



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

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

(11)



1-0039304

(51)¹⁹ G06F 9/445

(13) B

(21) 1-2019-02006

(22) 13/07/2017

(86) PCT/CN2017/092744 13/07/2017

(87) WO 2018/059076 A1 05/04/2018

(30) 201610855624.0 27/09/2016 CN

(45) 25/04/2024 433

(43) 25/06/2019 375A

(73) HUAWEI TECHNOLOGIES CO., LTD. (CN)

Huawei Administration Building, Bantian, Longgang District, Shenzhen, Guangdong
518129, China

(72) YANG, Qibin (CN); DUAN, Xiaojun (CN).

(74) Công ty TNHH một thành viên Sở hữu trí tuệ VCCI (VCCI-IP CO.,LTD)

(54) PHƯƠNG PHÁP VÀ THIẾT BỊ PHỤC HỒI BỘ NHỚ

(57) Sáng chế đề cập đến phương pháp và thiết bị phục hồi bộ nhớ. Trong đó phương pháp bao gồm các bước: khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, xác định quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nên sau, trong đó danh mục xử lý nên sau bao gồm quy trình xử lý của một hoặc nhiều ứng dụng, quy trình xử lý mà trong đó bộ nhớ cần được phục hồi là quy trình xử lý nằm trong số quy trình xử lý của một hoặc nhiều ứng dụng và đáp ứng điều kiện là trị số tuyệt đối của độ sai lệch giữa bộ nhớ được chiếm giữ bởi quy trình xử lý và trị số áp lực bộ nhớ thấp hơn ngưỡng thiết đặt trước, và trị số áp lực bộ nhớ là độ sai lệch giữa ngưỡng bộ nhớ và bộ nhớ hữu dụng hiện thời của hệ thống; và gửi lệnh xử lý tới nhân hệ thống, để kích hoạt nhân hệ thống thực hiện việc xử lý trên quy trình xử lý mà trong đó bộ nhớ cần được phục hồi, để phục hồi bộ nhớ được chiếm giữ bởi quy trình xử lý mà trong đó bộ nhớ cần được phục hồi. Bằng cách thực hiện các phương án của sáng chế, khả năng các lần dừng gián đoạn của thiết bị đầu cuối có thể được giảm.



Lĩnh vực kỹ thuật được đề cập

[0001] Sáng chế đề cập đến lĩnh vực công nghệ thông tin, và cụ thể là đến phương pháp và thiết bị phục hồi bộ nhớ.

Tình trạng kỹ thuật của sáng chế

[0002] Với sự phát triển của các thiết bị đầu cuối (chẳng hạn như các điện thoại thông minh và các máy tính bảng), các thiết bị đầu cuối đã trở thành những sản phẩm thiết yếu trong cuộc sống hàng ngày của con người. Trong thực tế, biết rằng, khi người dùng vừa mua thiết bị đầu cuối, người dùng sử dụng thiết bị đầu cuối rất trôi chảy, và các ứng dụng chạy rất nhanh. Sau khi thiết bị đầu cuối được sử dụng trong một thời gian dài, với việc gia tăng các ứng dụng được cài đặt, nhiều quy trình xử lý và các dịch vụ vô ích vẫn chạy trong nền sau, và dữ liệu quá mức được lưu trữ khi người dùng duyệt các trạng mạng và sử dụng các chương trình ứng dụng (Application, viết tắt là APP). Các chương trình này làm giảm bộ nhớ hữu dụng của hệ thống và khiến thiết bị đầu cuối bị dừng gián đoạn.

[0003] Thao tác thông thường là kiểm tra xem hệ thống có đủ bộ nhớ khả dụng khi hệ thống phân bổ bộ nhớ hay không. Nếu bộ nhớ hữu dụng hiện thời của hệ thống không thể hỗ trợ yêu cầu phân bổ bộ nhớ hiện tại, thao tác dọn sạch bộ nhớ hệ thống được bắt đầu, ví dụ, phục hồi bộ nhớ đệm hệ thống hoặc dừng quy trình xử lý. Tuy nhiên, với kết quả này, ứng dụng yêu cầu tài nguyên bộ nhớ bị khóa ở trạng thái chờ, kéo dài thời gian phản hồi của ứng dụng và bởi vậy làm tăng khả năng các lần dừng gián đoạn của thiết bị đầu cuối.

Bản chất kỹ thuật của sáng chế

[0004] Các phương án của sáng chế đề xuất phương pháp và thiết bị phục hồi bộ nhớ, để làm giảm khả năng các lần dừng gián đoạn của thiết bị đầu cuối.

[0005] Khía cạnh thứ nhất của các phương án của sáng chế bộc lộ phương pháp phục hồi bộ nhớ, phương pháp bao gồm các bước:

khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, xác định quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nền sau, trong đó danh mục xử lý nền sau bao gồm quy

trình xử lý của một hoặc nhiều ứng dụng, quy trình xử lý mà trong đó bộ nhớ cần được phục hồi là quy trình xử lý nằm trong số quy trình xử lý của một hoặc nhiều ứng dụng và đáp ứng điều kiện là trị số tuyệt đối của độ sai lệch giữa bộ nhớ được chiếm giữ bởi quy trình xử lý và trị số áp lực bộ nhớ thấp hơn ngưỡng thiết đặt trước, và trị số áp lực bộ nhớ là độ sai lệch giữa ngưỡng bộ nhớ và bộ nhớ hữu dụng hiện thời của hệ thống; và gửi lệnh xử lý tới nhân (kernel) hệ thống, để kích hoạt nhân hệ thống thực hiện việc xử lý trên quy trình xử lý mà trong đó bộ nhớ cần được phục hồi, để phục hồi bộ nhớ được chiếm giữ bởi quy trình xử lý mà trong đó bộ nhớ cần được phục hồi.

[0006] Các bước của phương pháp nêu trên có thể được thực hiện bởi một hoặc nhiều mạch điều khiển bộ nhớ, một hoặc nhiều quy trình điều khiển bộ nhớ, hoặc ứng dụng.

[0007] Danh mục xử lý nền sau là danh mục hai chiều động mà được phân loại dựa vào mức độ quan trọng ứng dụng và quyền ưu tiên xử lý, và bao gồm quy trình xử lý của một hoặc nhiều ứng dụng. Nói cách khác, danh mục xử lý nền sau có thể bao gồm một hoặc nhiều ứng dụng, và một ứng dụng có thể bao gồm một hoặc nhiều quy trình xử lý. Danh mục xử lý nền sau được tạo ra trong hai trường hợp. Trường hợp thứ nhất là danh mục xử lý nền sau được tạo ra khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ. Nói chung, trường hợp này là ứng dụng được tới các tình huống trong đó việc phục hồi bộ nhớ cần được thực hiện ngay lập tức. Trường hợp thứ hai là danh mục xử lý nền sau được tạo ra khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ và hệ thống là ở trạng thái không hoạt động. Nói chung, trường hợp này có thể ứng dụng được khi người dùng đang tương tác với thiết bị đầu cuối và việc phục hồi bộ nhớ cần được thực hiện khi hệ thống không hoạt động để tránh làm ảnh hưởng đến trải nghiệm của người dùng.

[0008] Cụ thể là, theo phương án này của sáng chế, khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, chính sách phục hồi bộ nhớ có thể được đưa ra, và quy trình xử lý mà trong đó bộ nhớ cần được phục hồi được xác định từ danh mục xử lý nền sau dựa vào chính sách phục hồi bộ nhớ. Hơn nữa, lệnh xử lý được gửi tới nhân hệ thống, để kích hoạt nhân hệ thống thực hiện việc xử lý trên quy trình xử lý mà trong đó bộ nhớ cần được

phục hồi, để phục hồi bộ nhớ tương ứng với quy trình xử lý mà trong đó bộ nhớ cần được phục hồi, cho đến khi áp lực bộ nhớ hiện thời của hệ thống được giảm bớt.

[0009] Ngoài ra, các mối quan hệ khác nhau tồn tại giữa các ứng dụng. Do đó, khi bộ nhớ được chiếm giữ bởi quy trình xử lý nền sau được phục hồi, để ngăn ngừa quy trình dọn sạch khỏi bị lấy ra bởi quy trình xử lý liên quan, việc phục hồi bộ nhớ được thực hiện nhờ sử dụng ứng dụng như một bộ phận, và bộ nhớ được chiếm giữ bởi quy trình xử lý được bao gồm trong ứng dụng được phục hồi. Bộ nhớ được chiếm giữ bởi quy trình xử lý là bộ nhớ được chiếm giữ riêng bởi quy trình xử lý, và không bao gồm bộ nhớ được dùng chung bởi quy trình xử lý và quy trình xử lý khác.

[0010] Có thể thấy được rằng theo cách này, khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, việc phục hồi bộ nhớ một cách chủ động có thể được thực hiện. Khi chương trình ứng dụng thực sự cần đến một lượng lớn bộ nhớ, nhân hệ thống đã phục hồi một lượng lớn của bộ nhớ khả dụng. Do đó, khả năng các lần dừng gián đoạn của thiết bị đầu cuối có thể được giảm.

[0011] Theo cách thực hiện có thể, thao tác xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ được kích hoạt khi sự kiện chính thứ nhất được phát hiện, và ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ nhất tương ứng với sự kiện chính thứ nhất. Ngưỡng bộ nhớ có thể được thiết đặt trước ở dạng tĩnh, hoặc có thể được tạo cấu hình ở dạng động. Các sự kiện chính thứ nhất khác nhau tương ứng với các tình huống khác nhau, và các ngưỡng bộ nhớ thứ nhất tương ứng cũng khác nhau.

[0012] Theo cách thực hiện có thể, sự kiện chính thứ nhất bao gồm bất kỳ một trong số các sự kiện sau đây: sự kiện bắt đầu khởi động chương trình, sự kiện dọn sạch, và sự kiện tràn bộ nhớ OOM. Khi sự kiện chính thứ nhất xảy ra, việc phục hồi bộ nhớ cần được thực hiện ngay lập tức.

[0013] Theo cách thực hiện có thể, thao tác xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ được kích hoạt khi sự kiện chính thứ hai được phát hiện và khi xác định được rằng hệ thống là ở trạng thái không hoạt động, và ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ hai tương ứng với sự kiện chính thứ hai. Ngưỡng bộ nhớ có thể được thiết đặt trước ở dạng tĩnh, hoặc

có thể được tạo cấu hình ở dạng động. Các sự kiện chính thứ hai khác nhau tương ứng với các tình huống khác nhau, và các ngưỡng bộ nhớ thứ hai tương ứng cũng khác nhau.

[0014] Theo cách thực hiện này, thứ tự thời gian của các bước là: khi sự kiện chính thứ hai được phát hiện, đầu tiên được xác định xem hệ thống là ở trạng thái không hoạt động hay không; khi xác định được rằng hệ thống là ở trạng thái không hoạt động, bộ nhớ hữu dụng hiện thời của hệ thống được đọc; khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, quy trình mà trong đó bộ nhớ cần được phục hồi được xác định từ danh mục xử lý nền sau; hơn nữa, lệnh xử lý được gửi tới nhân hệ thống để kích hoạt nhân hệ thống thực hiện việc xử lý trên quy trình xử lý mà trong đó bộ nhớ cần được phục hồi, để phục hồi bộ nhớ tương ứng với quy trình xử lý mà trong đó bộ nhớ cần được phục hồi.

[0015] Theo cách thực hiện có thể, thao tác xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ được kích hoạt khi sự kiện chính thứ hai được phát hiện, thao tác xác định quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nền sau được kích hoạt khi xác định được rằng hệ thống là ở trạng thái không hoạt động, và ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ hai tương ứng với sự kiện chính thứ hai.

[0016] Theo cách thực hiện này, thứ tự thời gian của các bước là: khi sự kiện chính thứ hai được phát hiện, bộ nhớ hữu dụng hiện thời của hệ thống được đọc trước tiên; sau đó khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, được xác định xem hệ thống là ở trạng thái không hoạt động hay không; và khi xác định được rằng hệ thống là ở trạng thái không hoạt động, quy trình xử lý mà trong đó bộ nhớ cần được phục hồi được xác định từ danh mục xử lý nền sau; và hơn nữa, lệnh xử lý được gửi tới nhân hệ thống để kích hoạt nhân hệ thống thực hiện việc xử lý trên quy trình xử lý mà trong đó bộ nhớ cần được phục hồi, để phục hồi bộ nhớ tương ứng với quy trình xử lý mà trong đó bộ nhớ cần được phục hồi.

[0017] Theo cách thực hiện có thể, sự kiện chính thứ hai bao gồm bất kỳ một trong số các sự kiện sau đây: sự kiện hoàn thành khởi động chương trình, sự kiện bật màn hình, sự kiện tắt màn hình, sự kiện chạm màn hình, sự kiện chuyển đổi giao diện người dùng, sự kiện kết thúc chuyển đổi thao tác, và sự kiện phát

rộng.

[0018] Theo cách thực hiện có thể, việc xác định rằng hệ thống là ở trạng thái không hoạt động bao gồm bước:

xác định xem tải hiện thời của hệ thống thấp hơn ngưỡng tải hay không; và khi tải hiện thời của hệ thống thấp hơn ngưỡng tải, xác định rằng hệ thống là ở trạng thái không hoạt động.

[0019] Có hai tình huống trong đó hệ thống là ở trạng thái không hoạt động. Tình huống thứ nhất là thiết bị đầu cuối đang không được sử dụng bởi người dùng. Ví dụ, thiết bị đầu cuối có thể được xét đến là ở trạng thái không hoạt động khi thiết bị đầu cuối đang ở chế độ chờ. Đối với ví dụ khác, khi việc khởi động của ứng dụng được kết thúc và màn hình hiển thị xuất hiện, nhưng sau đó không có sự tương tác với thiết bị đầu cuối, trong trường hợp này, hệ thống không có số lượng lớn các thao tác IO và có thể được xét đến là ở trạng thái không hoạt động. Tình huống thứ hai là hệ thống có thể được xét đến là ở trạng thái không hoạt động khi thiết bị đầu cuối đang được sử dụng bởi người dùng nhưng tải hệ thống hiện thời không ảnh hưởng đến trải nghiệm của người dùng, ví dụ, khi tải hệ thống thấp hơn ngưỡng tải.

[0020] Theo cách thực hiện có thể, khi các quy trình xử lý mà trong đó bộ nhớ cần được phục hồi được xác định, gửi lệnh xử lý tới nhân hệ thống bao gồm:

gọi các luồng để gửi các lệnh xử lý tới nhân hệ thống, trong đó mỗi luồng được sử dụng để gửi một hoặc nhiều lệnh xử lý.

[0021] Có thể có một hoặc nhiều quy trình xử lý được xác định mà trong đó bộ nhớ cần được phục hồi. Khi một quy trình xử lý yêu cầu gửi nhiều lệnh xử lý, các lệnh xử lý có thể được gửi liên tiếp từng lệnh một đối với quy trình xử lý, mà không cần kiểm tra, sau khi gửi một lệnh xử lý, trạng thái của bộ nhớ tương ứng với quy trình xử lý mà trong đó bộ nhớ được phục hồi bởi nhân hệ thống trước khi gửi lệnh xử lý khác.

[0022] Theo ứng dụng tùy chọn này, khi các quy trình xử lý mà trong đó bộ nhớ cần được phục hồi được xác định, các luồng có thể được gọi để gửi các lệnh xử lý tới nhân hệ thống. Cụ thể là, một luồng gửi nhiều lệnh xử lý nghĩa là luồng gửi các lệnh xử lý liên tiếp từng lệnh một, mà không cần kiểm tra, sau khi

gửi một lệnh xử lý, trạng thái của bộ nhớ tương ứng với quy trình xử lý mà trong đó bộ nhớ được phục hồi bởi nhân hệ thống trước khi gửi lệnh xử lý khác. Mỗi lệnh xử lý mang ký hiệu nhận dạng của một quy trình xử lý cần được phục hồi.

[0023] Có thể thấy được rằng cách này có thể nâng cao hiệu quả của việc gửi các lệnh xử lý tới nhân hệ thống bởi luồng, và có thể cũng cải thiện hiệu suất phục hồi bộ nhớ của nhân hệ thống.

[0024] Theo cách thực hiện có thể, việc xác định quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nền sau bao gồm bước:

xác định, dựa vào thứ tự tăng dần về mức độ quan trọng của ứng dụng, ít nhất một ứng dụng từ các ứng dụng được bao gồm trong danh mục xử lý nền sau; và xác định, dựa vào thứ tự tăng dần của các quyền ưu tiên xử lý, quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ các quy trình xử lý được bao gồm trong ít nhất một ứng dụng.

[0025] Theo cách thực hiện có thể, phương pháp còn bao gồm bước:

tạo danh mục xử lý nền sau khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ. Trường hợp này là ứng dụng được tới các tình huống trong đó việc phục hồi bộ nhớ cần được thực hiện ngay lập tức.

[0026] Theo cách thực hiện có thể, phương pháp còn bao gồm bước:

tạo danh mục xử lý nền sau khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ và hệ thống là ở trạng thái không hoạt động. Trường hợp này là ứng dụng được tới tình huống trong đó người dùng đang tương tác với thiết bị đầu cuối, và việc phục hồi bộ nhớ cần được thực hiện khi hệ thống không hoạt động để tránh khỏi việc ảnh hưởng đến trải nghiệm của người dùng.

[0027] Theo cách thực hiện có thể, việc tạo danh mục xử lý nền sau bao gồm:

xác định chỉ số của thành phần chính của mỗi ứng dụng hiện đang chạy trong nền sau, trong đó thành phần chính bao gồm một hoặc nhiều trong số các thành phần sau: quyền ưu tiên xử lý, thói quen sử dụng của người dùng, tài nguyên hệ thống được chiếm giữ bởi quy trình xử lý, và mối liên quan ứng dụng; thực hiện phép tính trọng số trên các chỉ số của tất cả các thành phần chính của

mỗi ứng dụng, để nhận được mức độ quan trọng của ứng dụng; phân loại tất cả các ứng dụng dựa vào các mức độ quan trọng của tất cả các ứng dụng; và phân loại, dựa vào các quyền ưu tiên xử lý, các quy trình xử lý được bao gồm trong mỗi ứng dụng được phân loại, để tạo ra danh mục xử lý nền sau.

[0028] Tài nguyên hệ thống có thể bao gồm nhưng không giới hạn ở tài nguyên tính toán, tài nguyên lưu trữ, tài nguyên CPU, và tài nguyên IO. Khi danh mục xử lý nền sau được tạo ra dựa vào các thành phần chính (quyền ưu tiên xử lý, thói quen sử dụng của người dùng, tài nguyên hệ thống được chiếm giữ bởi quy trình xử lý, và mối liên quan ứng dụng) của các ứng dụng, quy trình xử lý mà nó yêu cầu việc xử lý có thể sau đó được xác định từ danh mục xử lý nền sau dựa vào yêu cầu bộ nhớ. Theo cách này, trình tự xử lý có thể loại bỏ có thể được lựa chọn một cách chính xác, làm giảm khả năng loại bỏ lỗi/quá mức/thiếu hụt của các quy trình xử lý.

[0029] Khía cạnh thứ hai của các phương án của sáng chế bộc lộ thiết bị phục hồi bộ nhớ, thiết bị bao gồm các bộ phận chức năng được tạo cấu hình để thực hiện một vài hoặc tất cả các bước của phương pháp bất kỳ theo khía cạnh thứ nhất của các phương án của sáng chế. Khi thiết bị đầu cuối thực hiện một vài hoặc tất cả các bước của phương pháp bất kỳ theo khía cạnh thứ nhất, khả năng các lần dừng gián đoạn của thiết bị đầu cuối có thể được giảm.

[0030] Khía cạnh thứ ba của các phương án của sáng chế bộc lộ thiết bị đầu cuối, thiết bị bao gồm bộ xử lý và bộ nhớ, trong đó bộ nhớ được tạo cấu hình để lưu trữ lệnh, bộ xử lý được tạo cấu hình để thực hiện lệnh, và bộ xử lý thực hiện lệnh để thực hiện một vài hoặc tất cả các bước của phương pháp bất kỳ theo khía cạnh thứ nhất của các phương án của sáng chế. Khi thiết bị đầu cuối thực hiện một vài hoặc tất cả các bước của phương pháp bất kỳ theo khía cạnh thứ nhất, khả năng các lần dừng gián đoạn của thiết bị đầu cuối có thể được giảm.

[0031] Khía cạnh thứ tư của các phương án của sáng chế bộc lộ vật ghi đọc được bởi máy tính, trong đó vật ghi đọc được bởi máy tính lưu trữ chương trình, và chương trình cụ thể bao gồm lệnh được sử dụng để thực hiện một vài hoặc tất cả các bước của phương pháp bất kỳ theo khía cạnh thứ nhất của các phương án của sáng chế.

Mô tả văn tắt các hình vẽ

[0032] Để mô tả các giải pháp kỹ thuật trong các phương án của sáng chế một cách rõ ràng hơn, phần dưới đây mô tả vắn tắt các hình vẽ kèm theo cần cho việc mô tả các phương án. Rõ ràng là, các hình vẽ kèm theo trong phần mô tả dưới đây chỉ thể hiện một vài phương án của sáng chế, và người có trình độ trung bình trong lĩnh vực kỹ thuật tương ứng có thể dẫn ra các hình vẽ khác từ các hình vẽ kèm theo này mà không cần các sự nỗ lực sáng tạo.

[0033] Fig.1 là sơ đồ cấu trúc giản lược của hệ điều hành được bộc lộ trong một phương án của sáng chế;

[0034] Fig.2 là lưu đồ giản lược của phương pháp phục hồi bộ nhớ được bộc lộ trong một phương án của sáng chế;

[0035] Fig.2A là hình vẽ giản lược của việc nhận dạng mức độ quan trọng ứng dụng được bộc lộ trong một phương án của sáng chế;

[0036] Fig.2B là hình vẽ giản lược của mối liên quan ứng dụng được bộc lộ trong một phương án của sáng chế;

[0037] Fig.2C là hình vẽ của trình tự khởi động camera được bộc lộ trong một phương án của sáng chế;

[0038] Fig.2D là hình vẽ của trình tự khởi động camera khác được bộc lộ trong một phương án của sáng chế;

[0039] Fig.3 là lưu đồ giản lược của phương pháp phục hồi bộ nhớ khác được bộc lộ trong một phương án của sáng chế;

[0040] Fig.4 là lưu đồ giản lược của phương pháp phục hồi bộ nhớ khác được bộc lộ trong một phương án của sáng chế;

[0041] Fig.5 là sơ đồ cấu trúc giản lược của thiết bị phục hồi bộ nhớ được bộc lộ trong một phương án của sáng chế; và

[0042] Fig.6 là sơ đồ cấu trúc giản lược của thiết bị đầu cuối được bộc lộ trong một phương án của sáng chế.

Mô tả chi tiết sáng chế

[0043] Phần dưới đây mô tả rõ ràng và đầy đủ các giải pháp kỹ thuật trong các phương án của sáng chế dựa vào các hình vẽ kèm theo trong các phương án của sáng chế. Rõ ràng là, các phương án được mô tả chỉ là một vài mà không phải tất cả các phương án của sáng chế. Tất cả các phương án khác đạt được bởi

người có trình độ trung bình trong lĩnh vực kỹ thuật tương ứng dựa vào các phương án của sáng chế mà không có các sự nỗ lực sáng tạo sẽ nằm trong phạm vi bảo hộ của sáng chế.

[0044] Thuật ngữ "bao gồm", hoặc bất kỳ biến thể khác của nó được nêu trong bản mô tả, yêu cầu bảo hộ, và các hình vẽ kèm theo của sáng chế, được dự định bao hàm sự bao gồm không loại trừ. Ví dụ, quy trình xử lý, phương pháp, hệ thống, sản phẩm, hoặc thiết bị mà nó bao gồm hàng loạt các bước hoặc các bộ phận là không giới hạn ở các bước hoặc các bộ phận được liệt kê, mà một cách tùy ý còn bao gồm bước hoặc bộ phận không được liệt kê, hoặc một cách tùy ý còn bao gồm bước hoặc bộ phận kế thừa khác của quy trình xử lý, phương pháp, sản phẩm, hoặc thiết bị.

[0045] Các phương án của sáng chế bộc lộ phương pháp và thiết bị phục hồi bộ nhớ, để làm giảm khả năng các lần dừng gián đoạn của thiết bị đầu cuối. Dưới đây đưa ra chi tiết các phần mô tả một cách riêng biệt.

[0046] Phương pháp phục hồi bộ nhớ được đề xuất bởi các phương án của sáng chế chủ yếu được áp dụng cho thiết bị đầu cuối. Thiết bị đầu cuối có thể được gọi là thiết bị người dùng (User Equipment, viết tắt là UE), trạm di động (Mobile Station, viết tắt là MS), thiết bị đầu cuối di động (Mobile Terminal), hoặc tương tự. Một cách tùy ý, thiết bị đầu cuối có thể có khả năng truyền thông với một hoặc nhiều mạng lõi nhờ sử dụng mạng truy cập radio (Radio Access Network, viết tắt là RAN). Ví dụ, thiết bị đầu cuối có thể là điện thoại di động (cũng được gọi là điện thoại tế bào "cellular") hoặc máy tính có tính năng di động. Ví dụ, thiết bị đầu cuối theo cách khác có thể là thiết bị di động xách tay, thiết bị di động loại bỏ túi, thiết bị di động cầm tay, thiết bị di động lắp trong máy tính, hoặc thiết bị di động trên xe. Hệ điều hành của thiết bị đầu cuối có thể bao gồm nhưng không giới hạn ở hệ điều hành Android (Android), hệ điều hành iOS, hệ điều hành Symbian (Symbian), hệ điều hành BlackBerry (BlackBerry), hệ điều hành Windows Phone 8, hoặc tương tự.

[0047] Phương pháp phục hồi bộ nhớ được đề xuất bởi các phương án của sáng chế chủ yếu được áp dụng cho thiết bị đầu cuối với hệ điều hành Android.

[0048] Nhằm hiểu rõ hơn về các phương án của sáng chế, phần dưới đây đầu tiên mô tả sơ đồ cấu trúc gián lược của hệ điều hành được bộc lộ trong một phương án của sáng chế.

[0049] Dựa vào Fig.1, Fig.1 là sơ đồ cấu trúc giản lược của hệ điều hành được bộc lộ trong một phương án của sáng chế. Cụ thể là, hệ điều hành có thể là hệ điều hành Android. Hệ điều hành Android là hệ điều hành nguồn mở được phát triển bởi Google dựa vào nền sau Linux. Hệ điều hành Android là môi trường tạo lớp được xây dựng trên nhân Linux, và bao gồm các chức năng phong phú.

[0050] Như được thể hiện trên Fig.1, hệ điều hành 100 bao gồm lớp ứng dụng 110, lớp khung 120, lớp thư viện lõi 130, và lớp nhân 140.

[0051] Lớp ứng dụng 110 bao gồm tất cả các chương trình ứng dụng được yêu cầu khi thiết bị đầu cuối làm việc, ví dụ, trình duyệt thư điện tử (email), dịch vụ tin nhắn ngắn, lịch, sơ đồ, và trình duyệt. Các chương trình ứng dụng này có thể được viết nhờ sử dụng ngôn ngữ JAVA hoặc C/C++ chẳng hạn.

[0052] Lớp khung 120 là khung chương trình ứng dụng Android, và các chương trình ứng dụng Android tất cả đều được phát triển dựa vào khung chương trình. Khung chương trình cho phép người lập trình phát triển các chương trình ứng dụng mới và phong phú bằng cách cung cấp tập hợp các thư viện lớp cơ bản. Dựa vào các thư viện lớp, người lập trình có thể dễ dàng ứng dụng lợi thế phần cứng của thiết bị để thực hiện các chức năng, ví dụ, truy cập thông tin vị trí, chạy dịch vụ nền sau, thiết đặt đồng hồ cảnh báo, và bổ sung thông báo vào thanh trạng thái. Cần lưu ý rằng, lớp khung 120 được thể hiện trên Fig.1 giữ các môđun chức năng của lớp khung gốc của hệ điều hành Android. Ngoài ra, một vài sự điều chỉnh được thực hiện tới dịch vụ hệ thống 121. Như được thể hiện trên Fig.1, dịch vụ hệ thống 121 bao gồm môđun hệ thống quản lý tài nguyên (Resource Management System, viết tắt là RMS) 1211, môđun điều khiển ứng dụng 1212, và môđun tập hợp trạng thái tài nguyên hệ thống 1213. Môđun RMS 1211 chủ yếu được tạo cấu hình để quản lý thông tin được tập hợp bởi môđun tập hợp trạng thái tài nguyên hệ thống 1213, môđun điều khiển ứng dụng 1212 chủ yếu được tạo cấu hình cho việc xử lý nhanh của quy trình xử lý, và môđun tập hợp trạng thái tài nguyên hệ thống 1213 chủ yếu được tạo cấu hình để tập hợp và nhận dạng trạng thái tài nguyên hệ thống. Ngoài ra, như được thể hiện trên Fig.1, quy trình nhận biết 122 được bổ sung vào lớp khung 120. Quy trình nhận biết 122 bao gồm môđun RMS 1221, môđun chương trình hỗ trợ (wizard) tiết kiệm công suất 1222, và môđun điều khiển bộ

nhớ 1223. Môđun RMS 1221 chủ yếu được tạo cấu hình để xử lý cách thức điều khiển bộ nhớ mà nó chiếm thời gian thực hiện dài và thực hiện việc nhận dạng bật/tắt màn hình. Môđun chương trình hỗ trợ (wizard) tiết kiệm công suất 1222 chủ yếu được tạo cấu hình để nhận dạng trạng thái thao tác ứng dụng và xác định xem lượng pin được tiêu thụ bởi ứng dụng vượt quá ngưỡng pin cho phép hay không, để thực hiện việc xử lý điều khiển (ví dụ, ngắt quy trình xử lý, hạn chế khởi động ứng dụng, hoặc trì hoãn sự phân phối tin nhắn). Môđun điều khiển bộ nhớ 1223 chủ yếu được tạo cấu hình để quản lý và xử lý giao diện môđun ngoại vi, nhận dạng trạng thái tải/không làm việc của hệ thống, quản lý cấu hình của ngưỡng bộ nhớ không làm việc, nhận dạng trạng thái áp lực bộ nhớ, đề ra chính sách phục hồi bộ nhớ được chất lọc, thực hiện việc phân bổ bộ nhớ thông minh, và quản lý việc bảo dưỡng và kiểm thử.

[0053] Lớp thư viện lõi 130 chủ yếu cung cấp các thành phần của lớp khung 120 với tập hợp của các thư viện C/C++ (nằm dưới các thư viện dùng cho khung ứng dụng) dùng cho các sự thực hiện cơ bản của các thành phần. Lớp thư viện lõi 130 bao gồm trình nền (daemon) thu 131, và trình nền thu 131 chủ yếu được tạo cấu hình để tập hợp thông tin trạng thái tài nguyên hệ thống.

[0054] Lớp nhân 140 được bố trí ở lớp đáy của hệ điều hành Android, và có thể được xét đến như trừu tượng trực tiếp của lớp phần cứng và các nhóm phần mềm khác trong hệ thống. Lớp nhân 140 chủ yếu cung cấp hệ điều hành Android với các dịch vụ hệ thống lõi, bao gồm cơ chế bảo vệ, quản lý bộ nhớ, quản lý quy trình xử lý, quản lý tệp tin, quản lý truyền thông mạng, và quản lý ổ đĩa. Ngoài ra, theo phương án này của sáng chế, lớp nhân 140 theo cách khác có thể được tạo cấu hình cho sự phục hồi bộ nhớ đệm ở mức xử lý.

[0055] Dựa vào Fig.2, Fig.2 là lưu đồ giản lược của phương pháp phục hồi bộ nhớ được bộc lộ trong một phương án của sáng chế. Phương pháp có thể được áp dụng tới môđun điều khiển bộ nhớ 1223 được thể hiện trên Fig.1. Cụ thể là, môđun điều khiển bộ nhớ 1223 có thể là một hoặc nhiều luồng, hoặc môđun điều khiển bộ nhớ 1223 có thể là một hoặc nhiều quy trình xử lý. Như được thể hiện trên Fig.2, phương pháp có thể bao gồm các bước sau.

[0056] 201: Xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ khi sự kiện chính thứ nhất được phát hiện.

[0057] Theo phương án này của sáng chế, môđun điều khiển bộ nhớ 1223 có

thể phát hiện sự kiện chính của chương trình ứng dụng. Cụ thể là, thông tin chính (ví dụ, tên quy trình xử lý, ký hiệu nhận dạng người dùng (User Identifier, viết tắt là UID) và ký hiệu nhận dạng quy trình xử lý (Process Identifier, viết tắt là PID)) của chương trình ứng dụng có thể nhận được nhờ sử dụng chức năng chặn (hook) khi dịch vụ quản lý hoạt động (Activity Manager Service, viết tắt là AMS) bắt đầu sự kích hoạt (activity) của mỗi ứng dụng. Hơn nữa, AMS có thể gửi thông tin chính nhận được của chương trình ứng dụng tới môđun điều khiển bộ nhớ 1223. Do đó, môđun điều khiển bộ nhớ 1223 có thể phát hiện sự kiện chính của chương trình ứng dụng.

[0058] Các sự kiện chính của chương trình ứng dụng có thể bao gồm, nhưng không giới hạn ở, sự kiện bắt đầu khởi động chương trình, sự kiện dọn sạch, tràn bộ nhớ (Out Of Memory, viết tắt là OOM), sự kiện hoàn thành khởi động chương trình, sự kiện bật màn hình, sự kiện tắt màn hình, sự kiện chạm màn hình, giao diện chuyển đổi giao diện người dùng, sự kiện kết thúc chuyển đổi thao tác, sự kiện phát rộng, sự kiện thay đổi trạng thái/quét/kết nối mạng (ví dụ, Wi-Fi, 2/3/4G, Bluetooth, hồng ngoại, hệ thống định vị toàn cầu (Global Positioning Hệ thống, viết tắt là GPS), phát rộng radio, hoặc truyền thông trường gần (Near Field Communication, viết tắt là NFC)), sự kiện thay đổi kết nối ngoại vi (ví dụ, lắp/tháo bộ nạp), sự kiện báo hiệu/cảnh báo, sự kiện gọi điện thoại (cuộc gọi đến, thay đổi tín hiệu, hoặc tương tự), sự kiện đa phương tiện (thay đổi audio/video, hoặc phát audio được ghi), sự kiện liên quan đến đèn chớp, sự kiện lắp đặt/gỡ bỏ chương trình, sự kiện mức pin, và tương tự.

[0059] Theo phương án này của sáng chế, sự kiện mà yêu cầu việc phục hồi bộ nhớ ngay lập tức có thể được định rõ trước như sự kiện chính thứ nhất, và sự kiện chính thứ nhất có thể bao gồm nhưng không giới hạn ở sự kiện bắt đầu khởi động chương trình, sự kiện dọn sạch, và sự kiện OOM. Sự kiện bắt đầu khởi động chương trình chủ yếu là sự kiện chương trình ứng dụng được kích hoạt bởi người dùng mà yêu cầu khởi động ngay lập tức (ví dụ, người dùng gỡ ứng dụng WeChat). Sự kiện dọn sạch có thể bao gồm, nhưng không giới hạn ở, sự kiện dọn sạch bộ quản lý thiết bị đầu cuối, sự kiện dọn sạch một chạm, sự kiện gỡ bỏ ứng dụng, và sự kiện dọn sạch bộ nhớ đệm ứng dụng. Sự kiện OOM có thể bao gồm, nhưng không giới hạn ở, sự kiện mà bộ nhớ được yêu cầu để khởi động ứng dụng bộ nhớ lớn vượt quá ngưỡng bộ nhớ, sự kiện hệ thống OOM, và sự kiện dừng ứng dụng cưỡng bức.

[0060] Theo phương án này của sáng chế, hệ thống bộ nhớ có thể bao gồm: bộ nhớ được chiếm giữ và bộ nhớ khả dụng. Bộ nhớ được chiếm giữ chủ yếu là bộ nhớ được chiếm giữ bởi dịch vụ và quy trình xử lý hiện đang chạy quy trình xử lý và bộ nhớ được chiếm giữ bởi sự điều hành nhân. Bộ nhớ khả dụng là bộ nhớ không làm việc và bộ nhớ đệm của hệ thống.

[0061] Khi sự kiện chính thứ nhất được phát hiện, bộ nhớ hữu dụng hiện thời của hệ thống có thể nhận được, và được xác định xem bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ hay không. Khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, nó chỉ báo rằng áp lực bộ nhớ hiện có trong hệ thống và việc phục hồi bộ nhớ cần được thực hiện. Ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ nhất tương ứng với sự kiện chính thứ nhất.

[0062] Một cách tùy ý, ngưỡng bộ nhớ có thể được thiết đặt trước ở dạng tĩnh. Cụ thể là, có một vài trường hợp như sau:

[0063] (1) Tạo cấu hình ngưỡng bộ nhớ dựa vào tính năng sản phẩm.

[0064] Đối với các sản phẩm bộ nhớ truy cập ngẫu nhiên 2G, 3G, 4G, và 6G (Random Access Memory, viết tắt là RAM), bộ nhớ khả dụng của hệ thống là khác nhau, và tương ứng, ngưỡng bộ nhớ là cũng khác nhau. Ví dụ, đối với sản phẩm RAM 3G, ngưỡng bộ nhớ có thể được tạo cấu hình là 600 MB; đối với sản phẩm RAM 4G, ngưỡng bộ nhớ có thể được tạo cấu hình là 700 MB.

[0065] (2) Tạo cấu hình ngưỡng bộ nhớ nhỏ nhất có thể dựa vào tình huống ứng dụng.

[0066] Đối với các ứng dụng khác nhau và các tình huống khác nhau, sức tiêu thụ bộ nhớ là khác nhau. Do đó, nếu cách thức trong mục (1) được sử dụng để tạo cấu hình ngưỡng bộ nhớ thích hợp, vẫn có một vài nhược điểm. Có thể là ứng dụng bên thứ ba ngừng gián đoạn trong khi khởi động bởi vì ứng dụng không được xét đến. Đối với vấn đề này, các ứng dụng bên thứ ba hạng cao nhất trong thị trường ứng dụng có thể được phân tích ngoại tuyến, để nhận được thông tin về bộ nhớ lớn nhất được chiếm giữ bởi các ứng dụng khác nhau trong các tình huống khác nhau.

[0067] Dựa vào Bảng 1, Bảng 1 thể hiện thông tin về bộ nhớ lớn nhất được chiếm giữ bởi các ứng dụng bên thứ ba trong các tình huống điển hình. Cần lưu

ý rằng Bảng 1 chỉ liệt kê một vài ứng dụng làm các ví dụ.

Bảng 1

Ứng dụng	Tình huống ứng dụng	Bộ nhớ được chiếm giữ (RAM: KB)			Trị số trung bình (KB)
Video QQ	Xem đăng ký hàng ngày	554786	536644	502018	531149
Didi Chuxing	Tìm kiếm	468643	449101	390822	436189
Taobao	Vuốt nhanh trang chủ Tmall	468147	403831	394853	422277
Thời tiết Moji	Đưa ra timescape	380951	428168	424905	411341
Youku	Gõ vào màn hình chính để nhập trang hội viên Youku	464924	439859	255841	386875
Thời tiết Moji	Kiểm tra xem có gì mới	378418	348067	423343	383276
Trình duyệt UC	Vuốt nhanh trang Sina Weibo	348390	437752	324717	370286
Bản đồ Baidu	Gõ "Navigation (điều hướng)" để	505256	368885	226482	366874

Ứng dụng	Tình huống ứng dụng	Bộ nhớ được chiếm giữ (RAM: KB)			Trị số trung bình (KB)
	nhập trang điều hướng				
Các tin chính trong ngày	Các giao dịch trong ngày mua sắm của tôi (My-Mall-Today's Deals)	340781	387801	360222	362935
Những yếu tố vui vẻ (Happy Elements)	Đăng ký	352564	352913	353630	353036

[0068] Cụ thể là, khi thiết bị đầu cuối hoạt động và sự kiện hoàn thành khởi động (BOOT_COMPLETED) thu được, các ký hiệu nhận dạng (ví dụ, các tên ứng dụng) của tất cả các ứng dụng được cài đặt trên thiết bị đầu cuối có thể được truy vấn. Hơn nữa, thông tin được yêu cầu (ví dụ, các tên quy trình xử lý, các tình huống đang chạy, thông tin bộ nhớ được chiếm giữ lớn nhất) có thể nhận được từ Bảng 1 dựa vào các ký hiệu nhận dạng ứng dụng, để tạo ra danh mục bộ nhớ được chiếm giữ của các chương trình nội bộ. Danh mục bộ nhớ được chiếm giữ của các chương trình nội bộ có thể được lưu trữ nhờ sử dụng tệp ngôn ngữ đánh dấu mở rộng (Extensible Markup Language, XML) hoặc tệp TXT. Đối với ứng dụng khác nhau, thiết bị đầu cuối có thể truy vấn thông tin bộ nhớ được chiếm giữ lớn nhất của ứng dụng từ danh mục bộ nhớ được chiếm giữ của các chương trình nội bộ, và thiết đặt ngưỡng bộ nhớ tới bộ nhớ lớn nhất

được chiếm giữ của ứng dụng.

[0069] (3) Thực hiện việc tạo cấu hình dựa vào lược sử chiếm giữ của bộ nhớ bởi các quy trình xử lý chạy trên hệ thống.

[0070] Mỗi người dùng có một đặc điểm duy nhất. Sau khi thiết bị đầu cuối được sử dụng trong thời gian dài, các ứng dụng được cài đặt trên thiết bị đầu cuối không đồng nhất với các ứng dụng chạy trên thiết bị đầu cuối. Ngưỡng bộ nhớ có thể được tạo cấu hình dựa vào lược sử chiếm giữ của bộ nhớ bởi quy trình xử lý chạy trên hệ thống. Cụ thể là, thông tin trạng thái lịch sử (procstats) về sự chiếm giữ bộ nhớ trong suốt thời gian hoạt động của mỗi quy trình xử lý có thể được truy vấn khi thiết bị đầu cuối không được sử dụng bởi người dùng (ví dụ, vào buổi đêm chẳng hạn), để nhận được thông tin chiếm giữ bộ nhớ lớn nhất (ví dụ, tên chương trình ứng dụng và bộ nhớ lớn nhất được chiếm giữ). Hơn nữa, bộ nhớ lớn nhất được chiếm giữ được so với ngưỡng bộ nhớ hệ thống hiện thời. Nếu bộ nhớ lớn nhất được chiếm giữ lớn hơn so với ngưỡng bộ nhớ hệ thống hiện thời, bộ nhớ lớn nhất được chiếm giữ được sử dụng làm ngưỡng bộ nhớ hệ thống mới.

[0071] (4) Tạo cấu hình ngưỡng bộ nhớ nhờ sự phân phối đám mây.

[0072] Máy chủ đám mây có thể tập hợp thông tin về bộ nhớ được chiếm giữ bởi tất cả các ứng dụng chạy trên thiết bị đầu cuối trong các kịch khác nhau bản dựa vào ký hiệu nhận dạng của thiết bị đầu cuối, phát triển bảng tính nhờ sử dụng thông tin chẳng hạn như tên của ứng dụng mà nó chiếm bộ nhớ lớn hơn so với trị số thiết đặt trước (ví dụ, 300 MB chẳng hạn) và tình huống đang chạy của ứng dụng sau đó, và đẩy bảng tính tới thiết bị đầu cuối một cách thường xuyên (ví dụ, mỗi tháng chẳng hạn). Sau khi thu bảng tính, thiết bị đầu cuối duyệt thông tin ứng dụng trong bảng tính và danh mục ứng dụng nội bộ, và xác định thông tin chiếm giữ bộ nhớ lớn nhất của các ứng dụng được cài đặt trên thiết bị đầu cuối. Hơn nữa, bộ nhớ lớn nhất được chiếm giữ được so với ngưỡng bộ nhớ hệ thống hiện thời. Nếu bộ nhớ lớn nhất được chiếm giữ lớn hơn so với ngưỡng bộ nhớ hệ thống hiện thời, bộ nhớ lớn nhất được chiếm giữ được sử dụng làm ngưỡng bộ nhớ hệ thống mới.

[0073] Một cách tùy ý, ngưỡng bộ nhớ có thể được tạo cấu hình ở dạng động.

[0074] Cụ thể là, nhờ sử dụng thuật toán phân tích hành vi người dùng, các

ứng dụng có khả năng nhất đang chạy sau đó có thể được dự báo nhờ sử dụng thuật toán dự báo dựa vào thông tin chẳng hạn như thói quen sử dụng của người dùng, và lịch sử sử dụng, tần số sử dụng, và khoảng thời gian sử dụng của các ứng dụng, để tạo nên danh mục ứng dụng. Bộ nhớ lớn nhất được yêu cầu bởi các ứng dụng trong danh mục ứng dụng có thể nhận được, và bộ nhớ lớn nhất được so với ngưỡng bộ nhớ hệ thống hiện thời. Nếu bộ nhớ lớn nhất lớn hơn so với ngưỡng bộ nhớ hệ thống hiện thời, bộ nhớ lớn nhất được sử dụng làm ngưỡng bộ nhớ hệ thống mới.

[0075] 202: Tạo danh mục xử lý nền sau khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ.

[0076] Danh mục xử lý nền sau cần được tạo ra ngay lập tức khi xác định được rằng áp lực bộ nhớ hiện có trong hệ thống. Danh mục xử lý nền sau là danh mục hai chiều động mà được phân loại dựa vào mức độ quan trọng ứng dụng và quyền ưu tiên xử lý, và danh mục xử lý nền sau này bao gồm quy trình xử lý của một hoặc nhiều ứng dụng. Nói cách khác, danh mục xử lý nền sau có thể bao gồm một hoặc nhiều ứng dụng, và một ứng dụng có thể bao gồm một hoặc nhiều quy trình xử lý. Danh mục xử lý nền sau là danh mục được phân loại dựa vào mức độ quan trọng ứng dụng và quyền ưu tiên xử lý.

[0077] Cụ thể là, cách thức tạo danh mục xử lý nền sau bao gồm các bước sau:

(11) xác định chỉ số của thành phần chính của mỗi ứng dụng hiện đang chạy trong nền sau, trong đó thành phần chính bao gồm một hoặc nhiều trong số sau: quyền ưu tiên xử lý, thói quen sử dụng của người dùng, tài nguyên hệ thống được chiếm giữ bởi quy trình xử lý, và mối liên quan ứng dụng;

(12) thực hiện phép tính trọng số trên các chỉ số của tất cả các thành phần chính của mỗi ứng dụng, để nhận được mức độ quan trọng của ứng dụng;

(13) phân loại tất cả các ứng dụng dựa vào các mức độ quan trọng của tất cả các ứng dụng; và

(14) phân loại, dựa vào các quyền ưu tiên xử lý, các quy trình xử lý được bao gồm trong mỗi ứng dụng được phân loại, để tạo ra danh mục xử lý nền sau.

[0078] Theo phương án này, mỗi ứng dụng chạy trong nền sau của thiết bị

đầu cuối có thể bao gồm một hoặc nhiều quy trình xử lý, và thành phần chính của mỗi ứng dụng bao gồm một hoặc nhiều trong số sau: quyền ưu tiên xử lý, thói quen sử dụng của người dùng, tài nguyên hệ thống được chiếm giữ bởi quy trình xử lý, và mối liên quan ứng dụng. Mỗi thành phần chính có chỉ số tương ứng. Khi danh mục xử lý nền sau được tạo ra dựa vào các thành phần chính (quyền ưu tiên xử lý, thói quen sử dụng của người dùng, tài nguyên hệ thống được chiếm giữ bởi quy trình xử lý, và mối liên quan ứng dụng) của các ứng dụng, quy trình xử lý mà nó yêu cầu việc xử lý có thể được xác định sau đó từ danh mục xử lý nền sau dựa vào yêu cầu bộ nhớ. Theo cách này, trình tự xử lý có thể loại bỏ có thể được lựa chọn một cách chính xác, làm giảm khả năng loại bỏ lỗi/quá mức/thiếu hụt của các quy trình xử lý.

[0079] Hệ thống đánh giá mức quan trọng của mỗi quy trình xử lý, và mức quan trọng của quy trình xử lý cũng biểu diễn quyền ưu tiên của quy trình xử lý. Nói chung, mức quan trọng được biểu diễn bởi trị số `oom_adj` (điều chỉnh tràn bộ nhớ) và được ấn định tới mỗi quy trình xử lý. Hệ thống xác định các quy trình xử lý nào cần được kết thúc dựa vào `oom_adj`. Nói chung, chỉ số `oom_adj` được cung cấp bởi hệ thống. Hệ thống ấn định chỉ số `oom_adj` dựa vào trạng thái chạy ứng dụng hiện thời. Khoảng của chỉ số `oom_adj` là như sau: $-17 < \text{oom_adj} < 16$. Chỉ số `oom_adj` lớn hơn chỉ báo khả năng cao hơn là quy trình xử lý được lựa chọn được kết thúc bởi hệ thống, và mức ưu tiên thấp hơn của quy trình xử lý.

[0080] Thói quen sử dụng của người dùng có thể bao gồm nhưng không giới hạn ở bản ghi thời gian sử dụng, các lần sử dụng được tích lũy, khoảng thời gian mỗi lần sử dụng, và khoảng thời gian sử dụng được tích lũy mà là của mỗi ứng dụng. Dựa vào thói quen sử dụng của người dùng, thông tin quy trình xử lý liên quan có thể được xác định, ví dụ, các quy trình xử lý nào là các quy trình xử lý của ứng dụng mà thường xuyên được sử dụng bởi người dùng, các quy trình xử lý nào là các quy trình xử lý của ứng dụng mà người dùng đã sử dụng trong thời gian tương đối dài, hoặc các quy trình xử lý nào là các quy trình xử lý của ứng dụng mà được sử dụng gần đây bởi người dùng. Môđun RMS 1221 có thể được sử dụng để nhận biết thói quen sử dụng ứng dụng. Ngoài ra, mức độ quan trọng của ứng dụng có thể được nhận dạng nhờ xét đến ba thuật toán cùng với nhau.

[0081] Dựa vào Fig.2A, Fig.2A là hình vẽ giản lược của việc nhận dạng mức

độ quan trọng ứng dụng được bộc lộ trong một phương án của sáng chế. Như được thể hiện trên Fig.2A, ô nét đứt 1 biểu diễn mẫu nhận dạng mức độ quan trọng ứng dụng trên cơ sở học máy, ô nét đứt 2 biểu diễn mẫu nhận dạng mức độ quan trọng ứng dụng trên cơ sở quy tắc thống kê, và thuật toán thứ ba là nhận dạng dựa vào ứng dụng được sử dụng gần nhất (Latest Recent Used, viết tắt là LRU). Đối với ứng dụng a, giả sử rằng mức độ quan trọng được nhận dạng dựa vào việc học máy là $Ph(a)$, mức độ quan trọng được nhận dạng dựa vào quy tắc thống kê là $Pl(a)$, mức độ quan trọng được nhận dạng dựa vào ứng dụng được sử dụng gần nhất là $Pu(a)$, và các trọng số tương ứng lần lượt là Wh , Wl , và Wu . Sau đó công thức dùng để tính mức độ quan trọng của ứng dụng a là:

$$P(a) = Wh * Ph(a) + Wl * Pl(a) + Wu * Pu(a)$$

[0082] Các trọng số nhận được nhờ nhận biết liên tục trong suốt thời gian đào tạo và là độc lập với người dùng. Các trọng số được thiết đặt trong việc nhận dạng mức độ quan trọng ứng dụng được thêm vào và được cập nhật qua sự lặp lại phiên bản.

[0083] Chỉ số thói quen (habit) sử dụng của người dùng có thể được phân bổ dựa vào mức độ quan trọng ứng dụng, và khoảng của chỉ số thói quen là như sau: $0 < \text{habit} < 16$. Chỉ số thói quen thấp hơn chỉ báo việc sử dụng thường xuyên hơn của ứng dụng bởi người dùng.

[0084] Tài nguyên được chiếm giữ bởi quy trình xử lý có thể bao gồm nhưng không giới hạn ở tài nguyên tính toán, tài nguyên lưu trữ, tài nguyên bộ xử lý trung tâm (Central Processing Unit, viết tắt là CPU), và tài nguyên đầu vào/đầu ra (InputOutput, viết tắt là IO). Chỉ số tài nguyên (resource) hệ thống được chiếm giữ bởi quy trình xử lý có thể được thiết đặt dựa vào kích thước của tài nguyên được chiếm giữ, và khoảng của chỉ số tài nguyên là như sau: $0 < \text{tài nguyên (resource)} < 16$. Ví dụ, khi sự chiếm giữ CPU bởi quy trình xử lý lớn hơn so với 3%, chỉ số tài nguyên là 16. Nói chung, nhiều tài nguyên hệ thống hơn được chiếm giữ bởi quy trình xử lý chỉ báo rằng quy trình xử lý là thích hợp hơn được lựa chọn mà trong đó bộ nhớ tương ứng với quy trình xử lý cần được phục hồi.

[0085] Mỗi liên quan ứng dụng đề cập đến mỗi liên quan tồn tại giữa các ứng dụng. Cụ thể là, hai ứng dụng có thể được liên quan nhờ sử dụng một quy trình xử lý. Theo phương án này của sáng chế, mỗi liên quan ứng dụng có thể là mỗi

liên quan giữa ứng dụng nền sau và ứng dụng nền trước. Mỗi liên quan này là tạm thời và có thể được cập nhật theo thời gian thực dựa vào trạng thái xử lý hiện thời. Mỗi liên quan bao gồm nhưng không giới hạn ở getContentProvider, startActivity, bindService, unbindService, startService, stopService, và widget.

[0086] Dựa vào Fig.2B, Fig.2B là hình vẽ giản lược của mỗi liên quan ứng dụng được bộc lộ trong một phương án của sáng chế, trong đó Pid1, Pid2, Pid3, Pid4 ... biểu diễn các ký hiệu nhận dạng quy trình xử lý. Quy trình xử lý ngoài cùng bên trái trên Fig.2B được dựa vào bởi quy trình xử lý ngoài cùng bên phải và mỗi liên quan tồn tại giữa hai quy trình xử lý. Ví dụ, Service1 của quy trình xử lý Pid1 được liên quan đến quy trình xử lý Pid2 nhờ sử dụng bindService, và Service1 của quy trình xử lý Pid2 được liên quan đến quy trình xử lý Pid3 nhờ sử dụng startService.

[0087] Nói chung, trong thời gian dọn sạch quy trình xử lý nền sau, quy trình dọn sạch có thể có khả năng được lấy ra bởi quy trình xử lý được liên quan đến quy trình dọn sạch, hoặc sự bất thường về chức năng có thể xảy ra bởi vì quy trình xử lý được dọn sạch. Ví dụ, phương pháp đưa vào nền trước cần sử dụng quy trình cung cấp nền sau độc lập, và nếu quy trình cung cấp nền sau độc lập bị đóng băng (treo) hoặc được làm sạch, phương pháp đưa vào nền trước trở nên không bình thường. Khi ứng dụng nền sau là phát âm nhạc trong khi âm nhạc nằm trên quy trình xử lý dịch vụ nền sau độc lập, nếu quy trình xử lý dịch vụ độc lập bị đóng băng (treo) hoặc được làm sạch, ứng dụng phát âm nhạc trở nên không bình thường. Người dùng có thể thiết đặt chỉ số mỗi liên quan ứng dụng (application-relation) đối với thao tác lấy trọng số tiếp theo. Khoảng của chỉ số mỗi liên quan ứng dụng là như sau: $0 < \text{mỗi liên quan ứng dụng (application-relation)} < 16$. Nếu các ứng dụng liên quan trực tiếp, chỉ số mỗi liên quan ứng dụng có thể là 0; nếu các ứng dụng được kết hợp gián tiếp, chỉ số mỗi liên quan ứng dụng có thể là 16. Chỉ số mỗi liên quan ứng dụng cao hơn chỉ báo mỗi liên quan yếu hơn giữa một ứng dụng và ứng dụng khác.

[0088] Hơn nữa, đối với mỗi ứng dụng, phép tính trọng số có thể được thực hiện trên các chỉ số của tất cả các thành phần chính, để nhận được mức độ quan trọng của ứng dụng. Cụ thể là, thuật toán là như sau:

Chỉ số (Score) = oom_adj * P_oom_adj + thói quen (habit) * P_habit + tài nguyên (resource) * P_resource + mỗi liên quan ứng dụng (application-relation) *

$P_{\text{application-relation}}$, trong đó $P_{\text{oom_adj}} + P_{\text{habit}} + P_{\text{resource}} + P_{\text{application-relation}} = 1$. Chỉ số Score có thể được sử dụng để đo mức độ quan trọng của ứng dụng, và chỉ số Score nhỏ hơn chỉ báo mức độ quan trọng cao hơn của ứng dụng. Ví dụ, nếu chỉ số Score của ứng dụng 1 cao hơn chỉ số Score của ứng dụng 2, có thể xác định được rằng mức độ quan trọng của ứng dụng 2 cao hơn mức độ quan trọng của ứng dụng 1.

[0089] Hơn thế nữa, các ứng dụng có thể được phân loại dựa vào chỉ số Score. Đối với mỗi ứng dụng, khi ứng dụng bao gồm ít nhất hai quy trình xử lý, các quy trình xử lý được bao gồm trong ứng dụng có thể được phân loại dựa vào các quyền ưu tiên xử lý để tạo ra danh mục xử lý nền sau. Danh mục xử lý nền sau thực tế là danh mục hai chiều mà nó bao gồm các ứng dụng và các quy trình xử lý của các ứng dụng. Trong danh mục hai chiều này, các ứng dụng được phân loại dựa vào các mức độ quan trọng theo thứ tự giảm dần hoặc tăng dần, và các quy trình xử lý của các ứng dụng được phân loại dựa vào các quyền ưu tiên xử lý theo thứ tự giảm dần hoặc tăng dần.

[0090] Dựa vào Bảng 2, Bảng 2 là danh mục ứng dụng quy trình xử lý được bộc lộ trong một phương án của sáng chế.

Bảng 2

Ứng dụng 1	Quy trình xử lý 1
	Quy trình xử lý 2
Ứng dụng 2	Quy trình xử lý 3
	Quy trình xử lý 4
	Quy trình xử lý 5
Ứng dụng 3	Quy trình xử lý 6
	Quy trình xử lý 7
	Quy trình xử lý 8
...	...
Ứng dụng n	...

[0091] Các ứng dụng chạy trên hệ thống có thể được phân loại thành 18 loại, chủ yếu bao gồm: các dịch vụ hệ thống chính, các ứng dụng thường trú chính, các ứng dụng nền trước, các máy chủ hệ thống, các ứng dụng nền sau được nằm trên bởi ứng dụng nền trước, các dịch vụ hệ thống chung, các ứng dụng thường trú chung, các ứng dụng nằm trong nền sau, các ứng dụng nền sau người dùng hiểu được, các ứng dụng hữu hình, các ứng dụng nền sau quản lý điện thoại được bảo vệ, các ứng dụng chạy sau cùng trong nền trước, các ứng dụng nền sau trong mối quan hệ yếu với nền trước, các ứng dụng thường xuyên được sử dụng bởi người dùng, các ứng dụng với mức độ quan trọng được tạo cấu hình trước, các ứng dụng mà oom_adj của nó lớn hơn so với 2, các ứng dụng được nhận dạng bởi hệ thống như sự nguy hại, và các ứng dụng được nhận dạng thủ công như sự nguy hại (được liệt vào sổ đen).

[0092] Theo cách thực hiện tùy chọn, danh mục xử lý nền sau có thể là một danh mục (chẳng hạn như Bảng 2 nêu trên), hoặc danh mục xử lý nền sau có thể bao gồm ba danh mục (ví dụ, Bảng 2 nêu trên có thể được chia thành ba phần để tạo nên ba danh mục con) mà lần lượt là danh mục không thể khôi phục, danh mục quy trình xử lý quan trọng, và danh mục có thể loại bỏ được. Danh mục không thể khôi phục chủ yếu bao gồm các quy trình xử lý và các dịch vụ hệ thống. Việc thiếu các quy trình xử lý này sẽ ảnh hưởng đến việc hoạt động của tất cả các ứng dụng. Các ví dụ là các dịch vụ hệ thống chính, các ứng dụng thường trú chính, các ứng dụng nền trước, các máy chủ hệ thống, và các ứng dụng nền sau được nằm trên bởi ứng dụng nền trước. Danh mục quy trình xử lý quan trọng chủ yếu bao gồm các ứng dụng ngưỡng thiết bị, các ứng dụng người dùng hiểu được, và các ứng dụng quan trọng thường xuyên được sử dụng bởi người dùng, ví dụ, các dịch vụ hệ thống chung, các ứng dụng thường trú chung, các ứng dụng nằm trong nền sau, các ứng dụng nền sau người dùng hiểu được, các ứng dụng hữu hình, các ứng dụng nền sau quản lý điện thoại được bảo vệ, các ứng dụng chạy sau cùng trong nền trước, và các ứng dụng nền sau trong mối liên quan yếu với nền trước. Danh mục có thể loại bỏ được chủ yếu bao gồm các loại liên quan của các ứng dụng chẳng hạn như các ứng dụng thường xuyên được sử dụng bởi người dùng, các ứng dụng với mức độ quan trọng được tạo cấu hình trước, các ứng dụng mà oom_adj của nó lớn hơn so với 2, các ứng dụng được nhận dạng bởi hệ thống như sự nguy hại, và các ứng dụng được nhận dạng thủ công như sự nguy hại.

[0093] 203: Xác định quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nền sau.

[0094] Khi danh mục xử lý nền sau bao gồm ba danh mục con, danh mục phục hồi bộ nhớ được xử lý theo thứ tự sau: trong suốt thời gian phục hồi bộ nhớ, các quy trình xử lý trong danh mục có thể loại bỏ được phục hồi theo cách ưu tiên và sự phục hồi được thực hiện từ cuối của danh mục có thể loại bỏ được từ dưới lên; khi danh mục có thể loại bỏ được là trống và áp lực bộ nhớ hệ thống vẫn hiện hữu, các quy trình xử lý trong danh mục quy trình xử lý quan trọng được phục hồi và sự phục hồi vẫn được thực hiện từ cuối của danh mục quy trình xử lý quan trọng từ dưới lên. Dựa vào nguyên lý này, quy trình mà trong đó bộ nhớ cần được phục hồi có thể được xác định theo cách ưu tiên trước tiên từ danh mục có thể loại bỏ được, và sau đó quy trình mà trong đó bộ nhớ cần được phục hồi có thể được xác định từ danh mục quy trình xử lý quan trọng.

[0095] Một cách tùy ý, việc phân loại của các danh mục nêu trên có thể được thay đổi. Ví dụ, quy trình xử lý mà nằm trong danh mục quy trình xử lý quan trọng và mức quan trọng của nó là không đủ cao có thể bị giảm và được bao gồm trong danh mục có thể loại bỏ được.

[0096] Danh mục xử lý nền sau bao gồm các quy trình xử lý của các ứng dụng. Quy trình xử lý mà trong đó bộ nhớ cần được phục hồi là quy trình xử lý trong số các quy trình xử lý của các ứng dụng đáp ứng điều kiện là trị số tuyệt đối của độ sai lệch giữa bộ nhớ được chiếm giữ bởi quy trình xử lý và trị số áp lực bộ nhớ thấp hơn ngưỡng thiết đặt trước, và trị số áp lực bộ nhớ là độ sai lệch giữa ngưỡng bộ nhớ và bộ nhớ hữu dụng hiện thời của hệ thống. Nói cách khác, bộ nhớ được chiếm giữ bởi quy trình xử lý mà trong đó bộ nhớ cần được phục hồi có thể lớn hơn so với trị số áp lực bộ nhớ, hoặc có thể thấp hơn trị số áp lực bộ nhớ. Trị số tuyệt đối của độ sai lệch giữa bộ nhớ được chiếm giữ bởi quy trình xử lý mà trong đó bộ nhớ cần được phục hồi và trị số áp lực bộ nhớ nằm trong khoảng đưa ra. Nghĩa là, trị số tuyệt đối thấp hơn ngưỡng thiết đặt trước. Ngưỡng thiết đặt trước có thể được định rõ bởi người dùng, hoặc có thể là mặc định hệ thống. Điều này không giới hạn theo phương án này của sáng chế.

[0097] Các mức áp lực bộ nhớ được thiết đặt: mức cao, mức trung bình, và mức thấp. Mức cao chỉ báo rằng áp lực bộ nhớ là tương đối lớn, mức trung bình chỉ báo rằng áp lực bộ nhớ là vừa phải, và mức thấp chỉ báo rằng áp lực bộ nhớ

là tương đối thấp. Ngoài ra, khoảng trị số áp lực bộ nhớ được thiết đặt: khoảng thứ nhất, khoảng thứ hai, và khoảng thứ ba, trong đó khoảng thứ nhất thấp hơn khoảng thứ hai, và khoảng thứ hai thấp hơn khoảng thứ ba. Mức áp lực bộ nhớ hiện thời của hệ thống có thể được xác định dựa vào khoảng trị số áp lực bộ nhớ. Ví dụ, khi trị số áp lực bộ nhớ nằm trong khoảng thứ nhất, mức áp lực bộ nhớ hiện thời của hệ thống có thể được xác định là mức thấp; khi trị số áp lực bộ nhớ nằm trong khoảng thứ hai, mức áp lực bộ nhớ hiện thời của hệ thống có thể được xác định là mức trung bình; và khi trị số áp lực bộ nhớ nằm trong khoảng thứ ba, mức áp lực bộ nhớ hiện thời của hệ thống có thể được xác định là mức cao.

[0098] Khi mức áp lực bộ nhớ hiện thời của hệ thống được xác định là mức cao, chính sách dọn sạch quy trình xử lý mà trong đó bộ nhớ cần được phục hồi cần được sử dụng. Theo chính sách này, tài nguyên được chiếm giữ bởi quy trình xử lý hiện tại trong thiết bị đầu cuối cần được vô hiệu hóa và được phục hồi, để làm tăng bộ nhớ khả dụng của hệ thống.

[0099] Khi mức áp lực bộ nhớ hiện thời của hệ thống được xác định là mức trung bình, chính sách dọn sạch quy trình xử lý mà trong đó bộ nhớ cần được phục hồi trên mức hệ thống hoặc mức ứng dụng cần được sử dụng. Theo chính sách phục hồi bộ nhớ mức hệ thống, bộ nhớ đệm hệ thống có thể được phục hồi dựa vào mức độ quan trọng, để đạt được mục đích làm tăng bộ nhớ không làm việc. Một ứng dụng có thể có một hoặc nhiều quy trình xử lý. Do đó, chính sách phục hồi bộ nhớ mức ứng dụng, ví dụ, loại bỏ, nén, hoặc bỏ qua bộ nhớ đệm, có thể được sử dụng.

[0100] Khi mức áp lực bộ nhớ hiện thời của hệ thống được xác định là mức thấp, chính sách thực hiện việc nén mức quy trình xử lý hoặc phục hồi bộ nhớ đệm mức quy trình xử lý cần được sử dụng đối với quy trình xử lý mà trong đó bộ nhớ cần được phục hồi dựa vào mức độ quan trọng ứng dụng. Nếu mức độ quan trọng ứng dụng là tương đối cao, việc nén mức quy trình xử lý có thể được thực hiện đối với quy trình xử lý mà trong đó bộ nhớ cần được phục hồi, và nếu mức độ quan trọng ứng dụng là tương đối thấp, sự phục hồi bộ nhớ đệm mức quy trình xử lý có thể được thực hiện đối với quy trình xử lý mà trong đó bộ nhớ cần được phục hồi. Việc nén mức quy trình xử lý trên quy trình xử lý mà trong đó bộ nhớ cần được phục hồi là để thực hiện việc xử lý nén trên tài

nguyên bộ nhớ được chiếm giữ bởi quy trình xử lý chạy trên thiết bị đầu cuối nhờ sử dụng sự phân chia hoán đổi, nhằm đạt được mục đích làm giảm sự chiếm giữ của bộ nhớ vật lý của hệ thống. Sự phục hồi bộ nhớ đệm mức quy trình xử lý trên quy trình xử lý mà trong đó bộ nhớ cần được phục hồi là để giải phóng không gian bộ nhớ được chiếm giữ bởi bộ nhớ đệm trong thiết bị đầu cuối trước đó, để làm giảm bộ nhớ được chiếm giữ bộ nhớ đệm, làm tăng bộ nhớ không làm việc, và nhờ đó làm tăng hiệu quả phân bổ bộ nhớ.

[0101] Theo phương án này của sáng chế, quy trình xử lý mà trong đó bộ nhớ cần được phục hồi có thể được xác định từ danh mục xử lý nền sau dựa vào áp lực bộ nhớ hiện thời của hệ thống. Theo cách này, quy trình xử lý quan trọng không bị loại bỏ do lỗi, và việc loại bỏ sự quá mức hoặc thiếu hụt của các quy trình xử lý được tránh khỏi.

[0102] 204: Gửi lệnh xử lý tới nhân hệ thống, để kích hoạt nhân hệ thống thực hiện việc xử lý trên quy trình xử lý mà trong đó bộ nhớ cần được phục hồi, để phục hồi bộ nhớ được chiếm giữ bởi quy trình xử lý mà trong đó bộ nhớ cần được phục hồi.

[0103] Theo phương án này của sáng chế, các chính sách phục hồi bộ nhớ khác nhau có thể được sử dụng đối với các mức áp lực bộ nhớ khác nhau, và mỗi chính sách phục hồi bộ nhớ là tương ứng với một giao diện. Lệnh xử lý có thể được gửi tới nhân hệ thống nhờ sử dụng giao diện tương ứng với chính sách phục hồi bộ nhớ. Sau khi thu lệnh xử lý, nhân hệ thống sử dụng chính sách phục hồi bộ nhớ tương ứng để thực hiện việc xử lý trên quy trình xử lý mà trong đó bộ nhớ cần được phục hồi, để phục hồi bộ nhớ tương ứng với quy trình xử lý mà trong đó bộ nhớ cần được phục hồi.

[0104] Lệnh xử lý có thể được gửi tới nhân hệ thống nhờ sử dụng tín hiệu KILL_SIG. Thông thường, một lệnh xử lý có thể mang một ký hiệu nhận dạng của một quy trình xử lý mà trong đó bộ nhớ cần được phục hồi.

[0105] Có thể có một hoặc nhiều quy trình xử lý được xác định mà trong đó bộ nhớ cần được yêu cầu. Khi một quy trình xử lý yêu cầu gửi nhiều lệnh xử lý, các lệnh xử lý có thể được gửi liên tiếp từng lệnh một đối với quy trình xử lý, mà không cần kiểm tra, sau khi gửi một lệnh xử lý, trạng thái của bộ nhớ tương ứng với quy trình xử lý mà trong đó bộ nhớ được phục hồi bởi nhân hệ thống trước khi gửi lệnh xử lý khác.

[0106] Theo cách thực hiện tùy chọn, khi các quy trình xử lý mà trong đó bộ nhớ cần được phục hồi được xác định, việc gửi lệnh xử lý tới nhân hệ thống bao gồm bước:

gọi các luồng để gửi các lệnh xử lý tới nhân hệ thống, trong đó mỗi luồng được sử dụng để gửi một hoặc nhiều lệnh xử lý.

[0107] Theo ứng dụng tùy chọn này, khi các quy trình xử lý mà trong đó bộ nhớ cần được phục hồi được xác định, các luồng có thể được gọi để gửi các lệnh xử lý tới nhân hệ thống. Cụ thể là, việc thực hiện gửi nhiều lệnh xử lý bởi một luồng là: luồng gửi các lệnh xử lý thành công từng lệnh một, mà không cần kiểm tra, sau khi gửi một lệnh xử lý, trạng thái của bộ nhớ tương ứng với quy trình xử lý mà trong đó bộ nhớ được phục hồi bởi nhân hệ thống trước khi gửi lệnh xử lý khác. Sau khi hệ thống phục hồi bộ nhớ được chiếm giữ bởi quy trình xử lý mà trong đó bộ nhớ cần được phục hồi, áp lực bộ nhớ hiện thời của hệ thống có thể được giảm bớt.

[0108] Có thể thấy được rằng theo cách thực hiện này, không cần chờ nhân hệ thống kết thúc chính thức thao tác phục hồi trên bộ nhớ tương ứng với quy trình xử lý, và cơ chế phục hồi không đồng bộ của bộ nhớ tương ứng với các quy trình xử lý mà trong đó bộ nhớ cần được phục hồi có thể được sử dụng trực tiếp, để cải thiện hiệu suất phục hồi quy trình xử lý nền sau và nâng cao hiệu quả phục hồi bộ nhớ.

[0109] Dựa vào Fig.2C và Fig.2D, Fig.2C là hình vẽ của trình tự khởi động camera được bộc lộ trong một phương án của sáng chế, và Fig.2D là hình vẽ của trình tự khởi động camera khác được bộc lộ trong một phương án của sáng chế. Trên Fig.2C, khi bộ nhớ khả dụng của hệ thống chỉ là 619 MB, việc phục hồi bộ nhớ không được thực hiện khi chương trình ứng dụng camera bắt đầu khởi động. Khi bộ nhớ của hệ thống nằm dưới mức nước hệ thống, cơ chế loại bỏ bộ nhớ thấp (Low Memory Killer, viết tắt là LMK) được sử dụng cho việc phục hồi bộ nhớ. Có thể thấy được từ Fig.2C rằng mất tổng số 3,255 giây (s) từ khi khởi động toàn bộ camera chỉ ở bộ nhớ 619 MB đến khi xuất hiện nút chụp ảnh của camera. Trên Fig.2D, khi bộ nhớ khả dụng của hệ thống chỉ là 604 MB, giải pháp trong các phương án của sáng chế được sử dụng sao cho việc phục hồi bộ nhớ được thực hiện nhờ sử dụng chính sách tương ứng khi chương trình ứng dụng camera bắt đầu khởi động. Có thể thấy được từ Fig.2D rằng mất tổng số

1,85 giây từ khi khởi động toàn bộ camera chỉ ở bộ nhớ 604 MB đến khi xuất hiện nút chụp ảnh của camera. Từ sự so sánh giữa Fig.2C và Fig.2D, có thể thấy được rằng trong cùng môi trường thử nghiệm, khi bộ nhớ khả dụng không khác nhau nhiều, hiệu suất khởi động camera được cải thiện đáng kể trong giải pháp của sáng chế so với giải pháp LMK thông thường.

[0110] Có thể thấy được rằng trong quy trình xử lý của phương pháp được mô tả trên Fig.2, khi sự kiện chính thứ nhất được phát hiện và xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, việc phục hồi bộ nhớ một cách chủ động có thể được thực hiện. Khi chương trình ứng dụng thực sự cần đến một lượng lớn bộ nhớ, nhân hệ thống đã phục hồi một lượng lớn của bộ nhớ khả dụng. Do đó, khả năng các lần dừng gián đoạn của thiết bị đầu cuối có thể được giảm.

[0111] Dựa vào Fig.3, Fig.3 là lưu đồ giản lược của phương pháp phục hồi bộ nhớ khác được bộc lộ trong một phương án của sáng chế. Phương pháp có thể được áp dụng tới môđun điều khiển bộ nhớ 1223 được thể hiện trên Fig.1. Cụ thể là, môđun điều khiển bộ nhớ 1223 có thể là một hoặc nhiều luồng, hoặc môđun điều khiển bộ nhớ 1223 có thể là một hoặc nhiều quy trình xử lý. Như được thể hiện trên Fig.3, phương pháp có thể bao gồm các bước sau.

[0112] 301: Xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ khi sự kiện chính thứ hai được phát hiện và khi xác định được rằng hệ thống là ở trạng thái không hoạt động.

[0113] Theo phương án này của sáng chế, môđun điều khiển bộ nhớ 1223 có thể phát hiện sự kiện chính của chương trình ứng dụng. Cụ thể là, thông tin chính (ví dụ, tên quy trình xử lý, ký hiệu nhận dạng người dùng (User Identifier, viết tắt là UID) và ký hiệu nhận dạng quy trình xử lý (Process Identifier, viết tắt là PID)) của chương trình ứng dụng có thể nhận được nhờ sử dụng chức năng chặn (hook) khi dịch vụ quản lý hoạt động (Activity Manager Service, viết tắt là AMS) khởi động hoạt động (activity) của mỗi ứng dụng. Hơn nữa, AMS có thể gửi thông tin chính nhận được của chương trình ứng dụng tới môđun điều khiển bộ nhớ 1223. Do đó, môđun điều khiển bộ nhớ 1223 có thể phát hiện sự kiện chính của chương trình ứng dụng.

[0114] Các sự kiện chính của chương trình ứng dụng có thể bao gồm, nhưng không giới hạn ở, sự kiện bắt đầu khởi động chương trình, sự kiện dọn sạch,

tràn bộ nhớ (Out Of Memory, viết tắt là OOM), sự kiện hoàn thành khởi động chương trình, sự kiện bật màn hình, sự kiện tắt màn hình, sự kiện chạm màn hình, giao diện chuyển đổi giao diện người dùng, sự kiện kết thúc chuyển đổi thao tác, sự kiện phát rộng, sự kiện thay đổi trạng thái/quét/kết nối mạng (ví dụ, Wi-Fi, 2/3/4G, Bluetooth, hồng ngoại, GPS, phát rộng radio, hoặc truyền thông trường gần (Near Field Communication, viết tắt là NFC)), sự kiện thay đổi kết nối ngoại vi (ví dụ, bộ nạp được lắp/tháo chẳng hạn), sự kiện báo hiệu/cảnh báo, sự kiện gọi điện thoại (cuộc gọi đến, thay đổi tín hiệu, hoặc tương tự), sự kiện đa phương tiện (thay đổi audio/video hoặc phát audio được ghi), sự kiện liên quan đến đèn chớp, sự kiện lắp đặt/gỡ bỏ chương trình, sự kiện mức pin, và tương tự.

[0115] Theo phương án này của sáng chế, nếu thao tác tương tác người và máy hiện hữu trong hệ thống và hệ thống trực tiếp thực hiện việc phục hồi bộ nhớ, trải nghiệm của người dùng có thể bị ảnh hưởng bởi vì sự phản hồi nhanh chóng tới thao tác người dùng không thể nhận được từ CPU. Trong trường hợp này, việc phục hồi bộ nhớ ngay lập tức không thể được thực hiện.

[0116] Một sự kiện mà không yêu cầu phục hồi bộ nhớ ngay lập tức có thể được định rõ trước là sự kiện chính thứ hai. Sự kiện chính thứ hai có thể bao gồm nhưng không giới hạn ở sự kiện hoàn thành khởi động chương trình, sự kiện bật màn hình, sự kiện tắt màn hình, sự kiện chạm màn hình, sự kiện chuyển đổi giao diện người dùng, sự kiện kết thúc chuyển đổi thao tác, và sự kiện phát rộng.

[0117] Khi sự kiện chính thứ hai được phát hiện, đầu tiên được xác định xem hệ thống ở trạng thái không hoạt động hay không. Sau đó, khi xác định được rằng hệ thống là ở trạng thái không hoạt động, bộ nhớ hữu dụng hiện thời của hệ thống được đọc và được xác định xem bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ hay không. Khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, nó cho biết rằng áp lực bộ nhớ là hiện hữu trong hệ thống và việc phục hồi bộ nhớ cần được thực hiện. Ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ nhất tương ứng với sự kiện chính thứ hai. Trạng thái không hoạt động của hệ thống có thể bao gồm trạng thái không hoạt động thiết bị đầu cuối của thiết bị đầu cuối và trạng thái không hoạt động ứng dụng.

[0118] Một cách tùy ý, ngưỡng bộ nhớ có thể được thiết đặt trước ở dạng tĩnh;

hoặc ngưỡng bộ nhớ có thể được tạo cấu hình ở dạng động. Đối với thông tin cụ thể, tham chiếu các phần mô tả trên Fig.2, và các phần chi tiết không được mô tả lại ở đây.

[0119] Một cách tùy ý, việc xem hệ thống ở trạng thái không hoạt động hay không có thể được xác định nhờ đọc trạng thái chạy của thiết bị đầu cuối; hoặc việc xem hệ thống ở trạng thái không hoạt động hay không có thể được xác định nhờ xác định trạng thái tải hiện thời của hệ thống.

[0120] Có hai tình huống trong đó hệ thống là ở trạng thái không hoạt động. Tình huống thứ nhất là thiết bị đầu cuối đang không được sử dụng bởi người dùng. Ví dụ, thiết bị đầu cuối có thể được xét đến là ở trạng thái không hoạt động khi thiết bị đầu cuối đang ở chế độ chờ. Đối với ví dụ khác, khi việc khởi động của ứng dụng được kết thúc và màn hình hiển thị xuất hiện, nhưng sau đó không có sự tương tác với thiết bị đầu cuối, trong trường hợp này, hệ thống không có số lượng lớn các thao tác IO và có thể được xét đến là ở trạng thái không hoạt động. Tình huống thứ hai là hệ thống có thể được xét đến là ở trạng thái không hoạt động khi thiết bị đầu cuối đang được sử dụng bởi người dùng nhưng tải hệ thống hiện thời không ảnh hưởng đến trải nghiệm của người dùng, ví dụ, khi tải hệ thống thấp hơn ngưỡng tải.

[0121] Cụ thể là, xác định rằng hệ thống là ở trạng thái không hoạt động có thể bao gồm các bước sau:

[0122] (11) Xác định xem tải hiện thời của hệ thống thấp hơn ngưỡng tải hay không.

[0123] (12) Khi tải hiện thời của hệ thống thấp hơn ngưỡng tải, xác định rằng hệ thống là ở trạng thái không hoạt động.

[0124] Theo phương án này, ngưỡng tải có thể được thiết đặt. Ngưỡng tải có thể là trị số tới hạn của tải hệ thống mà nó ảnh hưởng đến trải nghiệm của người dùng. Nếu xác định được rằng tải hệ thống hiện thời thấp hơn ngưỡng tải, nó cho biết rằng tải hệ thống hiện thời không ảnh hưởng đến trải nghiệm của người dùng, và có thể xác định được rằng hệ thống là ở trạng thái không hoạt động.

[0125] 302: Tạo danh mục xử lý nền sau khi hệ thống ở trạng thái không hoạt động.

[0126] Theo phương án này của sáng chế, việc phục hồi bộ nhớ cần được

thực hiện khi hệ thống ở trạng thái không hoạt động. Tuy nhiên, danh mục xử lý nền sau là danh mục động. Do đó, danh mục xử lý nền sau cũng cần được tạo ra khi hệ thống ở trạng thái không hoạt động. Điều này có thể đảm bảo hiệu lực của danh mục xử lý nền sau.

[0127] 303: Xác định quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nền sau.

[0128] 304: Gửi lệnh xử lý tới nhân hệ thống, để kích hoạt nhân hệ thống thực hiện việc xử lý trên quy trình xử lý mà trong đó bộ nhớ cần được phục hồi, để phục hồi bộ nhớ tương ứng với quy trình xử lý mà trong đó bộ nhớ cần được phục hồi.

[0129] Có thể thấy được rằng trong quy trình xử lý của phương pháp được mô tả trên Fig.3, xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ khi sự kiện chính thứ hai được phát hiện và khi hệ thống ở trạng thái không hoạt động, và việc phục hồi bộ nhớ hiệu dụng có thể được thực hiện. Khi chương trình ứng dụng thực sự cần đến một lượng lớn bộ nhớ, nhân hệ thống đã phục hồi một lượng lớn của bộ nhớ khả dụng. Do đó, khả năng các lần dừng gián đoạn của thiết bị đầu cuối có thể được giảm.

[0130] Dựa vào Fig.4, Fig.4 là lưu đồ giản lược của phương pháp phục hồi bộ nhớ khác được bộc lộ trong một phương án của sáng chế. Phương pháp có thể được áp dụng tới môđun điều khiển bộ nhớ 1223 được thể hiện trên Fig.1. Cụ thể là, môđun điều khiển bộ nhớ 1223 có thể là một hoặc nhiều luồng, hoặc môđun điều khiển bộ nhớ 1223 có thể là một hoặc nhiều quy trình xử lý. Như được thể hiện trên Fig.4, phương pháp có thể bao gồm các bước sau.

[0131] 401: Xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ khi sự kiện chính thứ hai được phát hiện.

[0132] Đối với thông tin cụ thể về sự kiện chính thứ hai, tham chiếu các phần mô tả trên Fig.3, và các phần chi tiết không được mô tả lại ở đây. Đối với thông tin cụ thể về ngưỡng bộ nhớ, tham chiếu các phần mô tả trên Fig.2, và các phần chi tiết không được mô tả lại ở đây.

[0133] Theo phương án này của sáng chế, theo phương án được thể hiện trên Fig.3, khi sự kiện chính thứ hai được phát hiện, trước tiên được xác định xem hệ thống ở trạng thái không hoạt động hay không, và sau đó khi xác định được

rằng hệ thống là ở trạng thái không hoạt động, bộ nhớ hữu dụng hiện thời của hệ thống được đọc. Theo phương án được thể hiện trên Fig.4, tuy nhiên, khi sự kiện chính thứ hai được phát hiện, bộ nhớ hữu dụng hiện thời của hệ thống được đọc đầu tiên, và sau đó khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, xác định được xem hệ thống ở trạng thái không hoạt động hay không. Khi xác định được rằng hệ thống là ở trạng thái không hoạt động, bước 402 được thực hiện.

[0134] 402: Tạo danh mục xử lý nền sau khi xác định được rằng hệ thống là ở trạng thái không hoạt động.

[0135] 403: Xác định quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nền sau.

[0136] 404: Gửi lệnh xử lý tới nhân hệ thống, để kích hoạt nhân hệ thống thực hiện việc xử lý trên quy trình xử lý mà trong đó bộ nhớ cần được phục hồi, để phục hồi bộ nhớ tương ứng với quy trình xử lý mà trong đó bộ nhớ cần được phục hồi.

[0137] Có thể thấy được rằng trong quy trình xử lý của phương pháp được mô tả trên Fig.4, khi sự kiện chính thứ hai được phát hiện và xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, việc phục hồi bộ nhớ một cách chủ động có thể được thực hiện khi hệ thống là ở trạng thái không hoạt động. Khi chương trình ứng dụng thực sự cần đến một lượng lớn bộ nhớ, nhân hệ thống đã phục hồi một lượng lớn của bộ nhớ khả dụng. Do đó, khả năng các lần dừng gián đoạn của thiết bị đầu cuối có thể được giảm.

[0138] Dựa vào Fig.5, Fig.5 là sơ đồ cấu trúc giản lược của thiết bị phục hồi bộ nhớ được bộc lộ trong một phương án của sáng chế. Thiết bị phục hồi bộ nhớ có thể được tạo cấu hình để thực hiện một vài hoặc tất cả các bước của phương pháp phục hồi bộ nhớ được mô tả trên các hình vẽ từ Fig.2 đến Fig.4. Đối với thông tin cụ thể của nó, tham chiếu đến các phần mô tả liên quan của các hình vẽ từ Fig.2 đến Fig.4. Các phần chi tiết không được mô tả lại ở đây. Như được thể hiện trên Fig.5, thiết bị phục hồi bộ nhớ 500 có thể bao gồm:

bộ phận xác định 501, được tạo cấu hình để, khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, xác định quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nền sau,

trong đó danh mục xử lý nền sau bao gồm quy trình xử lý của một hoặc nhiều ứng dụng, quy trình xử lý mà trong đó bộ nhớ cần được phục hồi là quy trình xử lý nằm trong số quy trình xử lý của một hoặc nhiều ứng dụng và đáp ứng điều kiện là trị số tuyệt đối của độ sai lệch giữa bộ nhớ được chiếm giữ bởi quy trình xử lý và trị số áp lực bộ nhớ thấp hơn ngưỡng thiết đặt trước, và trị số áp lực bộ nhớ là độ sai lệch giữa ngưỡng bộ nhớ và bộ nhớ hữu dụng hiện thời của hệ thống; và

bộ phận gửi, được tạo cấu hình để gửi lệnh xử lý tới nhân hệ thống, để kích hoạt nhân hệ thống thực hiện việc xử lý trên quy trình xử lý mà trong đó bộ nhớ cần được phục hồi, để phục hồi bộ nhớ được chiếm giữ bởi quy trình xử lý mà trong đó bộ nhớ cần được phục hồi.

[0139] Một cách tùy ý, thao tác xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ được kích hoạt khi sự kiện chính thứ nhất được phát hiện, và ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ nhất tương ứng với sự kiện chính thứ nhất. Sự kiện chính thứ nhất bao gồm bất kỳ một trong số các sự kiện sau đây: sự kiện bắt đầu khởi động chương trình, sự kiện dọn sạch, và sự kiện tràn bộ nhớ OOM.

[0140] Một cách tùy ý, thao tác xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ được kích hoạt khi sự kiện chính thứ hai được phát hiện và khi xác định được rằng hệ thống là ở trạng thái không hoạt động, và ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ hai tương ứng với sự kiện chính thứ hai.

[0141] Một cách tùy ý, thao tác xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ được kích hoạt khi sự kiện chính thứ hai được phát hiện, thao tác xác định quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nền sau được kích hoạt khi xác định được rằng hệ thống là ở trạng thái không hoạt động, và ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ hai tương ứng với sự kiện chính thứ hai. Sự kiện chính thứ hai bao gồm bất kỳ một trong số các sự kiện sau đây: sự kiện hoàn thành khởi động chương trình, sự kiện bật màn hình, sự kiện tắt màn hình, sự kiện chạm màn hình, sự kiện chuyển đổi giao diện người dùng, sự kiện kết thúc chuyển đổi thao tác, và sự kiện phát rộng.

[0142] Một cách tùy ý, bộ phận xác định 501 cụ thể được tạo cấu hình để:

xác định xem tải hiện thời của hệ thống thấp hơn ngưỡng tải hay không; và

khi tải hiện thời của hệ thống thấp hơn ngưỡng tải, xác định rằng hệ thống là ở trạng thái không hoạt động.

[0143] Một cách tùy ý, khi các quy trình xử lý mà trong đó bộ nhớ cần được phục hồi được xác định, bộ phận gửi 502 cụ thể được tạo cấu hình để:

gọi các luồng để gửi các lệnh xử lý tới nhân hệ thống, trong đó mỗi luồng được sử dụng để gửi một hoặc nhiều lệnh xử lý.

[0144] Một cách tùy ý, bộ phận xác định 501 cụ thể được tạo cấu hình để:

xác định, dựa vào thứ tự tăng dần về mức độ quan trọng của ứng dụng, ít nhất một ứng dụng từ các ứng dụng được bao gồm trong danh mục xử lý nền sau; và

xác định, dựa vào thứ tự tăng dần của các quyền ưu tiên xử lý, quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ các quy trình xử lý được bao gồm trong ít nhất một ứng dụng.

[0145] Một cách tùy ý, thiết bị phục hồi bộ nhớ còn bao gồm:

bộ phận tạo 503, được tạo cấu hình để tạo danh mục xử lý nền sau khi bộ phận xác định 501 xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, hoặc được tạo cấu hình để tạo danh mục xử lý nền sau khi bộ phận xác định 501 xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ và hệ thống là ở trạng thái không hoạt động.

[0146] Một cách tùy ý, bộ phận tạo 503 cụ thể được tạo cấu hình để:

xác định chỉ số của thành phần chính của mỗi ứng dụng hiện đang chạy trong nền sau, trong đó thành phần chính bao gồm một hoặc nhiều trong số sau đây: quyền ưu tiên xử lý, thói quen sử dụng của người dùng, tài nguyên hệ thống được chiếm giữ bởi quy trình xử lý, và mối liên quan ứng dụng;

thực hiện phép tính trọng số trên các chỉ số của tất cả các thành phần chính của mỗi ứng dụng, để nhận được mức độ quan trọng của ứng dụng;

phân loại tất cả các ứng dụng dựa vào các mức độ quan trọng của tất cả các ứng dụng; và

phân loại, dựa vào các quyền ưu tiên xử lý, các quy trình xử lý được bao gồm trong mỗi ứng dụng được phân loại, để tạo ra danh mục xử lý nền sau.

[0147] Trong thiết bị phục hồi bộ nhớ được thể hiện trên Fig.5, khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, việc phục hồi bộ nhớ một cách chủ động có thể được thực hiện. Khi chương trình ứng dụng thực sự cần đến một lượng lớn bộ nhớ, nhân hệ thống đã phục hồi một lượng lớn của bộ nhớ khả dụng. Do đó, khả năng các lần dừng gián đoạn của thiết bị đầu cuối có thể được giảm.

[0148] Dựa vào Fig.6, Fig.6 là sơ đồ cấu trúc giản lược của thiết bị đầu cuối được bộc lộ trong một phương án của sáng chế. Thiết bị đầu cuối có thể được tạo cấu hình để thực hiện một vài hoặc tất cả các bước của phương pháp phục hồi bộ nhớ được thể hiện trên các hình vẽ từ Fig.2 đến Fig.4. Đối với thông tin cụ thể của nó, tham chiếu đến các phần mô tả liên quan của các hình vẽ từ Fig.2 đến Fig.4. Các phần chi tiết không được mô tả lại ở đây. Như được thể hiện trên Fig.6, thiết bị đầu cuối 600 bao gồm các thành phần chẳng hạn như mạch tần số radio (Radio Frequency, viết tắt là RF) 601, bộ nhớ 602, bộ phận đầu vào 603, bộ phận hiển thị 604, cảm biến 605, mạch audio 606, môđun mạng không dây sử dụng sóng vô tuyến (wireless fidelity, viết tắt là Wi-Fi) 607, bộ xử lý 608, và nguồn cấp 609. Người có trình độ trung bình trong lĩnh vực kỹ thuật tương ứng có thể hiểu rằng, cấu trúc thiết bị đầu cuối được thể hiện trên Fig.6 không tạo nên sự giới hạn ở điện thoại di động, và thiết bị đầu cuối có thể bao gồm nhiều hoặc một vài thành phần so với các thành phần được thể hiện trên hình vẽ, hoặc một vài thành phần có thể được kết hợp, hoặc các thành phần có thể được bố trí theo cách khác nhau.

[0149] Mạch RF 601 có thể được tạo cấu hình để thu và gửi thông tin, hoặc thu và gửi tín hiệu trong khi đàm thoại. Cụ thể, mạch RF 601 có thể thu thông tin đường xuống của trạm gốc, và gửi thông tin đường xuống tới bộ xử lý 608 cho việc xử lý. Ngoài ra, mạch RF 601 gửi dữ liệu đường lên liên quan tới trạm gốc. Nói chung, mạch RF 601 bao gồm nhưng không giới hạn ở anten, ít nhất một bộ khuếch đại, bộ thu phát, bộ ghép nối, bộ khuếch đại nhiễu thấp (Low Noise Amplifier, viết tắt là LNA), bộ song công, và tương tự. Ngoài ra, mạch RF 601 có thể cũng truyền thông với mạng và các thiết bị khác qua truyền thông radio. Chuẩn hoặc giao thức truyền thông bất kỳ có thể được sử dụng cho việc

truyền thông radio, bao gồm nhưng không giới hạn ở hệ thống truyền thông di động toàn cầu (Global System of Mobile Communications, viết tắt là GSM), dịch vụ radio gói chung (General Packet Radio Service, viết tắt là GPRS), đa truy cập phân chia theo mã (Code Division Multiple Access, viết tắt là CDMA), đa truy cập phân mã dải rộng (Wideband Code Division Multiple Access, viết tắt là WCDMA), phát triển dài hạn (Long Term Evolution, viết tắt là LTE), thư điện tử, và dịch vụ tin nhắn ngắn (short message service, viết tắt là SMS).

[0150] Bộ nhớ 602 lưu trữ chương trình máy tính và chương trình máy tính bao gồm chương trình ứng dụng 6021 và chương trình hệ điều hành 6022. Bộ xử lý 608 được tạo cấu hình để đọc chương trình máy tính trong bộ nhớ 602, và sau đó thực hiện phương pháp được định rõ trong chương trình máy tính. Ví dụ, bộ xử lý 608 đọc chương trình hệ điều hành 6022, để chạy hệ điều hành trên thiết bị đầu cuối 600 và thực hiện các chức năng khác nhau của hệ điều hành. Theo cách khác, bộ xử lý 608 đọc một hoặc nhiều chương trình ứng dụng 6021, để chạy ứng dụng trên thiết bị đầu cuối 600. Hệ điều hành 6022 bao gồm chương trình máy tính mà có thể thực hiện phương pháp phục hồi bộ nhớ được đề xuất theo phương án này của sáng chế, sao cho sau khi bộ xử lý 608 đọc chương trình hệ điều hành 6022 và chạy hệ điều hành, hệ điều hành có thể xử lý chức năng phục hồi bộ nhớ được đề xuất trong các phương án của sáng chế. Ngoài ra, bộ nhớ 602 còn lưu trữ dữ liệu khác 6023 khác với chương trình máy tính, trong đó dữ liệu khác 6023 có thể bao gồm dữ liệu được tạo ra sau khi hệ điều hành 6022 hoặc chương trình ứng dụng 6021 được chạy, và dữ liệu bao gồm dữ liệu hệ thống (ví dụ, các thông số cấu hình của hệ điều hành) và dữ liệu người dùng. Ngoài ra, bộ nhớ 602 thường bao gồm bộ nhớ trong và bộ nhớ ngoài. Bộ nhớ trong có thể là bộ nhớ truy cập ngẫu nhiên (RAM), bộ nhớ chỉ đọc (ROM), bộ nhớ đệm tốc độ cao (CACHE), hoặc tương tự. Bộ nhớ ngoài có thể là đĩa cứng, đĩa quang, ổ nhớ USB, đĩa mềm, ổ đĩa băng, hoặc tương tự. Chương trình máy tính thường được lưu trữ trong bộ nhớ ngoài, và trước khi thực hiện việc xử lý, bộ xử lý tải chương trình máy tính tới bộ nhớ trong từ bộ nhớ ngoài.

[0151] Bộ phận đầu vào 603 có thể được tạo cấu hình để thu thông tin số hoặc ký tự đầu vào và tạo ra đầu vào tín hiệu chính liên quan đến việc điều khiển chức năng và thiết đặt người dùng của thiết bị đầu cuối 600. Cụ thể là, bộ phận đầu vào 603 có thể bao gồm panen cảm ứng 6031 và các thiết bị đầu vào khác

6032. Panen cảm ứng 6031, cũng được gọi là màn hình cảm ứng, có thể thu thập thao tác chạm được thực hiện bởi người dùng trên hoặc trong vùng lân cận của panen cảm ứng 6031 (ví dụ, thao tác được thực hiện trên panen cảm ứng 6031 hoặc trong vùng lân cận của panen cảm ứng 6031 bởi người dùng nhờ sử dụng ngón tay, bút trỏ, hoặc vật hoặc phụ kiện thích hợp bất kỳ khác), và dẫn động thiết bị kết nối tương ứng theo chương trình thiết đặt trước. Một cách tùy ý, panen cảm ứng 6031 có thể bao gồm hai phần: thiết bị phát hiện chạm và bộ điều khiển chạm. Thiết bị phát hiện chạm phát hiện hướng chạm của người dùng, phát hiện tín hiệu được đưa tới bởi thao tác chạm, và truyền tín hiệu tới bộ điều khiển chạm. Bộ điều khiển chạm thu thông tin chạm từ thiết bị phát hiện chạm, chuyển đổi thông tin chạm thành các tọa độ điểm chạm, và gửi các tọa độ điểm chạm tới bộ xử lý 608, và có thể thu lệnh được gửi bởi bộ xử lý 608 và thực hiện lệnh. Ngoài ra, panen cảm ứng 6031 có thể được thực hiện theo nhiều loại, ví dụ, loại điện trở, loại điện dung, loại hồng ngoại, và loại sóng âm bề mặt. Cùng với panen cảm ứng 6031, bộ phận đầu vào 603 có thể còn bao gồm các thiết bị đầu vào khác 6032. Cụ thể là, các thiết bị đầu vào khác 6032 có thể bao gồm nhưng không giới hạn ở một hoặc nhiều trong số bàn phím vật lý, phím chức năng (chẳng hạn như phím điều khiển âm lượng hoặc phím chuyển mạch), bi xoay, chuột, và cần điều khiển.

[0152] Bộ phận hiển thị 604 có thể được tạo cấu hình để hiển thị thông tin được đưa vào bởi người dùng, thông tin được cấp tới người dùng, và các thực đơn khác nhau của điện thoại di động. Bộ phận hiển thị 604 có thể bao gồm panen hiển thị 6041. Một cách tùy ý, panen hiển thị 6041 có thể được tạo cấu hình ở dạng màn hình tinh thể lỏng (Liquid Crystal Display, viết tắt là LCD), điốt phát sáng hữu cơ (Organic Light-Emitting Diode, viết tắt là OLED), hoặc tương tự. Hơn nữa, panen cảm ứng 6031 có thể bao hàm panen hiển thị 6041. Sau khi phát hiện thao tác chạm trên hoặc gần panen cảm ứng 6031, panen cảm ứng 6031 gửi thao tác chạm tới bộ xử lý 608 để xác định loại sự kiện chạm. Sau đó bộ xử lý 608 đưa ra đầu ra trực quan tương ứng trên panen hiển thị 6041 dựa vào loại sự kiện chạm. Mặc dù trên Fig.6, panen cảm ứng 6031 và panen hiển thị 6041 được sử dụng như hai thành phần riêng biệt để thực hiện các chức năng đầu vào và đầu ra của điện thoại di động, trong một vài phương án, panen cảm ứng 6031 có thể được tích hợp với panen hiển thị 6041 để thực hiện các chức năng đầu vào và đầu ra của điện thoại di động.

[0153] Cảm biến 605 có thể là cảm biến ánh sáng, cảm biến chuyển động, hoặc cảm biến khác. Cụ thể là, cảm biến ánh sáng có thể bao gồm cảm biến ánh sáng xung quanh và cảm biến lân cận. Cảm biến ánh sáng xung quanh có thể điều chỉnh độ chói của panen hiển thị 6041 dựa vào độ chói của ánh sáng xung quanh. Cảm biến lân cận có thể tắt panen hiển thị 6041 và/hoặc chiếu sáng nền sau khi điện thoại di động di chuyển tới tại. Với loại cảm biến chuyển động, cảm biến gia tốc có thể phát hiện trị số của gia tốc theo mỗi hướng (thông thường có ba hệ trục), có thể phát hiện trị số và hướng của trọng lực khi điện thoại di động ở trạng thái tĩnh, và có thể được áp dụng tới ứng dụng mà nó nhận diện các thao tác điện thoại di động (ví dụ, chuyển đổi giữa các màn hình ngang và dọc, trò chơi liên quan, và hiệu chỉnh thao tác địa từ), chức năng liên quan đến nhận biết độ rung (chẳng hạn như tĩnh kể và va chạm kể), và tương tự. Điện thoại di động có thể còn được bố trí con quay hồi chuyển, áp kế, ẩm kế, nhiệt kế, cảm biến hồng ngoại, và các cảm biến khác, và chi tiết không được mô tả ở đây.

[0154] Mạch audio 606, loa 6061, và micrô 6062 có thể cung cấp giao diện audio giữa người dùng và điện thoại di động. Mạch audio 606 có thể truyền, tới loa 6061, tín hiệu điện mà nó thu được sau khi chuyển đổi dữ liệu audio thu được, và loa 6061 chuyển đổi tín hiệu điện thành tín hiệu âm thanh và đưa ra tín hiệu âm thanh. Theo khía cạnh khác, micrô 6062 chuyển đổi tín hiệu âm thanh được thu thập thành tín hiệu điện; mạch audio 606 thu tín hiệu điện và chuyển đổi tín hiệu điện thành dữ liệu audio, và đưa ra dữ liệu audio tới bộ xử lý 608 cho việc xử lý; và dữ liệu audio được xử lý được gửi tới, ví dụ, điện thoại di động khác qua mạch RF 601, hoặc dữ liệu audio được đưa ra tới bộ nhớ 602 để xử lý thêm.

[0155] Wi-Fi là công nghệ truyền không dây khoảng cách ngắn, và thiết bị đầu cuối có thể giúp, nhờ sử dụng môđun Wi-Fi 607, người dùng gửi và thu thư điện tử, duyệt trang web, hoặc truy cập đa phương tiện theo dòng chẳng hạn. Môđun Wi-Fi 607 cung cấp sự truy cập Internet dải rộng không dây cho người dùng. Mặc dù Fig.6 thể hiện môđun Wi-Fi 607, có thể hiểu rằng môđun Wi-Fi 607 không phải bộ phận cấu thành cần thiết của thiết bị đầu cuối và có thể được bỏ qua một cách hợp lý như được yêu cầu miễn là bản chất của sáng chế không được thay đổi.

[0156] Bộ xử lý 608 là trung tâm điều khiển của thiết bị đầu cuối, và được tạo

cấu hình để: kết nối các bộ phận khác nhau của toàn bộ thiết bị đầu cuối nhờ sử dụng các giao diện hoặc các đường truyền khác nhau, và thực hiện các chức năng khác nhau của thiết bị đầu cuối và xử lý dữ liệu bằng cách chạy hoặc thực hiện các chương trình và/hoặc các môđun phần mềm được lưu trữ trong bộ nhớ 602 và gọi dữ liệu được lưu trữ trong bộ nhớ 602, để thực hiện toàn bộ việc quản lý trên thiết bị đầu cuối. Một cách tùy ý, bộ xử lý 608 có thể bao gồm một hoặc nhiều bộ xử lý. Ví dụ, bộ xử lý 608 có thể bao gồm một hoặc nhiều bộ xử lý trung tâm, hoặc bao gồm một bộ xử lý trung tâm và một bộ phận xử lý đồ họa. Khi bộ xử lý 608 bao gồm nhiều bộ xử lý, các bộ xử lý có thể được tích hợp trong cùng một chip, hoặc có thể là các chip tách biệt với nhau. Một bộ xử lý có thể bao gồm một hoặc nhiều lõi bộ xử lý.

[0157] Thiết bị đầu cuối 600 còn bao gồm nguồn cấp 609 (ví dụ, pin) để cấp điện tới các thành phần. Tốt hơn là, nguồn cấp có thể được kết nối dưới dạng logic tới bộ xử lý 608 nhờ sử dụng hệ thống quản lý nguồn cấp. Theo cách này, các chức năng chẳng hạn như quản lý nạp, phóng, và tiêu thụ năng lượng được thực hiện nhờ sử dụng hệ thống quản lý nguồn cấp.

[0158] Mặc dù không được thể hiện, thiết bị đầu cuối có thể còn bao gồm camera, môđun Bluetooth, và tương tự, và chi tiết không được mô tả ở đây.

[0159] Theo phương án nêu trên, các bước khác nhau của phương pháp có thể được thực hiện dựa vào cấu trúc thiết bị đầu cuối, và lớp ứng dụng và nhân hệ điều hành đều có thể được xét đến như các thành phần của cấu trúc giảm lược của bộ xử lý 608.

[0160] Theo phương án này của sáng chế, bộ xử lý 608 được tạo cấu hình để thực hiện các thao tác sau đây bằng cách gọi mã chương trình được lưu trữ trong bộ nhớ 602:

 khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, xác định quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nền sau, trong đó danh mục xử lý nền sau bao gồm quy trình xử lý của một hoặc nhiều ứng dụng, quy trình xử lý mà trong đó bộ nhớ cần được phục hồi là quy trình xử lý nằm trong số quy trình xử lý của một hoặc nhiều ứng dụng và đáp ứng điều kiện là trị số tuyệt đối của độ sai lệch giữa bộ nhớ được chiếm giữ bởi quy trình xử lý và trị số áp lực bộ nhớ thấp hơn ngưỡng thiết đặt trước, và trị số áp lực bộ nhớ là độ sai lệch giữa ngưỡng bộ nhớ và bộ

nhớ hữu dụng hiện thời của hệ thống; và

gửi lệnh xử lý tới nhân hệ thống, để kích hoạt nhân hệ thống thực hiện việc xử lý trên quy trình xử lý mà trong đó bộ nhớ cần được phục hồi, để phục hồi bộ nhớ được chiếm giữ bởi quy trình xử lý mà trong đó bộ nhớ cần được phục hồi.

[0161] Thao tác xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ được kích hoạt khi sự kiện chính thứ nhất được phát hiện, và ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ nhất tương ứng với sự kiện chính thứ nhất.

[0162] Sự kiện chính thứ nhất bao gồm bất kỳ một trong số các sự kiện sau đây: sự kiện bắt đầu khởi động chương trình, sự kiện dọn sạch, và sự kiện tràn bộ nhớ OOM.

[0163] Thao tác xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ được kích hoạt khi sự kiện chính thứ hai được phát hiện và khi xác định được rằng hệ thống là ở trạng thái không hoạt động, và ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ hai tương ứng với sự kiện chính thứ hai.

[0164] Thao tác xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ được kích hoạt khi sự kiện chính thứ hai được phát hiện, thao tác xác định quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nền sau được kích hoạt khi xác định được rằng hệ thống là ở trạng thái không hoạt động, và ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ hai tương ứng với sự kiện chính thứ hai.

[0165] Sự kiện chính thứ hai bao gồm bất kỳ một trong số các sự kiện sau đây: sự kiện hoàn thành khởi động chương trình, sự kiện bật màn hình, sự kiện tắt màn hình, sự kiện chạm màn hình, sự kiện chuyển đổi giao diện người dùng, sự kiện kết thúc chuyển đổi thao tác, và sự kiện phát rộng.

[0166] Theo cách thực hiện tùy chọn khác, xác định, bởi bộ xử lý 608, rằng hệ thống là ở trạng thái không hoạt động bao gồm:

xác định xem tài hiện thời của hệ thống thấp hơn ngưỡng tài hay không; và

khi tài hiện thời của hệ thống thấp hơn ngưỡng tài, xác định rằng hệ thống là ở trạng thái không hoạt động.

[0167] Theo cách thực hiện tùy chọn, khi các quy trình xử lý mà trong đó bộ nhớ cần được phục hồi được xác định, việc gửi, bởi bộ xử lý 608, lệnh xử lý tới nhân hệ thống bao gồm bước:

gọi các luồng để gửi các lệnh xử lý tới nhân hệ thống, trong đó mỗi luồng được sử dụng để gửi một hoặc nhiều lệnh xử lý.

[0168] Theo cách thực hiện tùy chọn khác, việc xác định, bởi bộ xử lý 608, quy trình mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nền sau bao gồm các bước:

xác định, dựa vào thứ tự tăng dần về mức độ quan trọng của ứng dụng, ít nhất một ứng dụng từ các ứng dụng được bao gồm trong danh mục xử lý nền sau; và

xác định, dựa vào thứ tự tăng dần của các quyền ưu tiên xử lý, quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ các quy trình xử lý được bao gồm trong ít nhất một ứng dụng.

[0169] Theo cách thực hiện tùy chọn khác, bộ xử lý 608 còn được tạo cấu hình để thực hiện các thao tác sau đây bằng cách gọi mã chương trình được lưu trữ trong bộ nhớ 602:

tạo danh mục xử lý nền sau khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ.

[0170] Theo cách thực hiện tùy chọn khác, bộ xử lý 608 còn được tạo cấu hình để thực hiện các thao tác sau đây bằng cách gọi mã chương trình được lưu trữ trong bộ nhớ 602:

tạo danh mục xử lý nền sau khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ và hệ thống là ở trạng thái không hoạt động.

[0171] Theo cách thực hiện tùy chọn khác, việc tạo, bởi bộ xử lý 608, danh mục xử lý nền sau bao gồm các bước:

xác định chỉ số của thành phần chính của mỗi ứng dụng hiện đang chạy trong nền sau, trong đó thành phần chính bao gồm một hoặc nhiều trong số sau đây: quyền ưu tiên xử lý, thói quen sử dụng của người dùng, tài nguyên hệ thống được chiếm giữ bởi quy trình xử lý, và mối liên quan ứng dụng;

thực hiện phép tính trọng số trên các chỉ số của tất cả các thành phần chính của mỗi ứng dụng, để nhận được mức độ quan trọng của ứng dụng;

phân loại tất cả các ứng dụng dựa vào các mức độ quan trọng của tất cả các ứng dụng; và

phân loại, dựa vào các quyền ưu tiên xử lý, các quy trình xử lý được bao gồm trong mỗi ứng dụng được phân loại, để tạo ra danh mục xử lý nền sau.

[0172] Trong thiết bị đầu cuối 600 được mô tả trên Fig.6, khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, việc phục hồi bộ nhớ một cách chủ động có thể được thực hiện. Khi chương trình ứng dụng thực sự cần đến một lượng lớn bộ nhớ, nhân hệ thống đã phục hồi một lượng lớn của bộ nhớ khả dụng. Do đó, khả năng các lần dừng gián đoạn của thiết bị đầu cuối có thể được giảm.

[0173] Cần lưu ý rằng, để đơn giản phần mô tả, phương pháp nêu trên các phương án được thể hiện dưới dạng các chuỗi thao tác. Tuy nhiên, người có trình độ trung bình trong lĩnh vực kỹ thuật tương ứng sẽ nhận thấy rằng sáng chế không giới hạn ở trình tự thao tác được mô tả, bởi vì theo sáng chế, một vài bước có thể được thực hiện theo các thứ tự khác hoặc được thực hiện đồng thời. Ngoài ra, người có trình độ trung bình trong lĩnh vực kỹ thuật tương ứng sẽ nhận thấy rằng tất cả các phương án được mô tả trong bản mô tả đều là các phương án ví dụ, và các thao tác và các môđun liên quan là không nhất thiết bắt buộc đối với sáng chế.

[0174] Theo các phương án nêu trên, các phần mô tả của các phương án có các sự tập trung tương ứng. Đối với phần mà không được mô tả chi tiết trong một phương án, tham chiếu đến các phần mô tả liên quan trong các phương án khác.

[0175] Theo một vài phương án được đề xuất trong sáng chế này, cần hiểu rằng thiết bị được bộc lộ có thể được thực hiện theo các cách khác. Ví dụ, các phương án về thiết bị được mô tả chỉ là các ví dụ. Ví dụ, sự phân chia bộ phận chỉ là sự phân chia chức năng logic và có thể là sự phân chia khác theo sự thực hiện thực tế. Ví dụ, các bộ phận hoặc các thành phần có thể được kết hợp hoặc được tích hợp trong hệ thống khác, hoặc một vài dấu hiệu có thể được bỏ qua hoặc không được thực hiện. Ngoài ra, các sự ghép nối lẫn nhau hoặc các sự

ghép nối trực tiếp hoặc các sự kết nối truyền thông được thể hiện hoặc được thảo luận có thể là các sự ghép nối gián tiếp hoặc các sự kết nối truyền thông qua một vài giao diện, các thiết bị, hoặc các bộ phận, và có thể được thực hiện ở dạng điện hoặc các dạng khác.

[0176] Các bộ phận được mô tả dưới dạng các thành phần tách biệt có thể hoặc có thể không phải tách biệt về mặt vật lý. Các thành phần được thể hiện dưới dạng các bộ phận có thể hoặc có thể không phải các bộ phận vật lý, và có thể được bố trí ở một vị trí hoặc được phân bố trên các bộ phận mạng. Một vài hoặc tất cả các bộ phận có thể được lựa chọn tùy thuộc vào yêu cầu thực tế để đạt được các mục đích của các giải pháp của các phương án.

[0177] Ngoài ra, các bộ phận chức năng trong các phương án của sáng chế có thể được tích hợp trong một bộ phận xử lý, hoặc mỗi trong số các bộ phận có thể tồn tại đơn lẻ về mặt vật lý, hoặc hai hoặc nhiều các bộ phận được tích hợp thành một bộ phận. Bộ phận được tích hợp có thể được thực hiện ở dạng phần cứng, hoặc có thể được thực hiện ở dạng bộ phận chức năng phần mềm.

[0178] Khi bộ phận được tích hợp được thực hiện ở dạng bộ phận chức năng phần mềm và được bán hoặc được sử dụng dưới dạng sản phẩm độc lập, bộ phận được tích hợp có thể được lưu trữ trong vật ghi đọc được bởi máy tính. Dựa vào cách hiểu như vậy, các giải pháp kỹ thuật của sáng chế về bản chất, hoặc một phần góp phần vào kỹ thuật đã biết, hoặc tất cả hoặc một phần của các giải pháp kỹ thuật có thể được thực hiện ở dạng sản phẩm phần mềm. Sản phẩm phần mềm được lưu trữ trong vật ghi và bao gồm một vài lệnh để lệnh cho thiết bị máy tính (mà có thể là máy tính cá nhân, máy chủ, hoặc thiết bị mạng) thực hiện tất cả hoặc một phần của các bước của các phương pháp được mô tả trong các phương án của sáng chế. Vật ghi nêu trên bao gồm: phương tiện bất kỳ mà có thể lưu trữ mã chương trình, chẳng hạn như ổ nhớ tia chớp USB, bộ nhớ chỉ đọc (Read-Only Memory, viết tắt là ROM), bộ nhớ truy cập ngẫu nhiên (Random Access Memory, viết tắt là RAM), đĩa cứng tháo được, đĩa từ, hoặc đĩa quang.

[0179] Người có trình độ trung bình trong lĩnh vực kỹ thuật tương ứng có thể hiểu rằng tất cả hoặc một số của các bước của các phương pháp trong các phương án có thể được thực hiện bởi chương trình lệnh cho phần cứng liên quan thực hiện. Chương trình có thể được lưu trữ trong vật ghi đọc được bởi máy tính.

Vật ghi có thể bao gồm bộ nhớ tia chớp, bộ nhớ chỉ đọc (Read-Only Memory, viết tắt là ROM), bộ nhớ truy cập ngẫu nhiên (Random Access Memory, viết tắt là RAM), đĩa từ, hoặc đĩa quang.

[0180] Các phương án của sáng chế đã được mô tả chi tiết. Nguyên lý và cách thực hiện của sáng chế đã được làm rõ ở đây qua các ví dụ cụ thể. Phần mô tả về các phương án của sáng chế chỉ được đưa ra để giúp hiểu phương pháp và ý tưởng chính của sáng chế. Ngoài ra, người có trình độ trung bình trong lĩnh vực kỹ thuật tương ứng có thể thực hiện các sự thay đổi và các sự cải biến đối với sáng chế theo các cách thực hiện cụ thể và lĩnh vực ứng dụng dựa vào ý tưởng của sáng chế. Do đó, nội dung của bản mô tả sẽ không được hiểu là giới hạn sáng chế.

YÊU CẦU BẢO HỘ

1. Phương pháp phục hồi bộ nhớ, trong đó phương pháp này bao gồm các bước:

khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, xác định quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nền sau, trong đó danh mục xử lý nền sau bao gồm quy trình xử lý của một hoặc nhiều ứng dụng, quy trình xử lý mà trong đó bộ nhớ cần được phục hồi là quy trình xử lý nằm trong số quy trình xử lý của một hoặc nhiều ứng dụng và đáp ứng điều kiện là trị số tuyệt đối của độ sai lệch giữa bộ nhớ được chiếm giữ bởi quy trình xử lý và trị số áp lực bộ nhớ thấp hơn ngưỡng thiết đặt trước, và trị số áp lực bộ nhớ là độ sai lệch giữa ngưỡng bộ nhớ và bộ nhớ hữu dụng hiện thời của hệ thống; và

gửi lệnh xử lý tới nhân hệ thống, để kích hoạt nhân hệ thống thực hiện việc xử lý trên quy trình xử lý mà trong đó bộ nhớ cần được phục hồi, để phục hồi bộ nhớ được chiếm giữ bởi quy trình xử lý mà trong đó bộ nhớ cần được phục hồi.

2. Phương pháp theo điểm 1, trong đó thao tác xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ được kích hoạt khi sự kiện chính thứ nhất được phát hiện, và ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ nhất tương ứng với sự kiện chính thứ nhất.

3. Phương pháp theo điểm 2, trong đó sự kiện chính thứ nhất bao gồm bất kỳ một trong số các sự kiện sau đây: sự kiện bắt đầu khởi động chương trình, sự kiện dọn sạch, và sự kiện tràn bộ nhớ OOM.

4. Phương pháp theo điểm 1, trong đó thao tác xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ được kích hoạt khi sự kiện chính thứ hai được phát hiện và khi xác định được rằng hệ thống là ở trạng thái không hoạt động, và ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ hai tương ứng với sự kiện chính thứ hai.

5. Phương pháp theo điểm 1, trong đó thao tác xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ được kích hoạt khi sự kiện chính thứ hai được phát hiện, thao tác lựa chọn quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nền sau được kích hoạt khi xác định được rằng hệ thống là ở trạng thái không hoạt động, và ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ hai tương ứng với sự kiện chính thứ hai.

6. Phương pháp theo điểm 4 hoặc 5, trong đó sự kiện chính thứ hai bao

gồm bất kỳ một trong số các sự kiện sau đây: sự kiện hoàn thành khởi động chương trình, sự kiện bật màn hình, sự kiện tắt màn hình, sự kiện chạm màn hình, sự kiện chuyển đổi giao diện người dùng, sự kiện kết thúc chuyển đổi thao tác, và sự kiện phát rộng.

7. Phương pháp theo điểm bất kỳ trong số các điểm từ 4 đến 6, trong đó việc xác định rằng hệ thống là ở trạng thái không hoạt động bao gồm các bước:

xác định xem tải hiện thời của hệ thống thấp hơn ngưỡng tải hay không; và
khi tải hiện thời của hệ thống thấp hơn ngưỡng tải, xác định rằng hệ thống là ở trạng thái không hoạt động.

8. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 7, trong đó khi các quy trình xử lý mà trong đó bộ nhớ cần được phục hồi được xác định, việc gửi lệnh xử lý tới nhân hệ thống bao gồm bước:

gọi các luồng để gửi các lệnh xử lý tới nhân hệ thống, trong đó mỗi luồng được sử dụng để gửi một hoặc nhiều lệnh xử lý.

9. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 7, trong đó việc xác định quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nền sau bao gồm các bước:

xác định, dựa vào thứ tự tăng dần về mức độ quan trọng của ứng dụng, ít nhất một ứng dụng từ các ứng dụng được bao gồm trong danh mục xử lý nền sau; và

xác định, dựa vào thứ tự tăng dần của các quyền ưu tiên xử lý, quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ các quy trình xử lý được bao gồm trong ít nhất một ứng dụng.

10. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 9, trong đó phương pháp còn bao gồm bước:

tạo danh mục xử lý nền sau khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ.

11. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 9, trong đó phương pháp còn bao gồm bước:

tạo danh mục xử lý nền sau khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ và hệ thống là ở trạng thái không hoạt động.

12. Phương pháp theo điểm 10 hoặc 11, trong đó tạo danh mục xử lý nền sau bao gồm:

xác định chỉ số của thành phần chính của mỗi ứng dụng hiện đang chạy trong nền sau, trong đó thành phần chính bao gồm một hoặc nhiều trong số sau đây: quyền ưu tiên xử lý, thói quen sử dụng của người dùng, tài nguyên hệ thống được chiếm giữ bởi quy trình xử lý, và mối liên quan ứng dụng;

thực hiện phép tính trọng số trên các chỉ số của tất cả các thành phần chính của mỗi ứng dụng, để nhận được mức độ quan trọng của ứng dụng;

phân loại tất cả các ứng dụng dựa vào các mức độ quan trọng của tất cả các ứng dụng; và

phân loại, dựa vào các quyền ưu tiên xử lý, các quy trình xử lý được bao gồm trong mỗi ứng dụng được phân loại, để tạo ra danh mục xử lý nền sau.

13. Thiết bị phục hồi bộ nhớ, bao gồm:

bộ phận xác định, được tạo cấu hình để, khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ, xác định quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nền sau, trong đó danh mục xử lý nền sau bao gồm quy trình xử lý của một hoặc nhiều ứng dụng, quy trình xử lý mà trong đó bộ nhớ cần được phục hồi là quy trình xử lý nằm trong số quy trình xử lý của một hoặc nhiều ứng dụng và đáp ứng điều kiện là trị số tuyệt đối của độ sai lệch giữa bộ nhớ được chiếm giữ bởi quy trình xử lý và trị số áp lực bộ nhớ thấp hơn ngưỡng thiết đặt trước, và trị số áp lực bộ nhớ là độ sai lệch giữa ngưỡng bộ nhớ và bộ nhớ hữu dụng hiện thời của hệ thống; và

bộ phận gửi, được tạo cấu hình để gửi lệnh xử lý tới nhân hệ thống, để kích hoạt nhân hệ thống thực hiện việc xử lý trên quy trình xử lý mà trong đó bộ nhớ cần được phục hồi, để phục hồi bộ nhớ được chiếm giữ bởi quy trình xử lý mà trong đó bộ nhớ cần được phục hồi.

14. Thiết bị phục hồi bộ nhớ theo điểm 13, trong đó thao tác xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ được kích hoạt khi sự kiện chính thứ nhất được phát hiện, và ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ nhất tương ứng với sự kiện chính thứ nhất.

15. Thiết bị phục hồi bộ nhớ theo điểm 14, trong đó sự kiện chính thứ nhất bao gồm bất kỳ một trong số các sự kiện sau đây: sự kiện bắt đầu khởi động

chương trình, sự kiện dọn sạch, và sự kiện tràn bộ nhớ OOM.

16. Thiết bị phục hồi bộ nhớ theo điểm 13, trong đó thao tác xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ được kích hoạt khi sự kiện chính thứ hai được phát hiện và khi xác định được rằng hệ thống là ở trạng thái không hoạt động, và ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ hai tương ứng với sự kiện chính thứ hai.

17. Thiết bị phục hồi bộ nhớ theo điểm 13, trong đó thao tác xác định rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ được kích hoạt khi sự kiện chính thứ hai được phát hiện, thao tác lựa chọn quy trình xử lý mà trong đó bộ nhớ cần được phục hồi từ danh mục xử lý nền sau được kích hoạt khi xác định được rằng hệ thống là ở trạng thái không hoạt động, và ngưỡng bộ nhớ là ngưỡng bộ nhớ thứ hai tương ứng với sự kiện chính thứ hai.

18. Thiết bị phục hồi bộ nhớ theo điểm 16 hoặc 17, trong đó sự kiện chính thứ hai bao gồm bất kỳ một trong số các sự kiện sau đây: sự kiện hoàn thành khởi động chương trình, sự kiện bật màn hình, sự kiện tắt màn hình, sự kiện chạm màn hình, sự kiện chuyển đổi giao diện người dùng, sự kiện kết thúc chuyển đổi thao tác, và sự kiện phát rộng.

19. Thiết bị phục hồi bộ nhớ theo điểm bất kỳ trong số các điểm từ 16 đến 18, trong đó bộ phận xác định cụ thể được tạo cấu hình để:

xác định xem tải hiện thời của hệ thống thấp hơn ngưỡng tải hay không; và

khi tải hiện thời của hệ thống thấp hơn ngưỡng tải, xác định rằng hệ thống là ở trạng thái không hoạt động.

20. Thiết bị phục hồi bộ nhớ theo điểm bất kỳ trong số các điểm từ 13 đến 19, trong đó khi các quy trình xử lý mà trong đó bộ nhớ cần được phục hồi được xác định, bộ phận gửi cụ thể được tạo cấu hình để:

gọi các luồng để gửi các lệnh xử lý tới nhân hệ thống, trong đó mỗi luồng được sử dụng để gửi một hoặc nhiều lệnh xử lý.

21. Thiết bị phục hồi bộ nhớ theo điểm bất kỳ trong số các điểm từ 13 đến 19, trong đó bộ phận xác định cụ thể được tạo cấu hình để:

xác định, dựa vào thứ tự tăng dần về mức độ quan trọng của ứng dụng, ít nhất một ứng dụng từ các ứng dụng được bao gồm trong danh mục xử lý nền sau; và

xác định, dựa vào thứ tự tăng dần của các quyền ưu tiên xử lý, quy trình xử

lý mà trong đó bộ nhớ cần được phục hồi từ các quy trình xử lý được bao gồm trong ít nhất một ứng dụng.

22. Thiết bị phục hồi bộ nhớ theo điểm bất kỳ trong số các điểm từ 13 đến 21, trong đó thiết bị phục hồi bộ nhớ còn bao gồm:

bộ phận tạo, được tạo cấu hình để tạo danh mục xử lý nền sau khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ.

23. Thiết bị phục hồi bộ nhớ theo điểm bất kỳ trong số các điểm từ 13 đến 21, trong đó thiết bị phục hồi bộ nhớ còn bao gồm:

bộ phận tạo, được tạo cấu hình để tạo danh mục xử lý nền sau khi xác định được rằng bộ nhớ hữu dụng hiện thời của hệ thống thấp hơn ngưỡng bộ nhớ và hệ thống là ở trạng thái không hoạt động.

24. Thiết bị phục hồi bộ nhớ theo điểm 22 hoặc 23, trong đó bộ phận tạo cụ thể được tạo cấu hình để:

xác định chỉ số của thành phần chính của mỗi ứng dụng hiện đang chạy trong nền sau, trong đó thành phần chính bao gồm một hoặc nhiều trong số sau đây: quyền ưu tiên xử lý, thói quen sử dụng của người dùng, tài nguyên hệ thống được chiếm giữ bởi quy trình xử lý, và mối liên quan ứng dụng;

thực hiện phép tính trọng số trên các chỉ số của tất cả các thành phần chính của mỗi ứng dụng, để nhận được mức độ quan trọng của ứng dụng;

phân loại tất cả các ứng dụng dựa vào các mức độ quan trọng của tất cả các ứng dụng; và

phân loại, dựa vào các quyền ưu tiên xử lý, các quy trình xử lý được bao gồm trong mỗi ứng dụng được phân loại, để tạo ra danh mục xử lý nền sau.

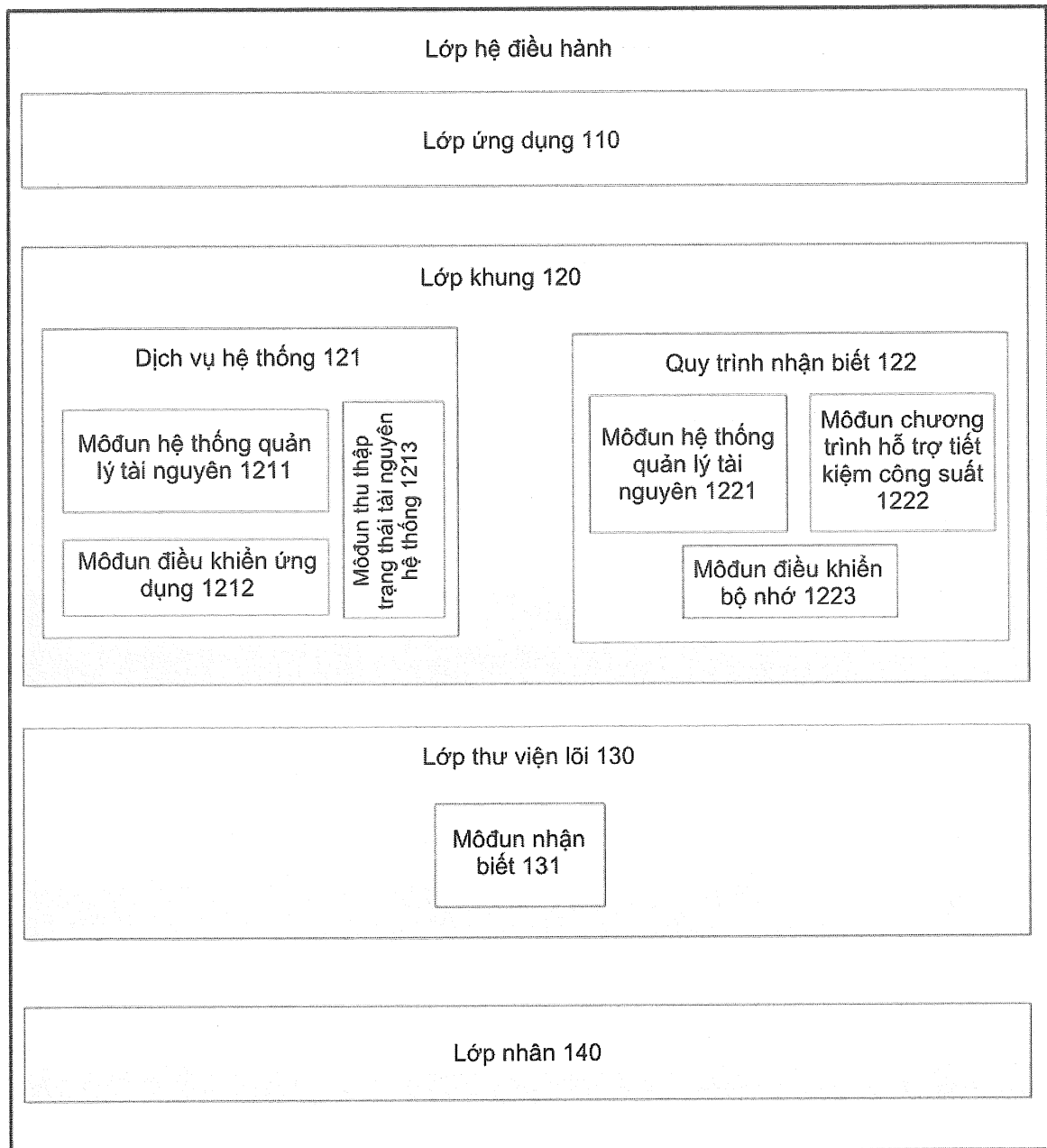


FIG. 1

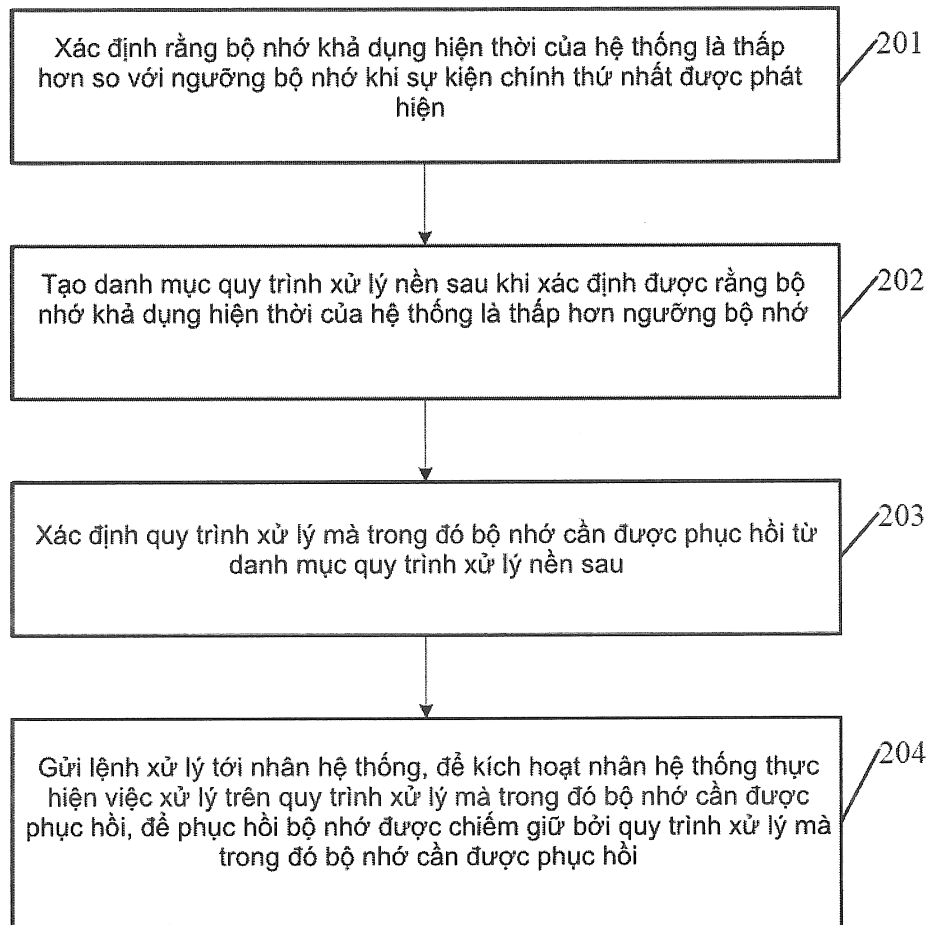


FIG. 2

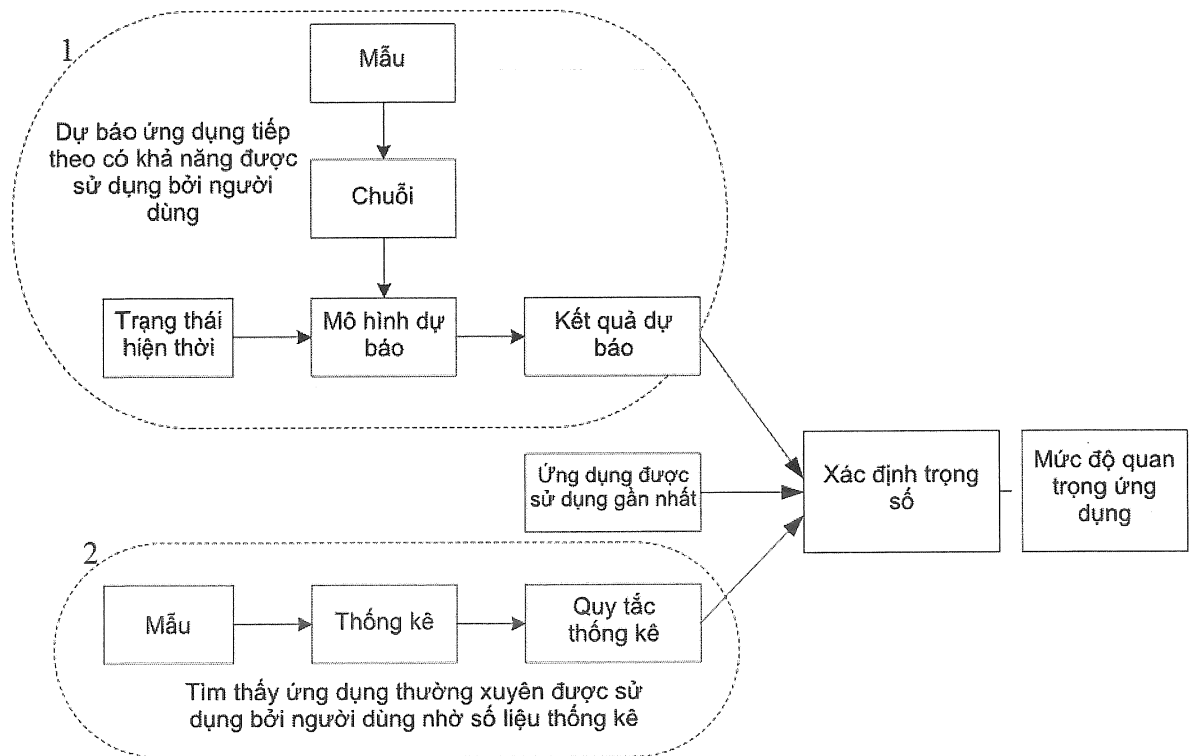


FIG. 2A

4/8

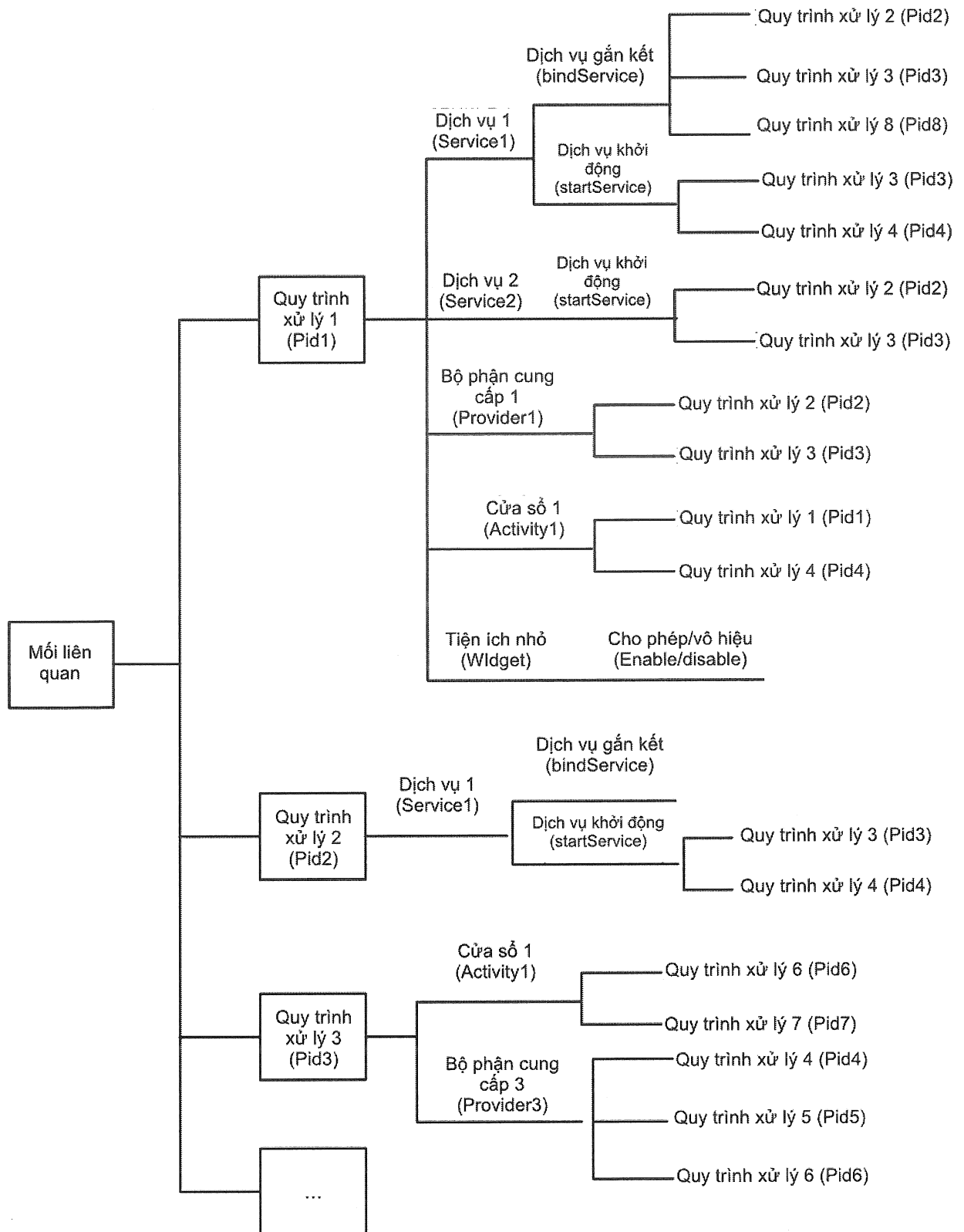


FIG. 2B

5/8

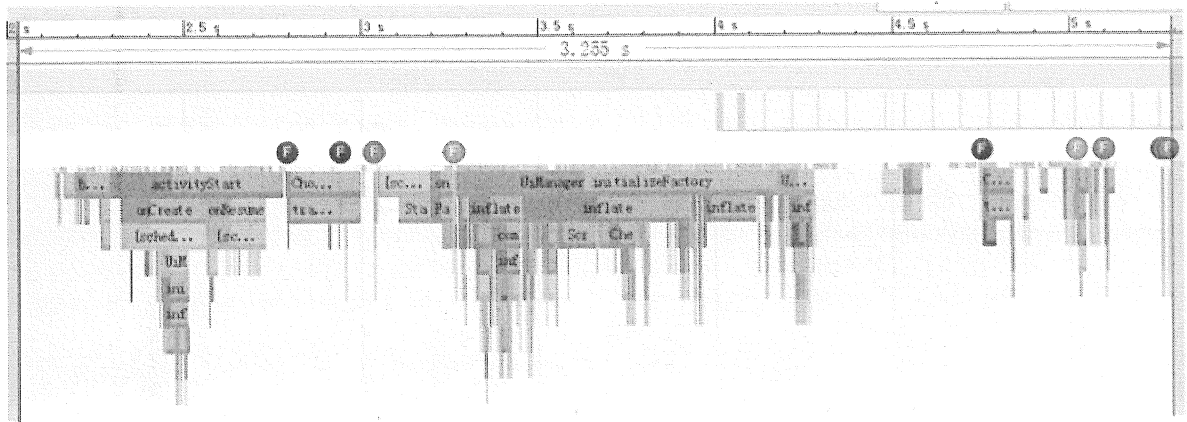


FIG. 2C

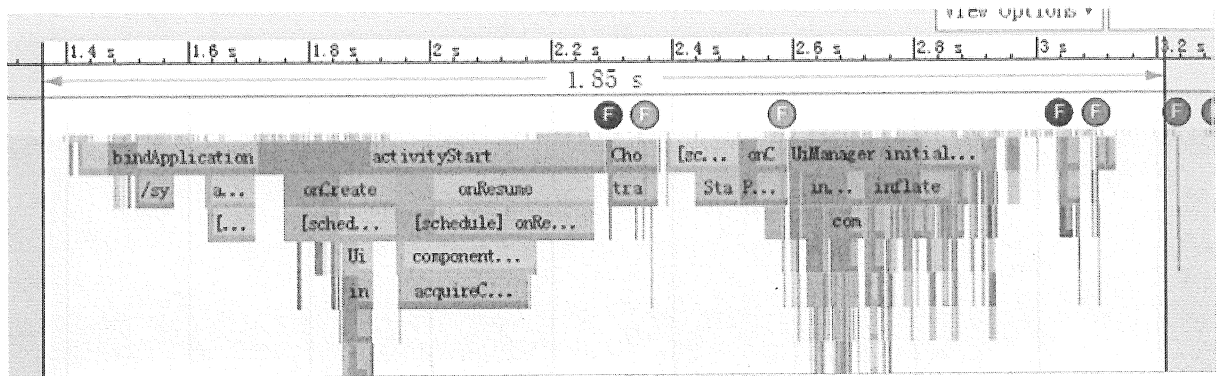


FIG. 2D

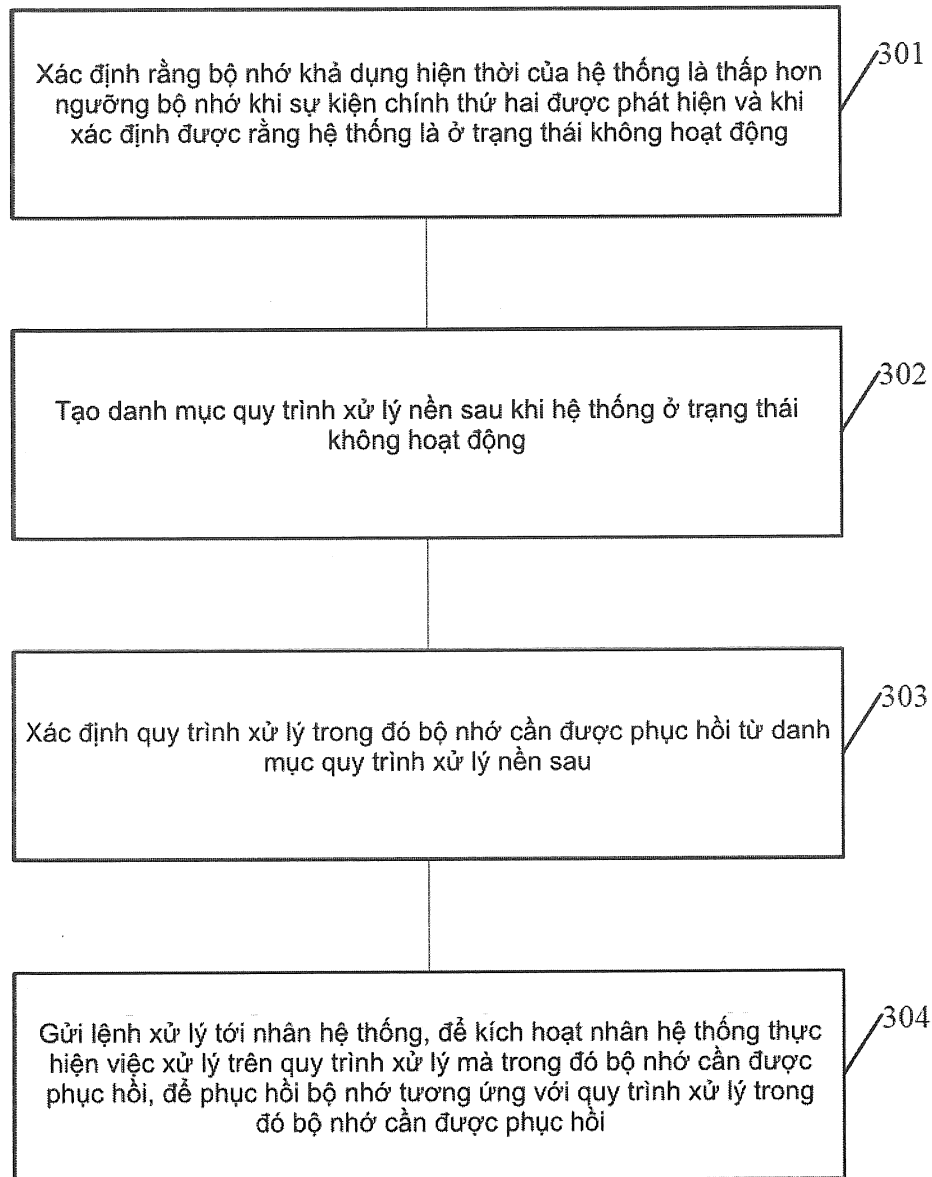


FIG. 3

7/8

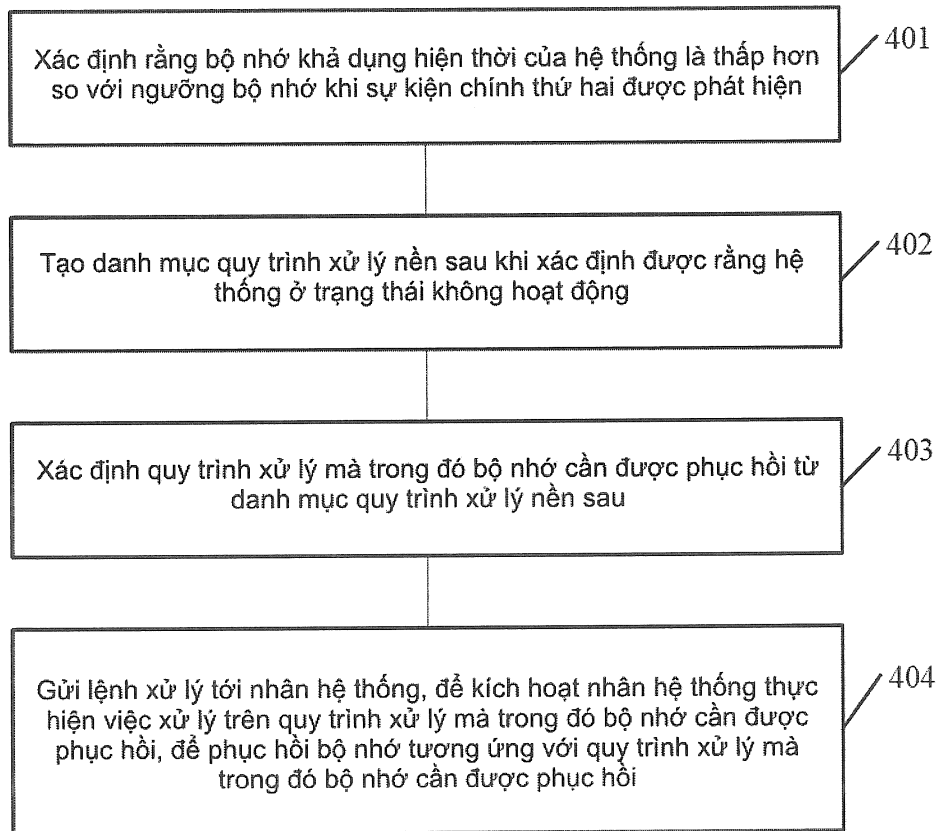


FIG. 4

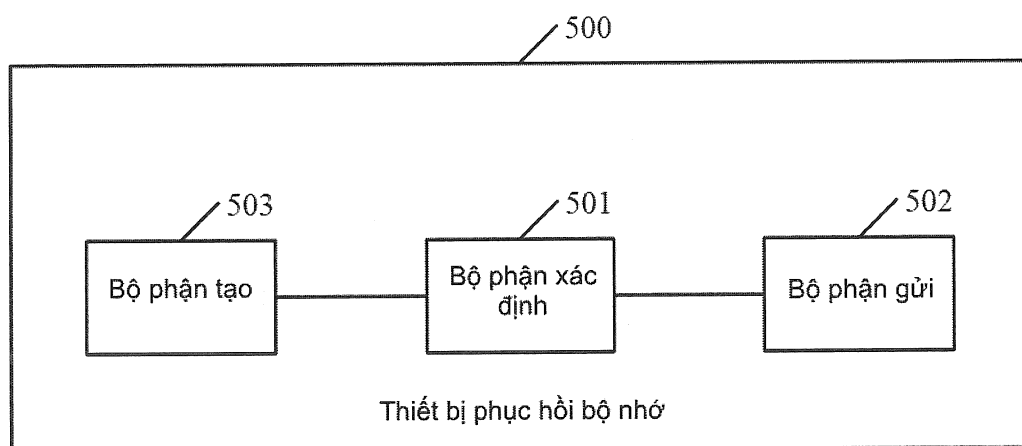


FIG. 5

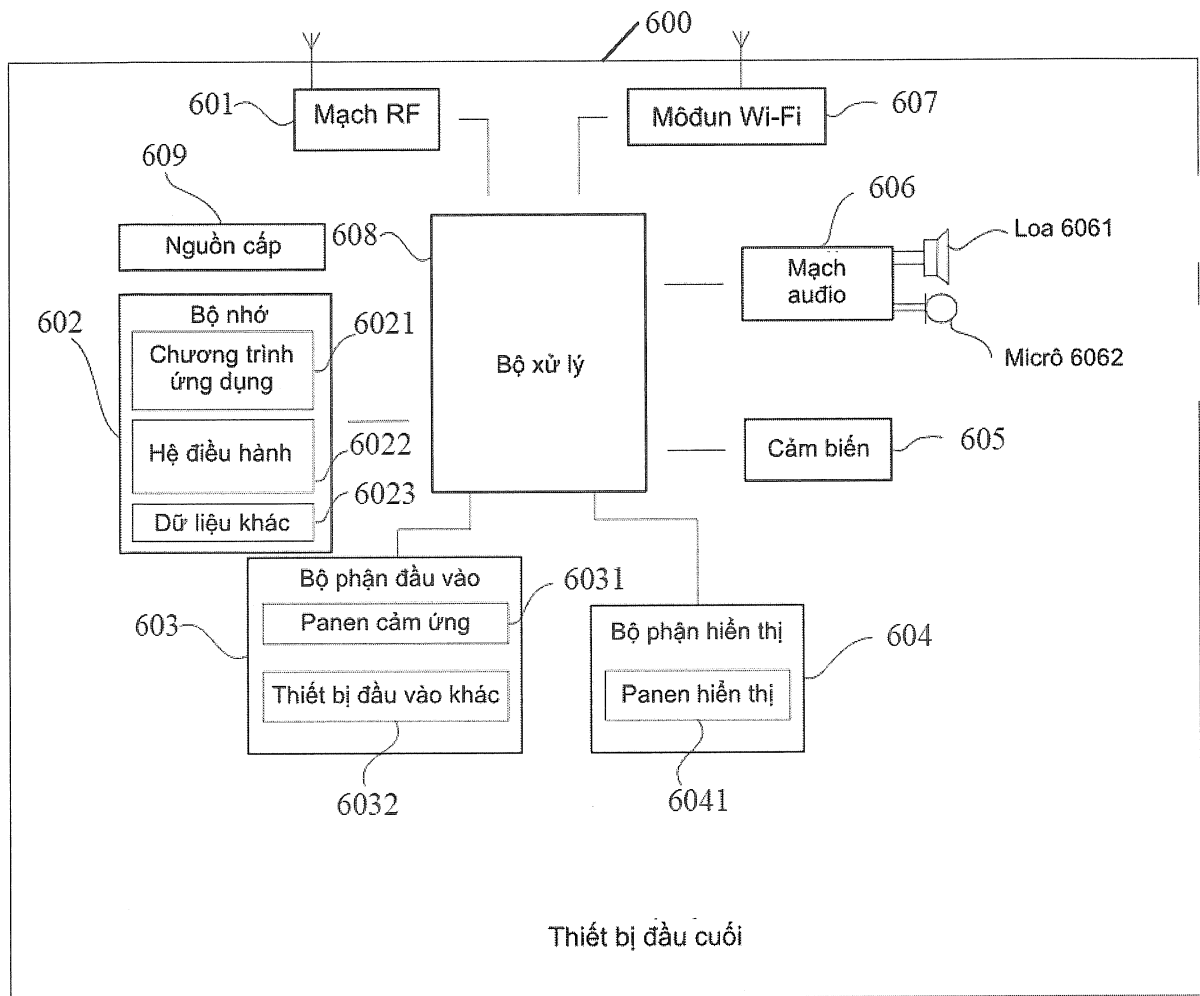


FIG. 6