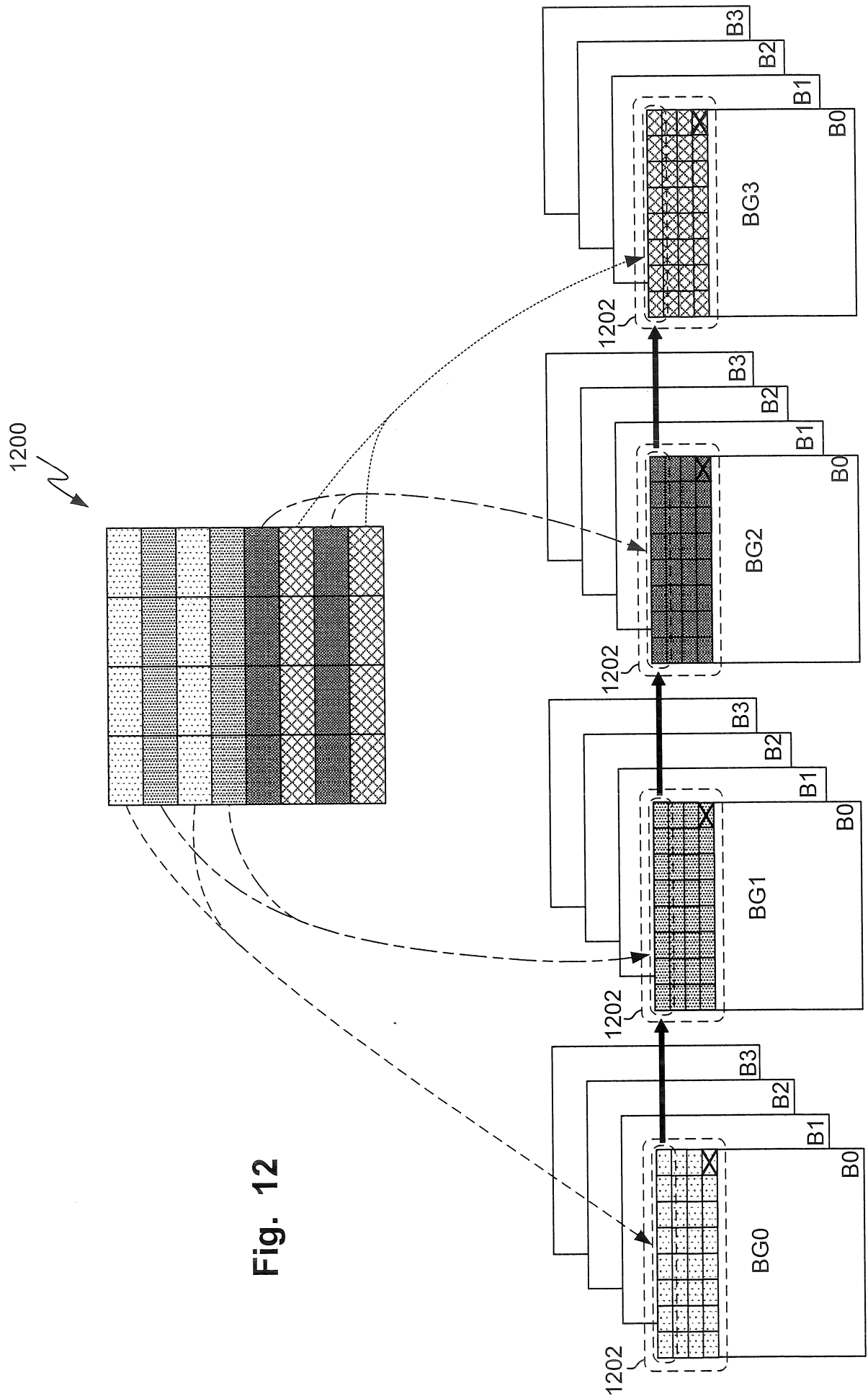
1100



Chú thích:

- | | | | |
|--|------------------------|--|-----------------------------|
| | dòng dữ liệu trong BG0 | | dòng siêu dữ liệu trong BG0 |
| | dòng dữ liệu trong BG1 | | dòng siêu dữ liệu trong BG1 |
| | dòng dữ liệu trong BG2 | | dòng siêu dữ liệu trong BG2 |
| | dòng dữ liệu trong BG3 | | dòng siêu dữ liệu trong BG3 |

Fig. 11



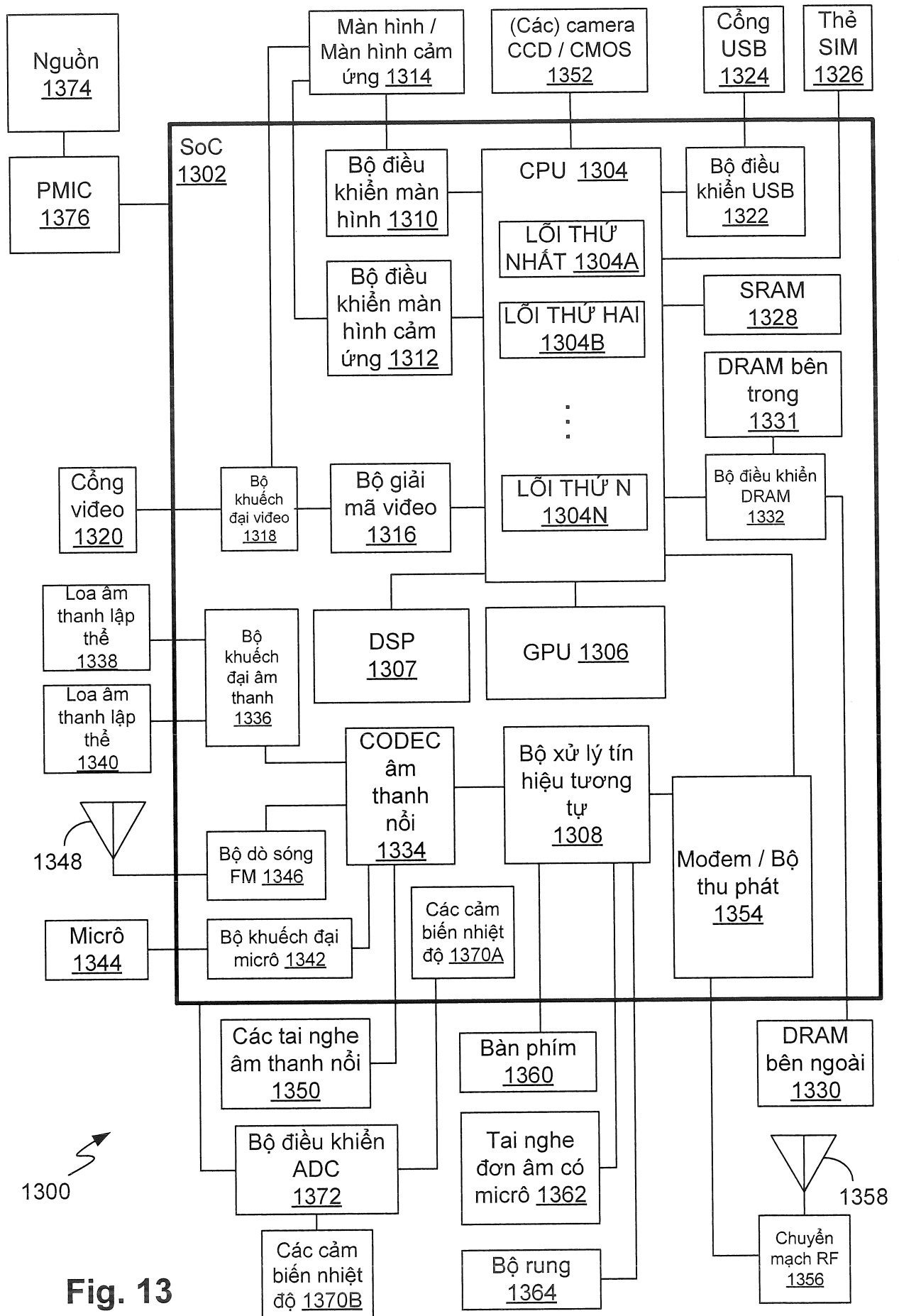


Fig. 13